

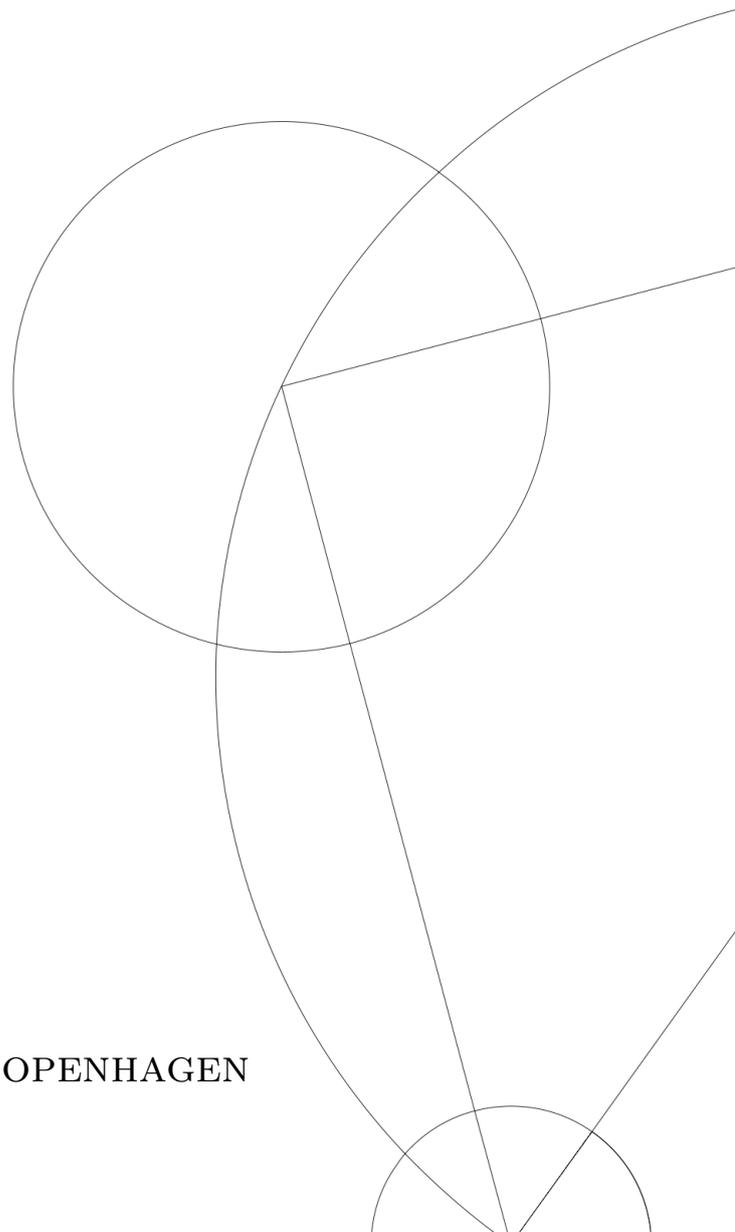


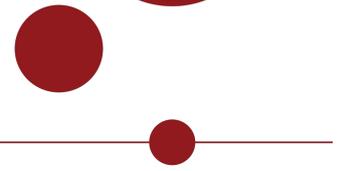
TREE LEVEL COLOUR FACTORS FOR SIMPLE COMPACT LIE GROUPS

Written by *Emil Hoffmann Kozuch*
September 30, 2020

Supervised by
Cristian Vergu and Emil Bjerrum-Bohr

UNIVERSITY OF COPENHAGEN





UNIVERSITY OF
COPENHAGEN

NAME OF INSTITUTE: Niels Bohr Institute
NAME OF DEPARTMENT: Theoretical Particle Physics and Cosmology
AUTHOR(S): Emil Hoffmann Kozuch
EMAIL: ksb688@alumni.ku.dk
TITLE AND SUBTITLE: Tree level colour factors for simple compact Lie groups
-
SUPERVISOR(S): Cristian Vergu and Emil Bjerrum-Bohr
HANDED IN: 30.09.2020
DEFENDED: 20.10.2020

NAME Emil Hoffmann Kozuch

SIGNATURE 

DATE 30/09 - 2020

Gotta Catch 'Em All!

Contents

Abstract	1
Introduction	2
Lie groups and algebras	3
General properties of Lie groups	3
Classification of all simple compact lie groups	5
QFT	8
From Quantum Mechanics to QFT	8
Interacting field theory and Feynman rules	11
Non-abelian gauge theory	14
DDM basis	17
Zeppenfeld basis	19
Birdtrack	24
Definitions	24
Symmetric and anti-symmetric tensors	26
Invariance and projection operators	28
Lie algebra in Birdtrack	32
The four families	35
$SU(N)$	35
$SO(N)$	39
$SP(N)$	42
Results and sanity checks	45
G_2	48
E_6	55
E_7	61
F_4 and E_8	63
Spinors in birdtrack	63
results and sanity checks	67
Summary	68
References	70
Appendix - Python Code	72

Abstract

In this paper two methods for how to calculate contractions of Lie algebra structure constants, for simple compact Lie groups, are derived. The two methods are both based on the pictorial representation of group elements called *Birdtracks*[3]. The easiest algebras to describe are the four families A_n , B_n , C_n and D_n plus g_2 , these algebras are described by the first method which decomposes the spaces $V \otimes V$ or $V \otimes \bar{V}$ into irreducible subspaces described by their primitive invariants. This method however is not able to decompose the algebras f_4 , e_6 , e_7 and e_8 into solvable diagrams. These algebras are instead described using the second method, where the adjoint representation is decomposed into subgroups of the more simple algebras, which allows one to calculate contraction of structure constants. Lastly a method for how to reformulate the colour factors for *tree*-level scattering amplitudes in terms of irreducible representations of the symmetric group enables us to easily compare the result of colour calculations for specific scattering calculations between different algebras.

Introduction

The focus of this thesis is to study some of the most common groups to appear in physics, the simple Compact Lie groups [26]. With the formulation of Yang-Mills theory [25] these groups became a part of the "Cook-book" available for physicists when formulating theories in a quantum field theory framework. In the early 1960's Gell-Mann sought to bring some order to what was known as the particle "Zoo", which in the 1950's was a large amount of newly discovered particles. Using the isospin and hypercharge of these particles he realised in 1961 that they could be organised in way resembling the irreducible representations of $su(3)$, and subsequently noticed a missing part in the diagram and predicted the existence of a new particle Ω^- [10]. This particle was later discovered in 1964 [1]. In that same year Gell-Mann proposed the existence of quarks and gluons [9] with the intent to bring some order to the particle "zoo", this theory used the formalism developed by Yang and Mills, and were realised using the fundamental representation of $su(3)$. Later in 1967 the electromagnetic and weak forces was combined by Weinberg [22] in what is known as the electroweak unification, a theory exhibiting a $su(2)$ symmetry. These theories together make up the very celebrated standard model of particle physics, that describes the interaction of all particles that we know exist.

Even though we have been unable to find evidence of particles outside of the standard model, theorists have not hesitated to further develop and propose new theories in an attempt to solve one of the last puzzles in fundamental physics, Gravity. Because even though the standard model has proven to be extremely successful, it is a well know result that gravity is not renormalizable in a quantum field theory [19]. This has lead to a whole new plethora of theories, from string theory to super-symmetry and conformal field theory. An interesting example of this is the $\mathcal{N} = 8$ supergravity theory by Cremmer and Julia [2], who developed a gravity theory using supersymmetry that exhibits a E_7 duality symmetry with a subgroup of $su(8)$. Another very interesting example where simple compact Lie groups appear is the Chern-Simons theory which is a topological quantum field theory developed by Edward Witten [24] where some of the observable's of the theory are given by Wilson loops [23]. Wittens work on topological quantum field theories was part of the work that earned him a Fields medal.

Given all these examples of simple compact Lie groups appearing in physics it is natural to ask how these groups work and how to do calculations with them. The focus of this thesis is thus to develop a framework in which we can formulate these calculations and to check if the approach is viable. The focus is on the appearance of Lie groups in Yang-Mills theory and the first two chapters are devoted to introducing Lie groups and explaining the formalism of Yang-Mills theory with a focus of explaining where the idea for the quantum field theory framework comes from. The focus will hereafter be on the colour factors of *tree*-level scattering amplitudes, where we in chapter 3 introduce the DDM-basis[4] and show that this basis is valid for all simple compact Lie groups. In the chapter about the Zeppenfeld basis an approach by Zeppenfeld [27] is applied to the DDM-basis, where the colour factors are reformulated in terms of irreducible representations of the permutations groups, with the gauge group factors summed over. What is now left is to calculate the summed over gauge group factors, therefore the notion of birdtracks[3] is introduced, a pictorial way to represent and perform calculations in group theory. Using the birdtrack notion each of the 9 simple compact Lie groups are treated separately, and methods for how to evaluate calculations (or birdtrack diagrams) are developed for each group individually. For the groups A_n ,

B_n , C_n , D_n and G_2 we use the method presented in [3], with decomposing the calculation in terms of primitive invariants of the group. For the groups E_6 , E_7 , E_8 and F_4 we find that it is not possible to find general reduction identities for the primitive invariants, therefore we will decompose these irreducible representations in terms of subgroups we already know how to calculate, the result is that we are only able to perform calculations using the adjoint representation of these groups, although this can surely be extended to other representations. The methods are all verified using known results from the theory of Lie groups.

Lie groups and algebras

In modern physics it is virtually impossible to not encounter group theory in some way, shape or form, when studying physics. Whenever a physical law or specific calculation expresses some kind of symmetry it can be described using group theory and once the specific group "responsible" for the symmetry has been found it can often lead to new discoveries. One of the best examples of this is when Murray Gellmann used an irreducible representation of the $\mathfrak{su}(3)$ algebra to predict the existence of the Ω^- particle, which was later confirmed experimentally [1]. Another way that group theory manifests itself in physics is in quantum mechanics when one solves the Schrödinger equation for the hydrogen atom, where the amount of degenerate eigenstates and how they appear is described by the $\mathfrak{so}(3)$ algebra. The instance of interest for us is how group theory appears in quantum field theory in calculations of scattering amplitudes in Yang-Mills theory, and how to calculate these group factors for the most general case. These Lie groups that appear in Yang-Mills theory are part of a class of groups called semi-simple compact Lie groups, and that is the interest of this section.

General properties of Lie groups

A lie group is a structure that is both a group and a smooth manifold. the fact that it is a smooth manifold means that in a small neighbourhood around the identity the manifold will look like flat euclidean space R^N and the unit vectors for that space T^a , also called generators since they generate the group, are members of the Lie algebra [14]. To go from the algebra to the group elements one can use the exponential map defined as:

$$g(\mathbf{a}) = e^{a^i T^i} \quad (1.1)$$

where the exponential of a matrix is defined as:

$$e^M = \sum_{n=0}^{\infty} \frac{(M)^n}{n!} \quad (1.2)$$

One can view the generators as an infinitesimal transformation around the identity, this can be seen by assuming the values a^i to be small, and then using the exponential map where factors of $O((a^i)^2)$ are ignored, leaving just a linear sum of generators and the identity. We will immediately restrict ourselves to the class of Lie groups called compact Lie groups where the amount of generators are finite and we can represent all these groups as matrix groups.

Like all matrix groups the compact lie groups are closed under multiplication, whereas Lie algebras are closed under under the lie bracket defined as:

$$[T^a, T^b] = f_c^{ab} T^c, \quad [T^a, T^b] = T^a T^b - T^b T^a \quad (1.3)$$

where we have defined the structure constants f_c^{ab} , and we sum over repeated indices. From the above expression we see that the structure constants are anti-symmetric in the first two indices and zero if they are equal to each other. Further the Lie algebra has to satisfy the Jacobi identity given by:

$$[T^a[T^b, T^c]] + [T^b[T^c, T^a]] + [T^c[T^a, T^b]] = 0 \quad (1.4)$$

Taking the trace of equation (1.3) we get:

$$tr([T^a, T^b]) = f_c^{ab} tr(T^c) \Rightarrow \quad (1.5)$$

$$tr(T^a T^b) - tr(T^b T^a) = f_c^{ab} tr(T^c) \Rightarrow \quad (1.6)$$

$$0 = f_c^{ab} tr(T^c) \quad (1.7)$$

Since the generators are the uni vectors, it is not possible to express any of the generators them as a linear combination of the other generators. Therefore we have two choices in eq. (1.7), either the structure constants are all zero, resulting in a rather simple structure, or we can choose the generators to be traceless $tr(T^a) = 0$, which is the choice we make.

Taking the left part part of eq. (1.3) and inserting it in the jacobi identity we get [26]:

$$f_c^{ab} f_g^{dc} + f_d^{bc} f_g^{da} + f_d^{cb} f_g^{db} = 0 \quad (1.8)$$

Defining matrices $(X^a)_c^b = f_c^{ab}$ and using that the two upper indices in the structure constants are anti-symmetric we get:

$$-(X^a)_d^b (X^c)_g^d + (X^c)_d^b (X^a)_g^d = -f_d^{ac} (X^d)_g^b \quad (1.9)$$

$$\Rightarrow [X^a, X^c] = f_d^{ac} X^d \quad (1.10)$$

from this we see that the structure constants themselves satisfy the commutator relation and thus it is valid to use them as generators for the algebra. This representation of the generators is called the adjoint representation and it is by far the most important representation. Later we will see and utilise other representations.

Next thing we define the Cartan-killing metric:

$$g^{ab} = tr(T^a T^b) = -f_d^{ac} f_c^{bd} \quad (1.11)$$

The metric has an inverse given by:

$$g^{ab} g_{bc} = \delta_c^a \quad (1.12)$$

The function of the metric is to raise and lower adjoint indices. So using the metric we can isolate the structure constants at the left side of eq. 1.3:

$$[T^a, T^b] T^d = f_c^{ab} T^c T^d \quad (1.13)$$

$$tr([T^a, T^b] T^d) = f_c^{ab} tr(T^c T^d) \quad (1.14)$$

$$tr([T^a, T^b] T^d) = f_c^{ab} g^{cd} \quad (1.15)$$

$$\frac{1}{C(r)} tr([T^a, T^b] T^d) = f^{abc} \quad (1.16)$$

where we here have defined the totally anti-symmetric structure constants. This constant is pretty important since it allows us to decompose any structure constant in terms of generators in any

representation one can think of. The constant $C(r)$ is the quadratic Casimir, a number that is dependent upon the specific representation in which it is calculated. It is possible also to define an invariant in the algebra called the quadratic invariant, that commutes with all elements of the algebra. This is called the quadratic casimir invariant and is give by:

$$\tilde{C}_2(R) = g_{ab}T^aT^b \quad (1.17)$$

where it satisfies the following condition:

$$[\tilde{C}_2(R), T_a^r] = 0 \quad (1.18)$$

This is proved in the following way:

$$[g_{ab}T^aT^b, T^c] = g_{ab}T^a[T^b, T^c] + g_{ab}[T^a, T^c]T^b \quad (1.19)$$

$$= f_{ad}^c(T)(T^aT^d + T^dT^a) \quad (1.20)$$

$$= 0 \quad (1.21)$$

Classification of all simple compact lie groups

If the algebra is simple the generators can be split into two sets $\{H\}$, $\{E\}$ where the generators H^i mutually commutes with each other:

$$[H^i, H^j] = 0 \quad (1.22)$$

This means that they simultaneously diagonalized and they form a linear space with:

$$H^i |x_i\rangle = x_i |x_i\rangle \quad (1.23)$$

The generators E operates as raising and lowering operators on this linear space. This is a generalisation of the classical case of representations of spin in quantum mechanics. There we define a generator that commutes with itself usually denoted by J_z , and two other generators that acts as raising and lowering operators denoted by J_+ and J_- .

The way we go about doing this is that after we have transformed the generators such that one of them is diagonal, the raising and lowering operators is then two different linear combinations of the two other generators. In this case we have a 1 dimensional ladder, which corresponds to a single generator in the set $\{H\}$. When using the raising generator on the top rung of the ladder the eigenvalue has to be zero, and the same is the case with the bottom rung of the ladder and the lowering operator, where the number of steps on the ladder is equal to the dimension of the matrix representation of the algebra. Each irreducible representation is defined as such a ladder.

Going back to the general case with N generators in the set $\{H\}$, now each H_i represents their own ladder, and can as a whole be viewed as a whole N dimensional lattice where the raising and lowering operators $\{E\}$ moves you around from one point to another in this lattice. Again after we have transformed our generators into a basis where we have the maximum set of $\{H\}$, the raising and lowering operators are linear combinations of the generators not in $\{H\}$. N also defines the rank of the algebra, and it turns out that it is possible to order all simple compact lie algebras and index them by their rank.

Again each representation of the algebra can be viewed as the "closed" set of points on this multidimensional ladder, where the raising and lowering operators either takes you to a point inside

the ladder where the eigenvalues of the $\{H\}$ operators are finite or outside the ladder where the eigenvalues of the $\{H\}$ set are zero. The defining representation is the representation consisting of the fewest number of points on this ladder, that satisfies the above condition, and is larger than just the trivial 1 dimensional representation.

Looking at the generators in the set $\{H\}$, we see that we can rewrite them as:

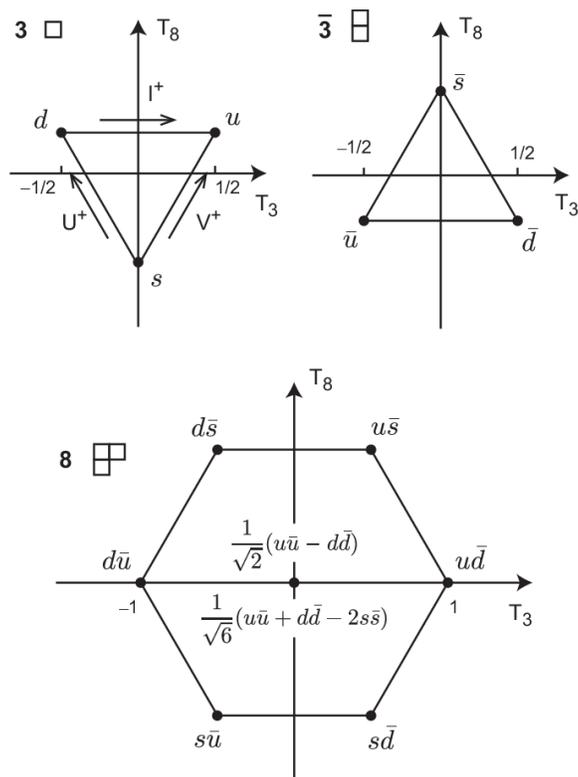


Figure 1.1: The weight diagram for the fundamental 3 dimensional representation of $su(3)$ along with its conjugate representation and the 8 dimensional adjoint representation [17]. Since $su(3)$ has two generators in the set $\{H\}$ the ladders exist in a two dimensional plane

$$f_c^{ab} = (T^a)_c^b = \beta^a(b)\delta_c^b \quad (1.24)$$

Where there is no summation over b . this shows us that we can write the commutator between the two sets $\{H\}$ and $\{E\}$ as:

$$[H^i, E^j] = f_c^{ij} E^c = \beta^i(b)\delta_c^j E^c = \beta^i(b)E^j \quad (1.25)$$

One can then define root vectors:

$$\vec{\beta}(a) = (\beta^1(a), \beta^2(a), \dots, \beta^l(a)) \quad (1.26)$$

Where l is the number of generators in the set $\{H\}$, that explicitly states how to move around in the l dimensional ladder. Now Cartan found that in this basis where the generators are part of either the two sets $\{H\}$ and $\{E\}$, it is possible to make generalised commutator relations between them, these are given by [26]:

- $[H^i, H^j] = 0$
- $[H^i, E_\alpha] = \alpha^i E_\alpha$
- $[E_\alpha, E_\beta] = N_{\alpha,\beta} E_{\alpha+\beta}$
- $[E_\alpha, E_{-\alpha}] = \alpha^i H^i$

These generators turn out to give very strict restrictions on the length of the root vectors and the angles between them and using these restrictions Killing was able to find all simple compact Lie algebras [12]. These algebras can be shown pictorially using their Dynkin diagram: where Dynkin

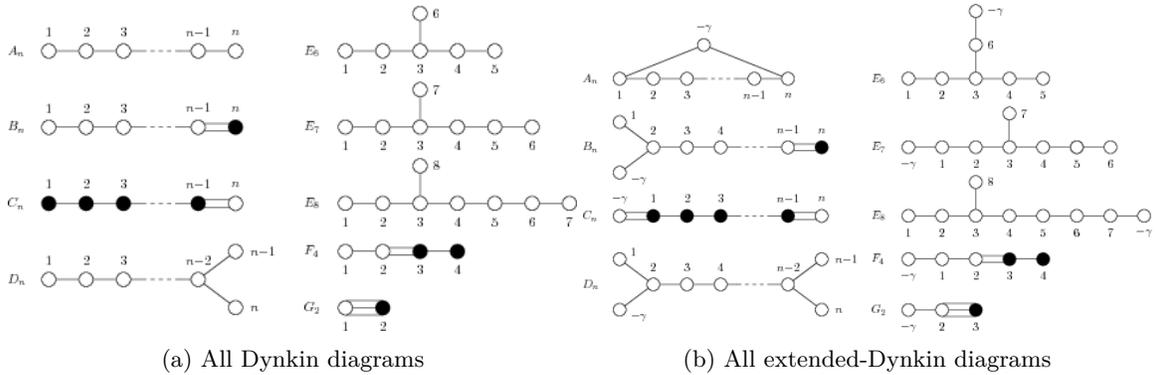


Figure 1.2: Dynkin diagrams [7]. A pictorial way of depicting the relevant information to construct each possible simple compact Lie algebra

diagram is a way of representing the relevant information for constructing the root vectors. Since the dimension of the lattice l is smaller than the number of raising/lowering operators, we only need a subset of the raising/lowering operators to move around in the lattice equal to the dimension of the lattice l . The dots in a Dynkin diagram represent these simple roots. If the angle between the roots is 90 degrees no line is drawn between the roots, and the 3 other possible angles you can have between roots is represented with either 1, 2 or 3 lines. The relative length between the roots is marked by the filling of the dots. If a dot is filled it is the shorter root and if it is unfilled it is the longer root. The extended Dynkin diagrams [28] are interesting in this instance since one can get the subgroups of the specific group by starting with its extended diagram and then removing one of the dots, we will make use of this result later on when we are to calculate contractions of structure constants for some of the exceptional algebras.

QFT

From Quantum Mechanics to QFT

Quantum field theory is arguably the most successful theory ever made. The ideas and formalism it produced laid the foundation for modern physics and the theory itself is still as relevant as it was when it was first written down. The reason for developing QFT sprung from a desire to be able to describe the experimental observations made inside particle accelerators of what happens when two particles collide. What was observed in those experiments was that when two particles collide sometimes other types of particles would emerge from the collision. For a modern physicist it might almost seem like a trivial fact of nature that particles can change into on another, but back then it was disturbing news, since the only viable theory for subatomic particles, Quantum Mechanics, did not allow such conversions. The reason for this is that probabilities in quantum mechanics are conserved:

$$\langle \Psi | \Psi \rangle = 1 \quad (2.1)$$

So in order to describe a theory where particles could disappear and reappear, we needed a theory where a particle is not conserved.

Even though quantum mechanics is insufficient to describe such interactions, our starting point in deriving this theory is still the 1 dimensional Hamiltonian from the Schrödinger equation given by:

$$H(P, Q) = \frac{1}{2m} P^2 + V(Q) \quad (2.2)$$

\hbar is set to 1 for simplicity. What we would like to describe with this general Hamiltonian is the probability for a particle in the state $|q'\rangle$ at time t' to end up in the state $|q''\rangle$ at time t'' , this can of course be described as:

$$\langle q'' | e^{-iH(t''-t')} | q' \rangle \quad (2.3)$$

now taking the time interval $t'' - t'$ and dividing it up into $N+1$ equal time steps and inserting N complete sets of position eigenstates, we get:

$$\langle q'' | e^{-iH(t''-t')} | q' \rangle = \int \prod_{j=1}^N dq_j \langle q'' | e^{-iH\delta t} | q_N \rangle \langle q_N | e^{-iH\delta t} | q_{N-1} \rangle \dots \langle q_1 | e^{-iH\delta t} | q' \rangle \quad (2.4)$$

Where I have set $\delta t = T/(N+1)$, and $T = t'' - t'$

Now consider the state $\langle q_2 | e^{-iH\delta t} | q_1 \rangle$, inserting the Hamiltonian (2.2) and using the Zassenhaus formula [13]:

$$e^{t(X+Y)} = e^{tX} e^{ty} e^{\frac{-t^2}{2}[X,Y]} \dots \quad (2.5)$$

we get:

$$\langle q_2 | e^{-i\frac{\delta t}{2m} P^2} e^{-i\delta t V(Q)} e^{O(\delta t^2)} \dots | q_1 \rangle \quad (2.6)$$

in the limit of small δt , we can discard all terms with order δt^2 or higher and inserting a complete

set of momentum egenstates gives us:

$$\begin{aligned}
\langle q_2 | e^{-iH\delta t} | q_1 \rangle &= \int dp_1 \langle q_2 | e^{-i\frac{\delta t}{2m} P^2} | p_1 \rangle \langle p_1 | e^{-i\delta t V(Q)} | q_1 \rangle \\
&= \int dp_1 e^{\frac{-i\delta t}{2m} p_1^2} e^{-i\delta t V(q_1)} \langle q_2 | p_1 \rangle \langle p_1 | q_1 \rangle \\
&= \int dp_1 e^{\frac{-i\delta t}{2m} p_1^2} e^{-i\delta t V(q_1)} e^{ip_1(q_2 - q_1)} \\
&= \int \frac{dp_1}{2\pi} e^{-iH\delta t} e^{ip_1(q_2 - q_1)}
\end{aligned}$$

inserting this expression into the eq. (2.4) we get:

$$\langle q'' | e^{-iH(t''-t')} | q' \rangle = \int \prod_{k=1}^N dq_k \prod_{j=0}^N \frac{dp_j}{2\pi} e^{ip_j(q_{j+1} - q_j)} e^{-iH\delta t} \quad (2.7)$$

defining $\dot{q}_j = \frac{q_{j+1} - q_j}{\delta t}$ and taking the limit $\delta t \rightarrow 0$ we can rewrite the expression into:

$$\langle q'' | e^{-iH(t''-t')} | q' \rangle = \int \mathcal{D}q \mathcal{D}p e^{i \int_{t'}^{t''} dt (p\dot{q} - H)} \quad (2.8)$$

If H does not contain powers of p higher than p^2 , and i the term that is quadratic in p, then the momentum part of the integral is a Gaussian integral and the prefactors from that integral can be included in the definition of $\mathcal{D}q$, giving us:

$$\langle q'' | e^{-iH(t''-t')} | q' \rangle = \int \mathcal{D}q e^{i \int_{t'}^{t''} dt (L)} \quad (2.9)$$

where the Lagrangian L is defined from the classical expression $L = p\dot{q} - H$. Since we are allowed to include terms linear in q representing external forces, we would like to distinguish those from the original Lagrangian, so that we overall write:

$$\langle q'' | e^{-iH(t''-t')} | q' \rangle = \int \mathcal{D}q e^{i \int_{t'}^{t''} dt (L_0 + f q)} \quad (2.10)$$

Now this is as far as we will get with quantum mechanics, however this equation is still rooted in quantum mechanics, and thus our goal to find a theory where particles can change into one another is still not satisfied. So to proceed from here we need to make some assumptions when changing the equations into what we need and thus we begin with Quantum Field Theory.

The procedure from here is now the following:

Take each q and replace it with $\varphi(x, t)$, and each source f and replace it with $J(x, t)$, also we will now be handling all expressions in 4-dimensions instead of just one (with time and space on an equal footing). Lastly we change the Lagrangian L into the Lagrangian density used for fields \mathcal{L} .

So since we are now using the Lagrangian density the Euler-Lagrange equation is also going to change a bit, so that it now looks like:

$$\partial_\mu \frac{\delta \mathcal{L}}{\delta(\partial_\mu \varphi)} - \frac{\delta \mathcal{L}}{\delta \varphi} = 0 \quad (2.11)$$

Now we turn to the Lagrangian density itself. Since we are dealing with a quantum theory we would appreciate if it is Lorentz-invariant and thus relativistic. The equation of motion for a field that satisfies that condition is given by the Klein-Gordon equation[18]:

$$(\partial^2 + m^2)\varphi = 0 \quad (2.12)$$

so we want to find a Lagrangian density that when it is inserted into the Euler-Lagrange equation to get the equation of motion returns the Klein-Gordon equation.

The Lagrange density that satisfies that condition is given by:

$$\mathcal{L} = \frac{1}{2}\partial_\mu\varphi\partial^\mu\varphi - \frac{1}{2}m^2\varphi^2 \quad (2.13)$$

We now take the right hand side of eq. (2.10), multiply with the ground state ψ_0 , integrate out the momenta q'' and q' , and make the limits $t' \rightarrow -\infty$ and $t'' \rightarrow \infty$. The resulting term we will now interpret as having the same meaning as the partition function from statistical mechanics, and is called the vacuum expectation value:

$$Z_0(J) = \langle 0|0 \rangle_J = \int \mathcal{D}q e^{i \int d^4x (\mathcal{L}_0 + J\varphi)} \quad (2.14)$$

To evaluate Z_0 we focus on the expression in the exponent of e :

$$S_0 = \int d^4x [\mathcal{L} + J\varphi] \quad (2.15)$$

and Fourier transform the fields φ according to:

$$\tilde{\varphi}(k) = \int d^4x e^{-ix} \varphi(x) \quad (2.16)$$

so inserting the Lagrangian density and Fourier transforming the fields we get:

$$S_0 = \frac{1}{2} \int \frac{d^4k}{(2\pi)^4} [-\tilde{\varphi}(k)(k^2 + m^2)\tilde{\varphi}(-k) + \tilde{J}(k)\tilde{\varphi}(-k) + \tilde{J}(-k)\tilde{\varphi}(k)] \quad (2.17)$$

changing integration variable to $\tilde{\chi}(k) = \tilde{\varphi}(k) - \tilde{J}(k)/(k^2 + m^2)$, which is merely just a shift by a constant we get $\mathcal{D}\varphi = \mathcal{D}\chi$ and this leads to:

$$S_0 = \frac{1}{2} \int \frac{d^4k}{(2\pi)^4} \left[\frac{\tilde{J}(k)\tilde{J}(-k)}{k^2 + m^2} - \tilde{\chi}(k)(k^2 + m^2)\tilde{\chi}(-k) \right] \quad (2.18)$$

where the integral over $\mathcal{D}\chi$ can easily be performed by noticing that $Z_0(0) = \langle 0|0 \rangle = 1$, which means that we end up with:

$$Z_0(J) = \mathbf{exp} \left[\frac{i}{2} \int \frac{d^4k}{(2\pi)^4} \frac{\tilde{J}(k)\tilde{J}(-k)}{k^2 - m^2} \right] \quad (2.19)$$

If we now Fourier transform the source terms \tilde{J} we get:

$$Z_0(J) = \mathbf{exp} \left[\frac{i}{2} \int d^4x d^4x' J(x)\Delta(x-x')J(x') \right] \quad (2.20)$$

Where the Feynman propagator $\Delta(x - x')$, defined as:

$$\Delta(x - x') = \int \frac{d^4k}{(2\pi)^4} \frac{e^{ik(x-x')}}{k^2 + m^2} \quad (2.21)$$

is a Green's function to the Klein-Gordon equation:

$$(-\partial_x^2 + m^2)\Delta(x - x') = \delta^4(x - x') \quad (2.22)$$

Interacting field theory and Feynman rules

Now that we have dealt with the free field, it is natural to proceed to an interacting theory.

So lets see what happens when we add a φ^3 term to our Lagrangian:

$$Z(J) = \langle 0|0 \rangle_J = \int \mathcal{D}q e^{i \int d^4x (\mathcal{L}_0 + J\varphi + g\varphi^3)} \quad (2.23)$$

We have also included a coupling constant g in our φ^3 term, that controls the strength of the coupling. To solve this we are going to use a neat little trick. lets say we have an integral of on the form:

$$I = \int dx e^{-x^2 - bx^4} \quad (2.24)$$

if we split up the two exponentials and expand the bx^4 term as a Taylor series we get:

$$I = \int dx e^{-x^2} \left[1 - bx^4 + \frac{(bx^4)^2}{2!} - \dots \right] \quad (2.25)$$

However there is another way of representing eq. (2.24), through a mathematical trick where a generating function is used. This can be done in the following way:

$$I = e^{-b \frac{d^4}{d^4c}} \int dx e^{-x^2 + cx} \Big|_{c=0} \quad (2.26)$$

since the Taylor expansion of the first exponent, will result in the same expression as eq. (2.25). So in order to describe equation we can use J as a generating function in eq. (2.23), so that we can express that equation as:

$$Z = e^{ig \int d^4x \left(\frac{1}{i} \frac{\delta}{\delta J(x)} \right)^3} \int \mathcal{D}q e^{i \int d^4x (\mathcal{L}_0 + J\varphi)} \quad (2.27)$$

but since the left over path integral is nothing more than our free field path integral we can exchange it with that result so that we get:

$$Z = e^{i \int d^4x \left(\frac{1}{i} \frac{\delta}{\delta J(x)} \right)^3} \mathbf{exp} \left[\frac{i}{2} \int d^4x d^4x' J(x) \Delta(x - x') J(x') \right] \quad (2.28)$$

expanding both of these exponentials as Taylor series we get:

$$Z(J) \propto \sum_{V=0}^{\infty} \frac{1}{V!} \left[ig \int d^4x \left(\frac{1}{i} \frac{\delta}{\delta J(x)} \right)^3 \right]^V \sum_{P=0}^{\infty} \frac{1}{P!} \left[\frac{i}{2} \int d^4y d^4z J(y) \Delta(y - z) J(z) \right]^P \quad (2.29)$$

This means that the number of sources that survive the functional derivatives for a particular value of V and P is $E = 2P - 3V$.

The way that we now calculate the vacuum expectation values is to take time ordered products of fields, which is done in the following way:

$$\langle 0|T\varphi(x_1)\varphi(x_2)\dots\varphi(x_n)|0\rangle|_{J=0} = \delta_1\delta_2\dots\delta_n Z_1|_{J=0} \quad (2.30)$$

where δ_n is defined as:

$$\delta_n = \frac{1}{i} \frac{\delta}{\delta J(x_n)} \quad (2.31)$$

So in order for the expression in eq. (2.30) to be anything other than zero, we need the surviving terms to have exactly n sources. These sources will we interpret as particles interacting with each other.

Further the partition function Z can then be viewed as an exponential containing all the diagrams:

$$Z = \mathbf{exp}[iW(J)] \quad (2.32)$$

so that for two particles interacting we get eq. (2.30) to be:

$$\begin{aligned} \langle 0|T\varphi(x_1)\varphi(x_2)|0\rangle|_{J=0} &= \delta_1\delta_2 Z_1(J)|_{J=0} \\ &= \delta_1\delta_2 iW(J)|_{J=0} + \delta_1 iW(J)|_{J=0} \delta_2 iW(J)|_{J=0} \\ &= \delta_1\delta_2 iW(J)|_{J=0} \end{aligned}$$

The fields can be redefined by shifting the vacuum expectation value of the fields by a constant, therefore we define them in way so that $\delta_1 iW(J)|_{J=0} = 0$ since this is a requirement by the LSZ formula that $\langle 0|\varphi(x)|0\rangle = 0$ in order for the LSZ formula to be valid. Where the LSZ formula is the method used to calculate the scattering amplitude of quantum fields interacting. The formula is given by:

$$\langle f|i\rangle = i^{n+n'} \int dx_1^4 e^{ik_1 x_1} (\partial_1^2 + m^2) \dots dx_{1'}^4 e^{-ik_{1'} x_{1'}} (\partial_{1'}^2 + m^2) \dots \langle 0|T\varphi(x_1)\dots\varphi(x_{1'})\dots|0\rangle \quad (2.33)$$

Where the calculation is over the initial to final state of the particles interacting. The last term in the formula is the vacuum expectation value that we have spent some time developing here. The story about removing the single source diagrams is part of a larger story of adding counterterms to the Lagrangian whose sole purpose is to cancel unwanted parts of the final calculation. This is a very important part of QFT and is essential in when calculating any scattering process beyond the most simple ones. However it is a rather involved subject and not essential to this project, so I will not go over it here.

If we repeat our calculation for 2 particles but instead with 4 particles this time, we get:

$$\begin{aligned} \langle 0|T\varphi(x_1)\varphi(x_2)\varphi(x_{1'})\varphi(x_{2'})|0\rangle|_{J=0} &= \delta_1\delta_2\delta_{1'}\delta_{2'} Z_1(J)|_{J=0} \\ &= \delta_1\delta_2\delta_{1'}\delta_{2'} iW(J)|_{J=0} + \delta_1\delta_2 iW(J)|_{J=0} \delta_{1'}\delta_{2'} iW(J)|_{J=0} \\ &\quad + \delta_1\delta_{1'} iW(J)|_{J=0} \delta_2\delta_{2'} iW(J)|_{J=0} + \delta_1\delta_{2'} iW(J)|_{J=0} \delta_{1'}\delta_2 iW(J)|_{J=0} \end{aligned}$$

Here $Z_1(J)$ stands for all diagrams including disconnected diagrams, and $W(J)$ is defined as all connected diagrams. and we choose to only consider connected diagrams. The three terms where the functional derivatives are divided up does not contribute to the scattering calculation of the four particles since in these terms the particles do only interact pairwise. Therefore another choice is taken, that we will only look at the diagrams that actually contribute to the scattering process of all the particles, which is the terms where all the functional derivatives act on the same $W(J)$. The final expression is therefore given by:

$$\langle 0|T\varphi(x_1)\varphi(x_2)\varphi(x_{1'})\varphi(x_{2'})|0\rangle|_{J=0} = \delta_1\delta_2\delta_{1'}\delta_{2'}Z_1(J)|_{J=0} \quad (2.34)$$

Since there are 4 functional derivatives and 4 external sources there are $4!$ ways of combining those up with each other. Thus in total we will get 24 expressions for each term that survives the expansion in eq. (2.29). Assuming that the coupling constant is small, we can direct our focus on the term that gives the lowest order in g . The 24 terms for that diagram turns out to be divided into 3 groups each with 8 identical diagrams, so that we in total end up with:

$$\begin{aligned} \langle 0|T\varphi(x_1)\varphi(x_2)\varphi(x_{1'})\varphi(x_{2'})|0\rangle|_{J=0} &= (ig)^2 \left(\frac{1}{i}\right)^5 \int d^4z d^4y \Delta(y-z) \\ &\cdot [\Delta(x_1-y)\Delta(x_2-y)\Delta(x_{1'}-z)\Delta(x_{2'}-z) \\ &+ \Delta(x_1-y)\Delta(x_{1'}-y)\Delta(x_2-z)\Delta(x_{2'}-z) \\ &+ \Delta(x_1-y)\Delta(x_{2'}-y)\Delta(x_2-z)\Delta(x_{1'}-z)] \end{aligned}$$

These three calculations can then each be represented with a diagram:

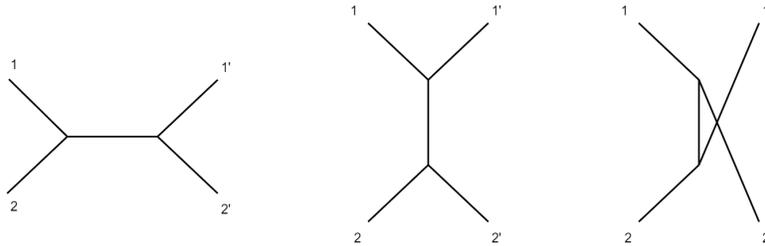


Figure 2.1: The 3 diagrams that are the result of eq. (2.34)

These types of diagrams are known as Feynman diagrams, and the specific diagrams we are dealing with here is the tree diagrams. Tree diagrams are all the diagrams that do not involve any kind of closed loop, and they all have the same order of the coupling constant g . The rule for creating these diagrams is as follows:

If you have n particles interacting with each other you need to have n lines that have one end not

connected to anything. Then look at the interacting term, the amount of factors in that term is the number of lines that should exit from each vertex, so for a φ^3 theory we have to connect the lines in vertices of three each. If it had been a φ^4 theory we would have had to connect 4 lines to each vertex. Then construct all possible topological distinct diagrams. In the past calculation we only included the terms with the lowest order of g , but in the general case there will of course be plenty of diagrams that have an order of g much larger, actually an infinite amount of them. It turns out that each time a diagram goes up one order in g (and because g is assumed small then gets a lower overall value and contributes less) the corresponding diagram gets an extra loop where momenta can circle around in. This is the reason that corrections to the tree diagrams are called loop corrections or loop diagrams. In this thesis I will however not focus on loops.

The value of each diagram can then be calculated by letting the momentum flow through the lines from left to right. Then to construct the calculation of the diagrams simply set up a chain of propagators such that they represent the flow of momenta through each diagram.

Non-abelian gauge theory

Now that we have described how a scattering processes in QFT works and how Feynman diagrams arise lets move straight on to the meat of the matter. What we have described so far is interactions for a field with spin=0, what we could do now is to introduce fields with spin=1/2, and from there develop the theory for fermions. However that is not the focus of this thesis, instead we will jump directly on to describing bosons, particles that have spin=1[18].

First let's start with observing how we could make a Lagrangian for the most simple spin 1 field, the photon. We know that photons is nothing more than electromagnetic waves, and that these waves can be described by Maxwell's equations. From Maxwell's equation it can further be deduced that the electric and magnetic fields can be described by two other fields, a vector and scalar field that combined has a lower dimensionality than the electric and magnetic field, that are given by:

$$\mathbf{E} = -\nabla\varphi - \dot{\mathbf{A}} \quad (2.35)$$

$$\mathbf{B} = \nabla \times \mathbf{A} \quad (2.36)$$

thus simplifying all equations involving electric and magnetic fields. However these new fields are not uniquely defined, and by shifting them in a specific way, the magnetic and electric fields will remain unchanged. These shifts are given by:

$$\varphi' = \varphi + \dot{\Gamma} \quad (2.37)$$

$$\mathbf{A}' = \mathbf{A} - \nabla\Gamma \quad (2.38)$$

and are called gauge transformations. By switching to relativistic notation we can make it all a little more compact by introducing $A_\mu = (\varphi, \mathbf{A})$, then we define the field strength by:

$$F^{\mu\nu} = \partial^\mu A^\nu - \partial^\nu A^\mu \quad (2.39)$$

Maxwell's equations can then be reframed as:

$$\partial_\nu F^{\mu\nu} = J^\mu \quad (2.40)$$

$$\epsilon_{\mu\nu\rho\sigma} \partial^\rho F^{\mu\nu} = 0 \quad (2.41)$$

where $J^\mu = (\rho, \mathbf{J})$.

The gauge transformation of eq. 2.37 and 2.38, can be expressed as $A'^\mu = A^\mu - \partial^\mu \Gamma$, so gauge transforming the field strength $F^{\mu\nu}$ we get:

$$F'^{\mu\nu} = \partial^\mu A'^\nu - \partial^\nu A'^\mu \quad (2.42)$$

$$= \partial^\mu (A^\nu + \partial^\nu \Gamma) - \partial^\nu (A^\mu + \partial^\mu \Gamma) \quad (2.43)$$

$$= \partial^\mu A^\nu - \partial^\nu A^\mu + (\partial^\mu \partial^\nu - \partial^\nu \partial^\mu) \Gamma \quad (2.44)$$

$$= \partial^\mu A^\nu - \partial^\nu A^\mu + (\partial^\mu \partial^\nu - \partial^\mu \partial^\nu) \Gamma \quad (2.45)$$

$$= \partial^\mu A^\nu - \partial^\nu A^\mu = F^{\mu\nu} \quad (2.46)$$

which shows us that the field strength is gauge invariant. Proceeding as with the free scalar field, what we would like to do now is to find a Lagrangian that when put into the Euler-Lagrange equation returns the equation of motion, which in this case are Maxwell's equations. The solution to this is given by:

$$\mathcal{L} = \frac{-1}{4} F^{\mu\nu} F_{\mu\nu} + J^\mu A_\nu \quad (2.47)$$

This equation is also Lorentz invariant, Gauge invariant, parity and time reversal invariant (two symmetries that I do not have time to cover here). Now that we have the Lagrangian the way we solve the path integral and develop interactions follows the same path as for the scalar fields, albeit with a few extra hurdles such as having to fix the gauge freedom.

An interesting fact however is that the gauge invariance shown here is not the most general case that we can construct and the general is what we will explore now. When doing quantum electrodynamics one finds that QED can be understood as having a local U(1) symmetry, such that the electron fields transforms as $\varphi(x) \rightarrow U(x)\varphi$. The only way this could be a symmetry of the QED Lagrangian is if all derivatives are replaced with covariant derivatives, not unlike what one needs to do in general relativity, given by $D_\mu = \partial_\mu - ieA_\mu$. Generalising this to the most general case we can define the gauge fields A_μ to be matrices that exhibit the same properties as the generators of the group elements $U(x)$ that we want it to be invariant under. Then we define the gauge transformation of the fields as:

$$A_\mu \rightarrow U(x)A_\mu U^{-1}(x) + \frac{i}{e} U(x) \partial_\mu U^{-1}(x) \quad (2.48)$$

the covariant derivative then also transforms as:

$$D_\mu \rightarrow U(x)D_\mu D^{-1}(x) \quad (2.49)$$

and is given by:

$$D_\mu = \partial_\mu - igA_\mu \quad (2.50)$$

where the partial derivative has an identity matrix multiplied with the same dimension as the gauge fields. If we look at U(1) with $U(x) = \mathbf{exp}[-ie\Gamma(x)]$, we see that the transformation of the gauge field returns what we got for Maxwell's equations. Now instead of U(1) we can define $U(x)$ more generally as:

$$U(x) = e^{-ig\Gamma^a(x)T^a} \quad (2.51)$$

where the T^a 's are generators of a compact Lie group. The field strength of this theory can then be defined as:

$$F = \frac{i}{g} [D_\mu, D_\nu] = \partial_\mu A_\nu - \partial_\nu A_\mu - ig [A_\mu, A_\nu] \quad (2.52)$$

if it had been like in QED the last terms would have been equal to zero, since the gauge fields here were not matrices. But now since they are matrices they do not disappear, this is known as Yang-Mills theory [25]. Using that the gauge fields A_μ exhibits the same properties as the generators of the gauge group, we can expand it in terms of these generators:

$$A_\mu = A_\mu^c(x) T^c \quad (2.53)$$

Putting this into the definition of the field strength we get:

$$F_{\mu\nu} = (\partial_\mu A_\nu^c - \partial_\nu A_\mu^c) T^c - ig A_{m\mu}^a ig A_{n\nu}^b [T^a, T^b] \quad (2.54)$$

$$= (\partial_\mu A_\nu^c - \partial_\nu A_\mu^c) T^c - ig A_{m\mu}^a ig A_{n\nu}^b f^{abc} T^c \quad (2.55)$$

$$= F_{\mu\nu}^c T^c \quad (2.56)$$

The field strength transforms as $F^{\mu\nu} \rightarrow U(x) F^{\mu\nu} U^{-1}(x)$ it is not gauge invariant, so the usual expression $F^{\mu\nu} F_{\mu\nu}$ can not be used as the kinetic term. however to create a gauge invariant term we simply use:

$$\begin{aligned} \text{tr}(F^{\mu\nu} F_{\mu\nu}) &\rightarrow \text{tr}(U(x) F^{\mu\nu} U(x)^{-1} U(x) F_{\mu\nu} U(x)^{-1}) \\ &= \text{tr}(F^{\mu\nu} U(x)^{-1} U(x) F_{\mu\nu} U(x)^{-1} U(x)) \\ &= \text{tr}(F^{\mu\nu} F_{\mu\nu}) \\ &= F^{\mu\nu a} F_{\mu\nu}^b \text{tr}(T^a T^b) \end{aligned}$$

The last equal sign is not part of the gauge invariance for the kinetic term, however it is paramount for what we are doing here. Because the reason that we used semi-simple lie group generators in eq. (2.51) is because those generators are compact, which means that $\text{tr}(T^a T^b) \geq 0$, and with this condition we make sure that we do not end up with any kinetic terms that have the wrong sign, since this could result in a Hamiltonian that is unbounded from below. Therefore in a gauge theory we are restricted to only use the following groups or products of these groups as our gauge groups: $A_n, B_n, C_n, D_n, G_2, F_4, E_6, E_7$ and E_8 .

Now what would normally be done is to solve the path integral for the Lagrangian $\mathcal{L} = -\frac{1}{2} \text{tr}(F^{\mu\nu} F_{\mu\nu})$, so that we can get the propagator and essentially develop it all in the same manner as we did for a spin 0 field. However this is rather cumbersome to go over and for our purpose it is not needed, since many of the results will turn out to be very similar to the ones for spin 0 fields, just with some precautions due to the gauge invariance. Therefore writing out the Lagrangian we get:

$$\mathcal{L} = -\partial^\mu A^{\nu c} \partial_\mu A_\nu^c + \partial^\mu A^{\nu c} \partial_\nu A_\mu^c - g f^{abc} A^{a\mu} A^{b\nu} \partial_\mu A_\nu^c - \frac{1}{4} g^2 f^{abe} f^{cde} A^{a\mu} A^{b\nu} A_\mu^c A_\nu^d \quad (2.57)$$

The Feynman rules for how to draw diagrams that we have already developed almost translates directly to our gauge invariant theory, therefore we get a 3 vertex diagram from the terms with 3 gauge fields and a 4 vertex diagram from the term with four gauge fields. We can draw this as shown in figure (2.2):

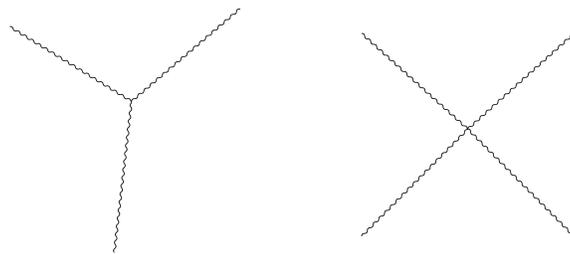


Figure 2.2: The vertices we have for our free field gauge theory

Then when drawing Feynman diagrams for a scattering amplitude for purely gluonic calculations these are the fundamental diagrams that we need to use. These diagrams each contain one or two factors of the Lie algebra structure constants structure f^{abc} , and so their respective scattering amplitudes also contains factors of structure constants. When summing over all the diagrams that arise from a tree level calculation of gluonic scattering we do not end up with the same expression as we did in the spin 0 case. What we instead end up with is an expression where we have expanded all structure constants in terms of their fundamental representation and collected the terms that share the same group factor. This all results in a formula for the amplitude that is given by:

$$\mathcal{M} = \sum_{\pi \in S_n} M(\pi_1, \pi_2, \dots, \pi_n) \text{tr}(T^{\pi_1} T^{\pi_2} \dots T^{\pi_n}) \quad (2.58)$$

Where the sum is over all permutations of the symmetric group S_n [21]. Here we see that each element in the amplitude is split up in a kinematic part and group part. Our focus for the rest of this paper will be to find a way to calculate the traces of structure constants also called colour factors.

DDM basis

From the last section we know that the standard way of calculating scattering amplitudes for gauge fields is given by eq. (2.58):

$$\mathcal{M}(1, 2, \dots, n) = \sum_{\pi \in S_n} M(a_{\pi_1}, a_{\pi_2}, \dots, a_{\pi_n}) \text{tr}(T^{a_{\pi_1}} T^{a_{\pi_2}} \dots T^{a_{\pi_n}}) \quad (3.1)$$

Furthermore from calculations of gluon amplitudes [6] we know that the kinematic part of the subamplitudes has a cyclic invariance:

$$M(\pi_1, \pi_2, \dots, \pi_n) = M(\pi_2, \dots, \pi_n, \pi_1) \quad (3.2)$$

which is also shared by the traces (since traces of matrices are cyclic invariant), this enables us to fix one of the "legs" in the amplitude, which means that we can reduce the above expression to:

$$\mathcal{M}_n = \sum_{\pi \in S_{n-1}} M(\pi_1, a_{\pi_2}, \dots, a_{\pi_n}) \text{tr}(T^{\pi_1} T^{a_{\pi_2}} \dots T^{a_{\pi_n}}) \quad (3.3)$$

Since the trace of a product of matrices is cyclic invariant independent of which matrices are multiplied together, and since the kinematic factors does not depend on the group symmetry, this decomposition holds for any of the semi-simple compact lie groups (SU(n), SO(n), SP(n), G_2 , F_4 , E_6 , E_7 and E_8).

The reason we want to reduce the amount of expressions in the sum is simply because it will make it easier for us in the future to calculate scattering amplitudes if we have fewer terms in our expression. Especially because the hard part to calculate is often the kinematic factors, so the fewer off them we have, the better of we are. Luckily for us, it is however possible to further reduce the expression for the scattering amplitude in eq. (3.3), using a decomposition shown in [4], called the DDM basis.

The DDM basis builds on the decomposition from eq. (3.3) and utilises two different relations in order to be proved, the first is:

$$f^{\pi_1 \pi_2 x_1} f^{x_1 \pi_3 x_2} \dots f^{x_{n-3} \pi_{n-1} \pi_n} = (-1)^{n-2} \text{tr}(T^{\pi_1} [T^{\pi_2}, [T^{\pi_3}, \dots, [T^{\pi_{n-1}}, T^{\pi_n}] \dots]]) \quad (3.4)$$

This relation can be derived from the commutator relation, and the definition of the structure constants:

$$[T^a, T^b] = i f^{abc} T^c \Rightarrow (-i)[T^a, T^b] = f^{abc} T^c \quad (3.5)$$

$$f^{abc} = -i \text{tr}(T^a [T^b, T^c]) \quad (3.6)$$

So the calculation goes like this:

$$\begin{aligned} f^{\pi_1 \pi_2 x_1} f^{x_1 \pi_3 x_2} f^{x_2 \pi_4 \pi_5} &= (-i) \text{tr}(T^{\pi_1} [T^{\pi_2}, T^{x_1}]) f^{x_1 \pi_3 x_2} f^{x_2 \pi_4 \pi_5} \\ &= (-i) \text{tr}(T^{\pi_1} [T^{\pi_2}, T^{x_1} f^{x_1 \pi_3 x_2}]) f^{x_2 \pi_4 \pi_5} \\ &= (-i)^2 \text{tr}(T^{\pi_1} [T^{\pi_2}, [T^{\pi_3}, T^{x_2}]]) f^{x_2 \pi_4 \pi_5} \\ &= (-i)^2 \text{tr}(T^{\pi_1} [T^{\pi_2}, [T^{\pi_3}, T^{x_2} f^{x_2 \pi_4 \pi_5}]]) \\ &= (-i)^3 \text{tr}(T^{\pi_1} [T^{\pi_2}, [T^{\pi_3}, [T^{\pi_4}, T^{\pi_5}]]]) \end{aligned}$$

From here it is easy to see how to generalise it to the case of $n - 2$ structure constants, and N generators in the fundamental representation. Since this relation is derived using only the commutator relation and the definition of the structure constants, it holds for all semi-simple compact Lie groups.

The second relation that the DDM basis builds on is the Kleiss-Kuijff relation[15], which is given by:

$$\mathcal{M}(1, \{\alpha\}, n, \{\beta\}) = (-1)^{n_\beta} \sum_{\sigma \in OP\{\alpha\}\{\beta^T\}} \mathcal{M}(1, \sigma(\{\alpha\}\{\beta^T\}), n) \quad (3.7)$$

Her the sum over $OP\{\alpha\}\{\beta^T\}$ stands for the sum over all ordered products of the two sets $\{\alpha\}$ and $\{\beta\}$, where an ordered product stands for any permutation that mixes the two sets and that preserves the relative ordering between the elements internally in each of the sets. This is also called the shuffle product.

This is also a calculation only depending on the kinematic part of the amplitude and is thus

not depending on the choice of gauge group. Inserting the Kleiss-Kuijff relation in eq. 3 yields:

$$\mathcal{M}_n = \sum_{(\{\alpha\}\{\beta\}) \in S_{n-1}} \text{tr}(T^{(1, \{\alpha\}, n, \{\beta\})}) \cdot (-1)^{n_\beta} \sum_{\sigma \in OP\{\alpha\}\{\beta^T\}} \mathcal{M}(1, \sigma(\{\alpha\}\{\beta^T\}), n) \quad (3.8)$$

Even though the two sets $\{\alpha\}$ and $\{\beta\}$ together contains $n-2$ elements, the first sum is still over the permutation of $n-1$ elements, since for every permutation of the two elements $\{\alpha\}$ and $\{\beta\}$ there are $n-1$ ways of setting the cut between the two sets. eg. all elements can be put in $\{\alpha\}$ and zero in $\{\beta\}$, or $n-3$ elements in $\{\alpha\}$ and 1 in $\{\beta\}$ and so forth until all elements are in the set $\{\beta\}$. Which then in total gives S_{n-1} ways of permuting and slicing the two sets.

The number of ordered permutations of the two sets $\{\alpha\}$ and $\{\beta\}$ is equal to the binomial coefficient of $\binom{\alpha+\beta}{\beta}$, when summing over all the cuts between the sets (how many elements each set gets), we get the sum:

$$\sum_{\beta=0}^{n-2} \binom{n-2}{\beta} = 2^{n-2} \quad (3.9)$$

where $n-2 = \alpha + \beta$. So using this summation of the elements on the two sums in eq. (3.8), we now have 2^{n-2} sub-amplitudes of the form $\sum_{\text{cuts}} \mathcal{M}(1, \sigma(\{\alpha\}\{\beta^T\}), n)$, and are left with a sum over the $(n-2)!$ elements of the permutations of the set $\{\alpha\} \cup \{\beta\}$. Since every permutation of the set $\{\alpha\} \cup \{\beta\}$ gives 2^{n-2} different sub-amplitudes, when summing over all the permutations we get for each of the $(n-2)!$ permutations, 2^{n-2} sub-amplitudes that are identical but have different traces associated with them. These traces exactly match the traces from the identity in eq. (3.4), so that we can combine the 2^{n-2} traces to a single term containing structure constants:

$$\mathcal{M}(1, 2, \dots, n) = \sum_{\pi \in S_{n-2}} f^{1\pi_2 x_1} f^{x_1 \pi_3 x_2} \dots f^{x_{n-3} \pi_{n-1} n} M(1, \pi_2, \pi_3, \dots, \pi_{n-1}, n) \quad (3.10)$$

Since this decomposition is achieved using only identities that does not depend on the Lie group, this holds for any of the 9 semi-simple compact Lie groups.

Zeppenfeld basis

What we achieved in section () is a method where all tree level calculations of scattering amplitudes of N particles are reduced to a sum over $N-2$ subamplitudes. This is a really nice result since it reduces the amount of calculations we have to do by a factor of $N(N-1)$. However since the calculations still depend on the gauge group of the generators it leaves us with no tangible way of comparing how the calculations for the different groups differ from each other. In order to compare how the group calculations differ from each other we need to find a way to disconnect the gauge group terms with the kinematic terms. A that allows us to do this was found in 1988 by Zeppenfeld [27] where he used the irreducible representations of the permutation group to reformulate calculations of scattering amplitudes into an orthogonal basis and this is the approach we will be using here.

Applying Zeppenfelds approach to the DDM basis is actually fairly straight forward and it can almost entirely be copied from his paper.

Defining colour factors of the form:

$$\Theta_{1n}^{(\alpha;kl)} = \frac{n_\alpha}{(n-2)!} \sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) T_{1n}^{(\pi)} \quad (4.1)$$

Where α labels the irreducible representations of the permutation group and n_α is the dimension of the corresponding representation. π is a permutation of the $n-2$ elements that is permuted in eq. (3.10):

$$T_{1n}^{(\pi)} = (F^{\pi_2} F^{\pi_3} \dots F^{\pi_{n-1}})_{1n} = f^{1\pi_2 x_1} f^{x_1 \pi_3 x_2} \dots f^{x_{n-3} \pi_{n-1} n} \quad (4.2)$$

where F_{ij}^1 is a generator in the adjoint representation. The elements $D_{kl}^{(\alpha)}(\pi)$ are a representation matrix for a specific irreducible orthogonal representation of the permutation group S_{n-2} . The new colour factors can be inverted using the orthogonal basis for the representation of the S_{n-2} elements, so this result is only realisable in that representation:

$$\begin{aligned} T_{1n}^{(\pi)} &= \sum_{\alpha} \sum_{k,l=1}^{n_\alpha} D_{kl}^{(\alpha)}(\pi) \Theta_{1n}^{(\alpha;kl)} \\ &= \sum_{\alpha} \sum_{k,l=1}^{n_\alpha} D_{kl}^{(\alpha)}(\pi) \sum_{\rho \in S_{n-2}} D_{kl}^{(\alpha)}(\rho) T_{1n}^{(\rho)} \\ &= \sum_{\rho \in S_{n-2}} \sum_{\alpha} \sum_{k,l=1}^{n_\alpha} \frac{n_\alpha}{(n-2)!} D_{kl}^{(\alpha)}(\pi) D_{kl}^{(\alpha)}(\rho) T_{1n}^{(\rho)} \\ &= \sum_{\rho \in S_{n-2}} \delta_{\pi\rho^{-1}} T_{1n}^{(\rho^{-1})} \\ &= T_{1n}^{(\pi)} \end{aligned}$$

Where the identity used between line 3 and 4 is [11]:

$$\sum_{\alpha} \sum_{k,l=1}^{n_\alpha} \frac{n_\alpha}{(n-2)!} D_{kl}^{(\alpha)}(\pi) D_{kl}^{(\alpha)}(\rho)^\dagger = \delta_{\pi\rho} \quad (4.3)$$

and:

$$D_{kl}^{(\alpha)}(\rho)^\dagger = D_{kl}^{(\alpha)}(\rho^{-1}) = D_{lk}^{(\alpha)}(\rho) \quad (4.4)$$

and:

$$n_\alpha^2 = (n-2)! \quad (4.5)$$

Which stems from the fact that we use the orthogonal representation of the symmetric group. multiplying two of the new colour factors together then gives:

$$\Theta_{1n}^{(\alpha;kl)}(\Theta_{1n}^{(\beta;ij)})^\dagger = \frac{n_\alpha}{(n-2)!} \sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) T_{1n}^{(\pi)} \left(\frac{n_\beta}{(n-2)!} \sum_{\rho \in S_{n-2}} D_{ij}^{(\beta)}(\rho) T_{1n}^{(\rho)} \right)^\dagger \quad (4.6)$$

$$= \frac{n_\alpha n_\beta}{((n-2)!)^2} \sum_{\pi, \rho \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) \left(D_{ij}^{(\beta)}(\rho) \right)^\dagger T_{1n}^{(\pi)} (T_{1n}^{(\rho)})^\dagger \quad (4.7)$$

$$= \frac{n_\alpha n_\beta}{((n-2)!)^2} \sum_{\pi, \rho \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) D_{ji}^{(\beta)}(\rho^{-1}) T_{1n}^{(\pi)} (T_{1n}^{(t\rho)}) \quad (4.8)$$

t is explained below (4.9)

$$= \frac{n_\alpha n_\beta}{((n-2)!)^2} \sum_{\pi, \rho \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) D_{jm}^{(\beta)}(t(t\rho^{-1}\pi)) D_{mi}^{(\beta)}(\pi^{-1}) \text{tr}(T^{(\pi)} T^{(t\rho)}) \quad (4.10)$$

$$= \frac{n_\alpha n_\beta}{((n-2)!)^2} \sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) D_{mi}^{(\beta)}(\pi^{-1}) \sum_{\rho \in S_{n-2}} D_{jm}^{(\beta)}(t(t\rho^{-1}\pi)) \text{tr}(T^{(\pi)} T^{(t\rho)}) \quad (4.11)$$

$$= \frac{n_\alpha n_\beta}{((n-2)!)^2} \sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) D_{mi}^{(\beta)}(\pi^{-1}) \sum_{\rho \in S_{n-2}} D_{jm}^{(\beta)}(t(t\rho^{-1}\pi)) \text{tr}(T^{(1)} T^{(t\rho\pi^{-1})}) \quad (4.12)$$

$$= \frac{n_\alpha n_\beta}{((n-2)!)^2} \sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) D_{mi}^{(\beta)}(\pi^{-1}) \sum_{\rho \in S_{n-2}} D_{jm}^{(\beta)}(t(t\rho^{-1}\pi)) \text{tr}(T^{(1)} T^{((t\rho^{-1}\pi)^{-1})}) \quad (4.13)$$

$$= \frac{n_\alpha n_\beta}{((n-2)!)^2} \frac{(n-2)!}{n_\alpha} \delta_{\alpha\beta} \delta_{ki} \delta_{lm} \sum_{\rho \in S_{n-2}} D_{jm}^{(\beta)}(t\rho) \text{tr}(T^{(1)} T^{(\rho^{-1})}) \quad (4.14)$$

$$= \frac{n_\alpha}{(n-2)!} \delta_{\alpha\beta} \delta_{ki} \sum_{\rho \in S_{n-2}} D_{jl}^{(\beta)}(t\rho) \text{tr}(T^{(\rho)} T^{(1)}) \quad (4.15)$$

between eq. (4.7) and eq. (4.8) the identities used are[11]:

$$\left(D_{ij}^{(\beta)}(\rho) \right)^\dagger = D_{ji}^{(\beta)}(\rho^{-1}) \quad \text{Same as in Zeppenfelds paper}$$

which is a feature of the irreducible representations matrices of S_{n-2} since we utilise them in their orthogonal form, and

$$\text{tr}(T^{(\rho)})^\dagger = \text{tr}(T^{(t\rho)})$$

where t represents a transposition such that the set $\{\sigma_1, \sigma_2, \dots, \sigma_n\}$ under t transforms into:

$$\{\sigma_1, \sigma_2, \dots, \sigma_n\} \rightarrow \{\sigma_n, \sigma_{n-1}, \dots, \sigma_1\} \quad (4.16)$$

between eq. (4.11) and eq. (4.12) the identity used is:

$$\text{tr}(T^\pi T^\rho) = \text{tr}(T^{\sigma\pi} T^{\sigma\rho})$$

This identity is the exact same as in Zeppenfeld's paper.
Between eq. (4.13) and eq. (4.14) the identity used is[11]:

$$\sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) D_{mi}^{(\beta)}(\pi^{-1}) = \frac{(n-2)!}{n_\alpha} \delta_{\alpha\beta} \delta_{ki} \delta_{ml} \quad (4.17)$$

which again is the same as in Zeppenfeld's paper.

Since we can express the colour factors $T_{1n}^{(\pi)}$ in terms of the new colour factors $\Theta_{1n}^{(\alpha;kl)}$ and irreducible representation matrices for S_{n-2} , we can express the amplitudes as:

$$\mathcal{M} = \sum_{\alpha} \sum_{k,l=1}^{n_\alpha} \Theta_{1n}^{(\alpha;kl)} \sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) \mathcal{M}^{(\pi)} \quad (4.18)$$

where the kinematic part $\mathcal{M}^{(\pi)}$ is:

$$\mathcal{M}^{(\pi)} = \mathcal{M}(1, \pi_2, \pi_3, \dots, \pi_{n-1}, n) \quad (4.19)$$

So then calculating the amplitude squared gives us:

$$\begin{aligned} |\mathcal{M}|^2 &= \sum_{\alpha} \sum_{k,l=1}^{n_\alpha} \Theta_{1n}^{(\alpha;kl)} \sum_{\pi \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) \mathcal{M}^{(\pi)} \left(\sum_{\beta} \sum_{i,j=1}^{n_\beta} \Theta_{1n}^{(\beta;ij)} \sum_{\rho \in S_{n-2}} D_{ij}^{(\beta)}(\rho) \mathcal{M}^{(\rho)} \right)^\dagger \\ &= \sum_{\alpha} \sum_{\beta} \sum_{k,l=1}^{n_\alpha} \sum_{i,j=1}^{n_\beta} \Theta_{1n}^{(\alpha;kl)} (\Theta_{1n}^{(\beta;ij)})^\dagger \sum_{\pi, \rho \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) (D_{ij}^{(\beta)}(\rho))^\dagger \mathcal{M}^{(\pi)} (\mathcal{M}^{(\rho)})^\dagger \\ &= \sum_{\alpha} \sum_{\beta} \sum_{k,l=1}^{n_\alpha} \sum_{i,j=1}^{n_\beta} \frac{n_\beta}{(n-2)!} \delta_{\alpha\beta} \delta_{ki} \delta_{lm} \sum_{\sigma \in S_{n-2}} D_{jm}^{(\beta)}(t\sigma) \text{tr}(T^{(\sigma)} T^{(1)}) \sum_{\pi, \rho \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) (D_{ij}^{(\beta)}(\rho))^\dagger \mathcal{M}^{(\pi)} (\mathcal{M}^{(\rho)})^\dagger \\ &= \sum_{\alpha} \sum_{k,l,j=1}^{n_\alpha} \frac{n_\alpha}{(n-2)!} \sum_{\sigma \in S_{n-2}} D_{jl}^{(\alpha)}(t\sigma) \text{tr}(T^{(\sigma)} T^{(1)}) \sum_{\pi, \rho \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) (D_{kj}^{(\alpha)}(\rho))^\dagger \mathcal{M}^{(\pi)} (\mathcal{M}^{(\rho)})^\dagger \\ &= \sum_{\alpha} \sum_{k,l,j=1}^{n_\alpha} C^{(\alpha;jl)} \sum_{\pi, \rho \in S_{n-2}} D_{kl}^{(\alpha)}(\pi) (D_{kj}^{(\alpha)}(\rho))^\dagger \mathcal{M}^{(\pi)} (\mathcal{M}^{(\rho)})^\dagger \end{aligned}$$

Where I have defined:

$$C^{(\alpha;jl)} = \frac{n_\alpha}{(n-2)!} \sum_{\sigma \in S_{n-2}} D_{jl}^{(\alpha)}(t\sigma) \text{tr}(T^{(\sigma)} T^{(1)}) \quad (4.20)$$

The factors $\text{tr}(T^{(\sigma)} T^{(1)})$ can then be calculated via their birdtrack diagrams:

$$1 \left(\begin{array}{c} \bullet \\ \text{---} \sigma_2 \text{---} \bullet \\ \text{---} \sigma_3 \text{---} \bullet \\ \text{---} \sigma_4 \text{---} \bullet \\ \text{---} \dots \text{---} \bullet \\ \text{---} \sigma_{n-1} \text{---} \bullet \\ \bullet \end{array} \right) n \quad (4.21)$$

Where the first and last vertical legs are fixed and the permutation is over the other vertical legs. These diagrams are thus only build up of structure constants, and can be calculated via the bird-track deconstruction of structure constants, which is given by (see section () for further explanation):

$$\text{Tree}(a,b,c) = \text{Loop}(a,b,c) - \text{Loop}(a,b,c)_{\text{self-energy}} \quad (4.22)$$

and by using the adjoint projection operator to expand each of the adjoint lines in the diagrams one ends up with.

Therefore for a diagram with n particles, we get $2(n-2)$ structure constants by eq. (4.21), and since each structure constant leads to a factor of 2 extra diagrams this will in total give $2^{2(n-2)}$ different diagrams to calculate. Furthermore for each of these $2^{2(n-2)}$ diagrams we have $n+2(n-3)$ adjoint projection operators (lines between the structure constants or), and if each of these projection operators contains 2 terms we further get another $2^{n+2(n-3)}$ diagrams. There may however be more than two terms in an adjoint projection operator, so that number will be denoted n_{po} . So in total for each factor of $\text{tr}(T^{(\sigma)}T^{(1)})$ which is given by eq. (4.21) we get $2^{2(n-2)}n_{po}^{n+2(n-3)}$ diagrams to calculate, and there are $(n-2)!$ different traces in an amplitude calculation, so in total when calculating an amplitude squared we need to calculate:

$$(n-2)!2^{2(n-2)}n_{po}^{n+2(n-3)} \quad (4.23)$$

different colour factors (or birdtrack diagrams). Comparing this to standard way of calculating scattering amplitudes from eq. (2.58) which we can draw in a way similar to eq. (4.21):

$$\text{Diagram with } \sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n \text{ between two horizontal lines} \quad (4.24)$$

we see that we in total have $(n!)^2$ diagrams, and each of these diagrams has n projection operators, which will then give n_{po}^n diagrams. From this we then in total get:

$$(n!)^2 n_{po}^n \quad (4.25)$$

different diagrams. So we see that the difference in the number of diagrams we need to calculate is eq. (4.23) divided by eq. (4.25) which gives:

$$\frac{2^{2(n-2)}n_{po}^{2(n-3)}}{n(n-1)(n!)} \quad (4.26)$$

For values of n larger than 4 and smaller than 74 gives a number larger than 1, which means that in that range it is more computationally heavy to do calculations this way since there are more terms

to calculate. On the other hand though, by calculating amplitudes using the Zeppenfeld basis the amount of kinematic factors we have to calculate has been reduced from $(n!)^2$ to $((n-2)!)^2$, which is often preferred since the kinematic factors can be very hard to calculate, and the colour factors can essentially just be calculated using simple algorithms (once those algorithms are found, and if the number of particles is too big, it might take a very long time and a lot of RAM to calculate all those diagrams though).

Birdtrack

It is often said that one of the biggest hurdles when solving a problem in physics is to find a notation to express the problem in a solvable way. One of the best examples of this is the Lagrangian formalism in classical mechanics, which is essentially just a rewriting of Newton's equations but which offers a whole new way of framing problems in terms of energies instead of forces.

In this section we will do something similar for group theory calculations, by using what is commonly known as *Birdtrack*. Birdtracks is essentially a pictorial way of solving various group calculations by drawing diagrams and summing up the result based on what rules the group has. The method itself is equivalent to writing up tensors and expanding them according to normal convention, but as shall hopefully be clear throughout the rest of this thesis, it will often be easier to get an overview of the problem by formalising it in Birdtracks. This section is based upon the book "Group Theory: Birdtracks, Lie's and Exceptional Groups" [3], written by Predrag Cvitanović.

Definitions

In the Birdtrack notation a propagator that connects two vertices is a Kronecker delta given by:

$$\delta^a_b = a \longrightarrow b \quad (5.1)$$

The arrows give a pictorial way of distinguishing upper and lower indices, where arrows always point away from the upper index and towards the lower index. This way of representing indices is only relevant when a representation of the group is complex, for real representations there is no difference between upper and lower indices. Any invariant tensor can in this formalism be realised as a vertex:

$$A_\alpha = A^{abc}_{de} = \begin{array}{c} a \\ b \\ c \\ d \\ e \end{array} \begin{array}{c} \longrightarrow \\ \longrightarrow \\ \longrightarrow \\ \longleftarrow \\ \longleftarrow \end{array} \boxed{A_\alpha} \quad (5.2)$$

where it does not matter if the vertex is drawn as a square, circle or simply a dot, however it is preferable if different types of objects and invariant tensors get their own vertex shape/colour to make it easier to distinguish them. The upper and lower indices are read separately so that their relative order does not matter, however it is important to note that the order of the indices in the

same row is to be upheld. The order of reading the indices is counterclockwise:

$$A^{ab}_{cde} = \text{Diagram of tensor A with inputs a, c, d and outputs b, e, and a counterclockwise arrow} \tag{5.3}$$

The above diagram is drawn in such a way that it can be interpreted as that the indices are cyclically invariant, however if this is not the case it should be clearly visible by the drawing of the diagram, as stated below:

$$A^{ab}_{cde} = \text{Diagram of tensor A with inputs c, a, d and outputs b, e, and a counterclockwise arrow} \tag{5.4}$$

Summing over indices is drawn as an internal line between two tensors or when two indices is summed over in the same tensor it is drawn as a line from the tensor to the tensor itself:

$$A^{ac}_{bd} B^{bd}_{ce} = \text{Diagram of tensors A and B with contraction lines} \quad A^{ab}_{bd} = \text{Diagram of tensor A with self-contraction} \tag{5.5}$$

A matrix that has both upper and lower indices is defined as an object that transforms transforms objects in the following way $\mathbf{M} = V^p \otimes \bar{V}^q \rightarrow V^p \otimes \bar{V}^q$, is in the Birdtrack notation given by:

$$M = \text{Diagram of a rectangular box M with multiple input and output lines} \tag{5.6}$$

The matrix multiplication is then defined just like the contraction of indices for tensors, just with the rule that if two matrices are multiplied the only indices that can be summed over is those that are on the sides that is oriented towards the other matrix. Notice that all the names on the lines have been dropped. This is due to the fact that we do not need them in order to know what is going on, the other rules we set up already completely describes everything we needs to know, so giving them a name will just be extra work for us without any benefit.

Also due to the way that we have depicted a matrix it is possible for us to draw the trace of a

matrix in a planar way:

$$\text{tr}(M) = \begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \vdots \\ \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \end{array} \text{M} \begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \vdots \\ \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \end{array} \quad (5.7)$$

Symmetric and anti-symmetric tensors

One of the most useful notations in birdtracks is the symmetrization or anti-symmetrization of tensor indices. Permuting of indices is a linear operation, and we can thus represent it by a matrix. We will here only consider completely symmetric or anti-symmetric tensors, and not matrices that are a mix of those. Lets start wit symmetrization.

If we have a 2 index tensor we have two options when permuting indices, these are given by:

$$\text{Identity: } \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \text{Flip: } \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad (5.8)$$

So when we symmetries a 2 index tensor we have to apply both of these operations and sum them up, then we have to divide with 2 since we are summing up two elementsw and to make sure that a symmetrization of a symmetrization does not change eh anything about the tensor. In the general case with a n-index tensor the action of symmetrizing p indices is thus given by a sum over all the $p!$ ways of permuting p elements:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} = \frac{1}{p!} \left\{ \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} + \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} + \dots + \text{all } p! \text{ permutations} \right\} \quad \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} \quad (5.9)$$

also a symmetrization of a subset of symmetrized indices also just returns the original symmetrized indices:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} \quad (5.10)$$

This also means that any permutation on the symmetrized indices just gives the indices back, thus any permutation has eigenvalue 1 on the symmetric space. Since calculations with these symmetrizations yields $p!$ diagrams they are often very tedious to work with, therefore it is preferable to work out identities to lessen our work. The first identity that we shall state is the reduction of one symmetrized index given by:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} = \frac{1}{p} \left\{ \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} + \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} + \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} + \dots \right\} = \frac{1}{p} \left\{ \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} + (p-1) \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \end{array} \right\} \quad (5.11)$$

Compared to the symmetric case where crossing two legs on a space just returns the original space, crossing two adjacent legs on an anti-symmetric space returns the same space multiplied with a factor of -1 . Again we continue as we did with the symmetric space and see if we can find a relation that eases up calculations with anti-symmetric subspaces so that we do not have to deal with $p!$ diagrams. This gives us:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} = \frac{1}{p} \left\{ \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} - \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} + \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} - \dots \right\} = \frac{1}{p} \left\{ \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} - (p-1) \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} \quad (5.17)$$

Tracing this identity we get:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} = \frac{1}{p} \left\{ \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} - (p-1) \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} = \frac{n-p+1}{p} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \quad (5.18)$$

and with repeated use of this formula we can again get formulas for tracing $(p-k)$ or simply all indices:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} = \frac{(n-k)!k!}{p!(n-p)!} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} = \frac{(n)!}{p!(n-p)!} \quad (5.19)$$

It is worth noticing that the identities for the anti-symmetric calculations are only valid if the number of indices to be anti-symmetrized is smaller than the dimension of the representation, a result well known from dealing with tensors. Since Birdtrack notation is just another way of representing tensors that result naturally also has to be realised here.

Invariance and projection operators

Lets start off by going back to the matrix and tensor notation, and define an invariant tensor as an object that satisfies:

$$x_{b_1 \dots b_q}^{a_1 \dots a_p} = G_{c_1}^{a_1} \dots G_{c_p}^{a_p} G_{b_1}^{d_1} \dots G_{b_q}^{d_q} x_{d_1 \dots d_q}^{c_1 \dots c_p} \quad (5.20)$$

for any group element g in \mathcal{G} , where G is a matrix representation of g . If a bilinear form, $m = x^a M_a^b y_b$ $\mathbf{M} : V^p \otimes \bar{V}^q \rightarrow V^p \otimes \bar{V}^q$, is invariant for all g , then the matrix M is an invariant matrix:

$$M_a^b = G_a^c G_d^b M_c^d \quad (5.21)$$

If M is unitary we can multiply it with G_b^e and thus find that invariant matrices commutes with all transformations:

$$[G, M] = 0 \quad (5.22)$$

If all indices on an invariant tensor is traced with invariant vectors the resulting scalar will be an invariant scalar, this is given by:

$$h = h_{cde}^{ab} x_b y_a s^e r^d z^c \quad (5.23)$$

and is represented with:

$$\begin{array}{c}
 \text{---} 1 \quad \left\{ \begin{array}{l} \text{---} b_1 \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} a_s \end{array} \right. \\
 \end{array} \quad (5.35)$$

They map the space $V \rightarrow V_1$. The hermitian conjugate of a Clebsch-Gordon coefficient is naturally given by:

$$\begin{array}{c}
 \left. \begin{array}{l} \text{---} b_1 \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} a_s \end{array} \right\} \text{---} 1 \\
 \end{array} \quad (5.36)$$

From the definition of projection operators we can express them in terms of these clebsch-gordon coefficients:

$$P_1 = C^1 C_1 = \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \left\{ \begin{array}{l} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} 1 \left. \begin{array}{l} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} P_1 \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \quad (5.37)$$

a specific projection was chosen here to emphasise that there is no sum over the contracted indices for the Clebsch-Gordon coefficients. The Clebsch-Gordon coefficients are orthonormal so that they satisfy:

$$\begin{array}{c} \mu \end{array} \left\{ \begin{array}{l} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} \left. \begin{array}{l} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} \begin{array}{c} \nu \end{array} = \delta_{\mu}^{\nu} \quad (5.38)$$

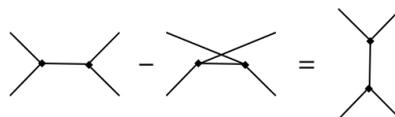
for two subspaces μ and ν . The completeness relation for the projection operators given by eq. (5.29), can be expressed as:

$$\sum_{\lambda} \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \left\{ \begin{array}{l} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} \lambda \left. \begin{array}{l} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \right\} \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ \vdots \\ \text{---} \end{array} \quad (5.39)$$

As a side note lets look back at eq. (5.19) where we see that an anti-symmetric tensor with n indices only has one independent component $d_n = 1$. The Clebsches for this case is proportional to the Levi-Civita tensor, which we pictorially draw as:

$$\begin{array}{c} C\epsilon^{a_n \dots a_2 a_1} = \begin{array}{c} \text{---} a_1 \\ \text{---} a_2 \\ \vdots \\ \text{---} \end{array} \\ \\ C\epsilon_{a_1 a_2 \dots a_n} = \begin{array}{c} a_1 \text{---} \\ a_2 \text{---} \\ \vdots \\ a_n \text{---} \end{array} \end{array} \quad (5.40)$$

The structure constants also satisfies the commutator relation, see eq. (1.10), which for the adjoint representation is called the Jacobi relation, which we can draw as:


(5.62)

Looking at eq. (1.17) we can also draw the pictorial diagram for the quadratic Casimir invariant. This is easily realised as:


(5.63)

So given all this what is it then that we want to achieve? Well essentially the goal of this paper is to calculate the colour factors defined in eq. (4.21), that are part of the c factors defined in eq. (4.20). The colour factors consists of structure constants where all indices are summed over, and these structure constants can all be expanded using the trace based relation defined in eq. (5.60). Then what we end up with is $2^{2(n-2)}$ diagrams where n is the number of particles in the scattering amplitude. Each of these diagrams can then be further decomposed by using the adjoint projection operator. To determine the adjoint projection operator we follow the procedure from this chapter and start by decomposing the $V \otimes \bar{V}$ space into irreducible subspaces by building these subspaces up from tree invariant tensors that are each composed of primitive invariants. Then lastly we can single out the adjoint space since it is the only one that annihilates invariant tensors. This method will take us very far, but as we shall later see we will actually need to develop other tricks in order to tackle some of the exceptional cases. Therefore what we will do now is dive in to each of the eight simple compact Lie groups and treat them separately.

The four families

Now with the method outlined in the last part of section , we are finally ready to begin our calculations of the colour factors described in eq. (4.21). In this chapter we will derive the method of how to calculate these colour factors for the four families: A_n , B_n , C_n and D_n . The notation we so far have used for the families of Lie groups is the one mathematicians use, however now we will switch to the notation most commonly used by physicists. In this notation we get that $A_n = su(n+1)$, $B_n = so(2n+1)$, $C_n = sp(2n)$ and $D_n = so(2n)$. The groups B_n and D_n both describes the special orthogonal group in either odd or even dimensions. The reason that they are split up is that on a fundamental level there is a difference in the way their Dynkin are drawn and thus a difference in their root systems, however since both groups preserve the same invariant tensor we will see that for our purposes we do not need to split up the algebra of $so(n)$ into even and odd dimensions, but simply treat it as one algebra. With this in mind let's get started.

$SU(N)$

The group $SU(n)$ is defined as the invariance group of the Kronecker delta δ_b^a and the Levi-Civita tensor $\epsilon^{a_1 a_2 \dots a_n}$, and our goal is to use these to find the adjoint projection operator, defined and

However if we also want the Levi-Civita symbol, defined in eq. (5.40), to be an invariant of the group, we need to find the projection operator that satisfies the invariant condition for the completely anti-symmetric Levi-Civita tensor:

$$\begin{array}{c} \curvearrowright \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \end{array} = 0 \tag{6.6}$$

Inserting each of the two projection operators into this condition we see that it is only P_2 , that satisfies the invariance condition. This can be seen by using eq. (5.17) to expand the expression from eq. (5.43), which gives:

$$\begin{aligned} 0 &= \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} = \frac{1}{n+1} \left\{ \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} - (n+1-1) \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} \right\} \\ &= \frac{1}{n+1} \left\{ \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} - n \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} \right\} \\ &\Rightarrow \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} = \frac{1}{n+1} \left\{ \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} - n \begin{array}{c} \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ \downarrow \downarrow \downarrow \dots \downarrow \\ \text{---} \\ 1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n+1 \end{array} \right\} = 0 \end{aligned} \tag{6.7}$$

This we see is equal to inserting P_2 into eq. (6.6). So finally we find that the adjoint projection operator for $su(n)$ is given by:

$$P_{A_{su(n)}} = \begin{array}{c} \overrightarrow{\quad} \\ \overleftarrow{\quad} \end{array} - \frac{1}{n} \begin{array}{c} \curvearrowright \\ \curvearrowleft \end{array} \tag{6.8}$$

and taking the trace of the adjoint projection operator, we get the dimension of the adjoint representation:

$$d_A = \text{tr}(P_{A_{su(n)}}) = \begin{array}{c} \overrightarrow{\quad} \\ \overleftarrow{\quad} \end{array} - \frac{1}{n} \begin{array}{c} \curvearrowright \\ \curvearrowleft \end{array} = n^2 - 1 \tag{6.9}$$

So what to do now? Well first of all we need to make sure that our method of calculation is correct. Luckily we actually know a lot about the $su(n)$ algebra from other sources, so our priority right now is to make sure we know how to calculate diagrams using the adjoint projection operators, then later on we will return to the question of if what we do gives the correct answer. So going boldly into the night we would now like to calculate the colour factor for 3 particle scattering which

is given by the diagram:

$$f^{abc} f_{acb} = \left(\text{diagram of a vertical line with two arcs connecting the ends} \right) \quad (6.10)$$

To calculate this diagram we start by expanding it using eq. (5.60) and collection equal terms, so that we end up with:

$$\left(\text{diagram of a vertical line with two arcs} \right) = 2 \left\{ \left(\text{diagram of two circles connected by a vertical line} \right) - \left(\text{diagram of two circles connected by two crossing lines} \right) \right\} \quad (6.11)$$

The two terms in this expression are straightforward to calculate and the calculation is given by:

$$\begin{aligned} \left(\text{diagram of two circles connected by a vertical line} \right) &= \left(\text{diagram of two circles with internal arcs} \right) - \frac{1}{n} \left(\text{diagram of two circles with internal arcs} \right) - \frac{1}{n} \left(\text{diagram of two circles with internal arcs} \right) + \frac{1}{n^2} \left(\text{diagram of two circles with internal arcs} \right) \\ &= \left(n - \frac{2}{n} \right) \left(\text{diagram of a single circle} \right) + \frac{1}{n^2} \left(\text{diagram of two circles with internal arcs} \right) = \left(n - \frac{2}{n} \right) \left(\text{diagram of a single circle} \right) \end{aligned} \quad (6.12)$$

$$\begin{aligned} \left(\text{diagram of two circles connected by two crossing lines} \right) &= \left(\text{diagram of two circles connected by two crossing lines} \right) = \left(\text{diagram of two circles with internal arcs} \right) - \frac{1}{n} \left(\text{diagram of two circles with internal arcs} \right) - \frac{1}{n} \left(\text{diagram of two circles with internal arcs} \right) + \frac{1}{n^2} \left(\text{diagram of two circles with internal arcs} \right) \\ &= \left(1 + \frac{1}{n^2} \right) \left(\text{diagram of two circles with internal arcs} \right) - \frac{2}{n} \left(\text{diagram of a single circle} \right) = -\frac{2}{n} \left(\text{diagram of a single circle} \right) \end{aligned} \quad (6.13)$$

putting these two terms back into eq. (6.11) we thus get:

$$2 \left\{ \left(n - \frac{2}{n} \right) \left(\text{diagram of a single circle} \right) - \frac{2}{n} \left(\text{diagram of a single circle} \right) \right\} = 2n \left(\text{diagram of a single circle} \right) = 2(n^3 - n) \quad (6.14)$$

Now we have seen the first example of how to calculate one of the colour factors from eq. 4.21. Even though what we calculated was the simplest case we could come up with, our method is still general enough that any possible diagram no matter the amount of particles will be possible for us to calculate. Therefore we can confidently say now that we have solved our first group $su(n)$. Even though this group has already been solved in the general literature it is still nice to see that the Birdtrack method is able to reproduce the results we already know.

$SO(N)$

With our recent success with $su(n)$ we now move on to tackle the case of calculating colour factors for the special orthogonal group $SO(n)$. Transformations in $SO(n)$ preserve a symmetric rank 2 non-degenerate tensor given by g^{ab} , where we define contractions with its inverse by:

$$g^{ab}g_{bc} = \delta_c^a \quad (6.15)$$

where g_{ab} is the inverse of g^{ab} . We call the invariant for the metric of the group, and with the metric we can raise and lower indices at will on any tensor. Since we raise and lower indices, we do no longer distinguish between upper and lower indices and we can therefore drop arrows on our lines. The metric will therefore have the following birdtrack representation:

$$g^{ab} = a \text{-----} b \quad (6.16)$$

Since we can raise and lower indices, we can now put both indices in the generators $(T^i)_b^a$ on equal footing, which gives:

$$(T^i)_b^a g_{ac} = g_{ac} (T^i)_b^a = (T^i)_{bc} \quad (6.17)$$

The invariance condition for the metric tensor is then given by:

$$0 = (T^i)_b^a g_{ac} + g_{ab} (T^i)_c^a \quad (6.18)$$

$$0 = (T^i)_{bc} + (T^i)_{cb} \quad (6.19)$$

$$(T^i)_{bc} = -(T^i)_{cb} \quad (6.20)$$

This can be drawn diagrammatically in birdtrack notation as:

$$\begin{array}{c} | \\ \text{b} \text{-----} \text{c} \end{array} + \begin{array}{c} | \\ \text{b} \text{-----} \text{c} \end{array} = 0 \quad (6.21)$$

Which shows us that the $so(n)$ generators are anti-symmetric. Since the metric tensor does not carry any indices, we have a little more room to play with when decomposing the $V \otimes V$ space. The total amount of spaces that we can make is given by:

$$\begin{array}{ll} \text{Identity:} & \text{---} \\ & \text{---} \\ \text{flip:} & \text{X} \\ \text{trace:} & \text{) (} \end{array} \quad (6.22)$$

Using the flip space as our decomposition space we insert it into eq. (5.26), and find that it satisfies the following characteristic equation:

$$\begin{aligned} & \text{Diagram: } \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} = \text{Diagram: } \begin{array}{c} \text{---} \\ \text{---} \end{array} \\ \Rightarrow & \left(\begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} - \begin{array}{c} \text{---} \\ \text{---} \end{array} \right) \left(\begin{array}{c} \diagdown \quad \diagup \\ \diagdown \quad \diagup \end{array} + \begin{array}{c} \text{---} \\ \text{---} \end{array} \right) = 0 \end{aligned} \quad (6.23)$$

From this we see that the eigenvalues are $\lambda_1 = 1$ and $\lambda_2 = -1$. We find the projection operators by inserting these into the definition of the projection operators given by eq. (5.27), which gives us:

$$\begin{aligned} P_1 &= \begin{array}{c} \text{---} \\ \text{---} \end{array} + \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ | \\ \text{---} \\ \text{---} \end{array} \\ P_2 &= \begin{array}{c} \text{---} \\ \text{---} \end{array} - \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ \blacksquare \\ \text{---} \\ \text{---} \end{array} \end{aligned} \quad (6.24)$$

By direct calculation we see that P_1 does not annihilate the trace subspace, but P_2 does annihilate this subspace:

$$\begin{aligned} \begin{array}{c} | \\ \text{---} \\ \text{---} \end{array} \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} &= \frac{1}{2} \left(\text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagdown \quad \diagup \end{array} + \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} \right) = \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagdown \quad \diagup \end{array} \\ \begin{array}{c} \blacksquare \\ \text{---} \\ \text{---} \end{array} \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} &= \frac{1}{2} \left(\text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagdown \quad \diagup \end{array} - \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} \right) = 0 \end{aligned} \quad (6.25)$$

Now since P_1 does not annihilate the trace subspace we can know that we need to further decompose the space. To do that we take the trace subspace and calculate the characteristic equation. From there we get the eigenvalues which we put into the definition of the projection operators, only this time the element that we treat as the identity is now the symmetric subspace. Since the result for the characteristic equation will give us 2 eigenvalues (just like for $su(n)$), we can further split the symmetric subspace up into two new subspaces and get their projection operators, however there is actually no reason at all for us to do that because we see that by inserting the anti-symmetric projection operator P_2 into the invariance condition for the metric tensor, eq. (6.21), we find that:

$$\begin{aligned} & \text{Diagram: } \begin{array}{c} \blacksquare \\ \text{---} \\ \text{---} \end{array} + \text{Diagram: } \begin{array}{c} \blacksquare \\ \text{---} \\ \text{---} \end{array} = 0 \\ \Rightarrow & \frac{1}{2} \left(\begin{array}{c} \text{---} \\ \text{---} \end{array} \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagdown \quad \diagup \end{array} - \begin{array}{c} \text{---} \\ \text{---} \end{array} \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} + \begin{array}{c} \text{---} \\ \text{---} \end{array} \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} - \begin{array}{c} \text{---} \\ \text{---} \end{array} \text{Diagram: } \begin{array}{c} \diagdown \quad \diagup \\ \diagdown \quad \diagup \end{array} \right) = 0 \end{aligned} \quad (6.26)$$

And since it is only the generators that satisfies this condition, we find that the anti-symmetric projection operator is the adjoint projection operator:

$$\text{)---(} = \text{---} \quad (6.27)$$

The dimension of the adjoint space can then be found by tracing its projection operator which gives:

$$d_{A_{so(n)}} = \text{tr}(P_{A_{so(n)}}) = \text{---} = \text{---} = \frac{1}{2} \left\{ \text{---} - \text{---} \right\} = \frac{n(n-1)}{2} \quad (6.28)$$

Which is a wellknow result.

This means that we are done with exploring $so(n)$, since we do not need any more from that group in order to calculate the colour factors from eq. (4.21). As we did with $su(n)$, we would like to do a calculation in order to verify if we have found the correct projection operator. To do this we again want to calculate the colour factor for a 3 particle scattering amplitude. When doing that we can reuse some of the results from the $su(n)$ chapter and start directly from eq. (6.11). Doing this we find that the first diagram in the expression gives:

$$\begin{aligned} \text{---} &= \frac{1}{4} \left\{ \text{---} - \text{---} - \text{---} + \text{---} \right\} \\ &= \frac{(n-2)}{4} \text{---} \end{aligned} \quad (6.29)$$

The next diagram gives:

$$\begin{aligned} \text{---} &= \text{---} = \frac{1}{4} \left\{ \text{---} - \text{---} - \text{---} + \text{---} \right\} \\ &= \frac{1}{4} \left\{ \text{---} + (2-n) \text{---} \right\} = \frac{(2-n)}{4} \text{---} \end{aligned} \quad (6.30)$$

So inserting both of these expressions in eq. (6.11) we get:

$$2 \left\{ \frac{(n-2)}{4} \bigcirc - \frac{(2-n)}{4} \bigcirc \right\} = (n-2) \bigcirc = \frac{n(n-1)(n-2)}{2} \quad (6.31)$$

Again what we have done here is that we have calculated the simplest colour factor that we can encounter from eq. (4.21), but the method with which we have calculated this is general enough that every colour factor we can encounter can be calculated from it. So with the success of $so(n)$ lets move on to tackle the group $sp(n)$.

$SP(N)$

The symplectic group $sp(n)$ is defined as the group that leaves a skew-symmetric tensor f^{ab} invariant. That the tensor is skew symmetric, means that:

$$f^{ab} = -f^{ba} \quad (6.32)$$

The skew symmetric tensor satisfies the following contraction identity:

$$f^{ab} f_{bc} = \delta_c^a \quad (6.33)$$

Where the tensor f_{ab} is the inverse of f^{ab} . The skew symmetric tensor acts as a metric on the symplectic space, and makes it possible for us to lower and raise indices at will. Because of this we will, just like with $so(n)$, remove all arrows on the lines representing the primitive invariant. However since the primitive invariant in this case satisfies eq. (6.32), we have to keep track of the sign when contracting indices. Therefore we will draw the primitive invariant in the following way:

$$\begin{aligned} f^{ab} &= a \overline{\longleftarrow} b \\ f^{ba} &= a \overline{\longrightarrow} b \end{aligned} \quad (6.34)$$

Where it no longer matter if the indices are up or down. The reason we draw the lines with a triangle pointing either up or down is to keep track of the relative ordering of the indices in f^{ab} , since interchanging them per. (6.32) changes sign when interchanging a and b . Pictorially we can think of this as rotating the line in eq. 6.34 180 degrees, this still allows us to connect the line as before we rotated it, but because we then have to interchange the indices we no have to pick up a sign. The arrow therefore helps us keep track of this sign. The sign convention for the arrows then follow from eq. (6.33), and gives:

$$\overline{\longleftarrow} \overline{\longrightarrow} = - \overline{\longleftarrow} \overline{\longleftarrow} = \overline{\longrightarrow} \overline{\longrightarrow} \quad (6.35)$$

The invariance condition for the skew symmetric invariant is then:

$$0 = (T^i)_b^a f_{ac} + f_{ab} (T^i)_c^a \quad (6.36)$$

$$0 = (T^i)_{bc} - (T^i)_{cb} \quad (6.37)$$

$$(6.38)$$

where we draw the generators as:

$$(T^i)_b^a f_{ac} = \begin{array}{c} | \\ \hline \blacktriangleright \end{array} \quad (6.39)$$

The invariance condition for the skew symmetric tensor then becomes:

$$\begin{array}{c} | \\ \hline \blacktriangleright \end{array} + \begin{array}{c} | \\ \circlearrowleft \\ \hline \blacktriangleright \end{array} = 0 \quad (6.40)$$

Now proceeding as for the group $so(n)$: since we do not have any arrows on our invariant we can draw all possible combinations for the space $V \otimes V$ as:

$$\begin{array}{ll} \text{Identity:} & \begin{array}{c} \hline \blacktriangleright \\ \hline \blacktriangleright \end{array} \\ \text{flip:} & \begin{array}{c} \diagdown \diagup \\ \diagup \diagdown \end{array} \\ \text{trace:} & \begin{array}{c} \curvearrowright \quad \curvearrowleft \end{array} \end{array} \quad (6.41)$$

Using the flip space as our decomposition space we insert it into eq. (5.26), and find that it satisfies the following characteristic equation:

$$\begin{array}{c} \diagdown \diagup \\ \diagup \diagdown \end{array} = \begin{array}{c} \hline \hline \end{array} \\ \Rightarrow \left(\begin{array}{c} \diagdown \diagup \\ \diagup \diagdown \end{array} - \begin{array}{c} \hline \hline \end{array} \right) \left(\begin{array}{c} \diagdown \diagup \\ \diagup \diagdown \end{array} + \begin{array}{c} \hline \hline \end{array} \right) = 0 \end{array} \quad (6.42)$$

From this we get the exact same eigenvalues as with the $so(n)$ case, namely $\lambda_1 = 1$ and $\lambda_2 = -1$. We again insert this into the definition of the projection operators eq. (5.27) and find the two projection operators to be:

$$\begin{array}{l} P_1 = \begin{array}{c} \hline \hline \end{array} + \begin{array}{c} \diagdown \diagup \\ \diagup \diagdown \end{array} = \begin{array}{c} | \\ \hline \hline \end{array} \\ P_2 = \begin{array}{c} \hline \hline \end{array} - \begin{array}{c} \diagdown \diagup \\ \diagup \diagdown \end{array} = \begin{array}{c} \blacksquare \\ \hline \hline \end{array} \end{array} \quad (6.43)$$

The next thing we would like to do is, just as for the case of $so(n)$, to check if these two projection operators annihilate the trace subspace. This time however it is the symmetric subspace that annihilates the trace and not the anti-symmetric. This is caused by the skew symmetric tensor that makes the trace subspace anti-symmetric. going through the case of decomposing that space

further we actually ends up with the exact same result as with $so(n)$, just with anti-symmetrization exchanged with symmetrization. Finding the adjoint projection operator now is not very hard, we simply insert the projection operators into the eq. (6.40) and see if any of them satisfies it. Luckily we do not have to look far. Inserting the symmetric projection operator we find that:

$$\begin{aligned}
 & \text{---} \downarrow + \text{---} \uparrow = 0 \\
 \Rightarrow & \frac{1}{2} \left(\text{---} \downarrow \downarrow + \text{---} \downarrow \uparrow + \text{---} \uparrow \downarrow + \text{---} \uparrow \uparrow \right) \\
 & = \frac{1}{2} \left(\text{---} \downarrow \downarrow + \text{---} \downarrow \uparrow - \text{---} \uparrow \downarrow - \text{---} \uparrow \uparrow \right) = 0
 \end{aligned} \tag{6.44}$$

where we have used eq. (6.34). This then tells us that the adjoint projection operator for $sp(n)$ is given by:

$$P_{A_{sp(n)}} = \text{---} \uparrow \downarrow \tag{6.45}$$

The dimension of the adjoint space for $sp(n)$ is then given by the now familiar expression:

$$d_{A_{sp(n)}} = \text{tr}(P_{A_{sp(n)}}) = \text{---} \uparrow \downarrow = \frac{1}{2} \left\{ \text{---} \uparrow \downarrow + \text{---} \downarrow \uparrow \right\} = \frac{n(n+1)}{2} \tag{6.46}$$

Which is all we need to know in order for us to calculate anything we can encounter with regards to gauge group colour factors for $sp(n)$. Now we would like to return to our usual check and see what the result is when we calculate the diagram from (6.11). First we need to exchange every generator with the "new" generators from eq. (6.39), and then we can expand the with the adjoint projection operator from eq. (6.45), where we make sure to keep track of the sign using eq. (6.35), so that we for the first diagram get:

$$\begin{aligned}
 & \text{---} \uparrow \downarrow = \frac{1}{4} \left\{ \text{---} \uparrow \downarrow \uparrow \downarrow + \text{---} \uparrow \downarrow \downarrow \uparrow + \text{---} \downarrow \uparrow \uparrow \downarrow + \text{---} \downarrow \uparrow \downarrow \uparrow \right\} \\
 & = \frac{(n+2)}{4} \text{---} \uparrow \downarrow + \frac{1}{4} \text{---} \uparrow \downarrow = -\frac{n(n+1)(n+2)}{4 \cdot 2}
 \end{aligned} \tag{6.47}$$

and the second terms yields:

$$\begin{aligned}
 \begin{array}{c} \text{diagram 1} \\ \text{diagram 2} \end{array} &= \frac{1}{4} \left\{ \begin{array}{c} \text{diagram 3} \\ \text{diagram 4} \\ \text{diagram 5} \\ \text{diagram 6} \end{array} \right\} \\
 &= \frac{1}{4} \left\{ \begin{array}{c} \text{diagram 7} \\ \text{diagram 8} \end{array} \right\} + (n+2) \begin{array}{c} \text{diagram 9} \\ \text{diagram 10} \end{array} = \frac{(n+2)(n+1)n}{4 \cdot 2}
 \end{aligned} \tag{6.48}$$

So in total when inserting these terms back into eq. (6.11), we get:

$$f^{abc} f_{acb} = 2 \left(-\frac{n(n+1)(n+2)}{4 \cdot 2} - \frac{n(n+1)(n+2)}{4 \cdot 2} \right) = -\frac{n(n+1)(n+2)}{2} \tag{6.49}$$

The method used here to calculate colour factors for $sp(n)$ is a bit cumbersome since it requires one to keep a close eye on the sign of the whole thing. So to calculate factors of $sp(n)$ we will from now on use another method described by: [3] namely that calculations done with $sp(n)$ are equal to calculations done with $so(n)$ provided that we make the exchanging $n \rightarrow -n$. Indeed when looking at eq. (6.31) we see that if we make the previous exchanging we recover the result from eq. (6.49).

Results and verification of correctness

Now that we know how to calculate diagrams for each of the families, we would like to check if our method of calculation is correct. To do this we need a unique scalar for the group that we can calculate, and one that we already from know literature. The number we want to calculate for each group is the dual Coxeter number h^\vee defined in [8], and reproduced in figure (6.1):

group	h^\vee
su(n)	n
so(n)	$n - 2$
sp(n)	$(n + 2) / 2$
G_2	4
F_4	9
E_6	12
E_7	18
E_8	30

Figure 6.1: A table of the dual coxeter numbers for for the simple-compact lie groups

The dual coxeter number is equal to half the Casimir number of the adjoint representation, the

adjoint casimir is defined in [5] and its relation with the dual coxeter number is also outlined.

$$f^{abc} f_{cbd} = (T^i)_c^a (T_i)_d^c = \begin{array}{c} a \longrightarrow \bigcirc \longrightarrow d \\ \text{---} \end{array} = 2\check{h} \begin{array}{c} a \longrightarrow d \\ \text{---} \end{array} \quad (6.50)$$

This is the same as in eq. (5.63), where the representation is the adjoint representation. This expression is also almost the same as our 3 particle calculation, where the only difference is that here we have not closed the last line. If we close it, it will give us an extra factor of the dimension of the adjoint space. So to check it we simply take the results we got for the 3 particle scattering and divide it by the dimension of the corresponding adjoint space, to which we get:

$$\begin{aligned} su(n) &= 2n = 2h_{su(n)}^v \\ so(n) &= n - 2 = h_{so(n)}^v \\ sp(n) &= n + 2 = 2h_{sp(n)}^v \end{aligned} \quad (6.51)$$

From this we see that $so(n)$ differs by a factor of two compared to the two other groups. To see why this does not match the case outlined in eq. (6.50), we have to look at how the invariant quadratic form of the Lie algebra is normalized. The normalization convention chosen in the mathematics literature is to set the length squared of the highest root in the Cartan algebra equal to 2, which for $su(n)$ and $sp(n)$ gives the result in eq. (6.50). For $so(n)$ however this results in an extra factor of two that we have to divide our result with, which explains the difference in eq. (6.51). This factor of two however only applies to $so(n)$ with $n \geq 5$. so for $so(3)$ and $so(4)$ we should still have to factor of 2 in eq. (6.51).

An extra check we can perform is to see if groups that are equivalent will give the same result when the same diagram is calculated. The two combinations of groups we will look at is $so(5) \cong sp(4)$ and $so(3) \cong su(2)$, to see why these groups are essentially the same, we can look at their Dynkin diagram from figure (1.2a) and see that for the respective groups these Dynkin diagrams are equivalent. From eq. (4.21) we see that for colour factors of 3 particle scattering amplitudes we have 1 diagram, for 4 particle scattering amplitudes we get $2! = 2$ diagrams, for 5 particle scattering amplitudes we get $3! = 6$ diagrams and for 6 particle scattering amplitudes we get $4! = 24$ diagrams. Each of these diagrams should give the same result when calculated for equivalent groups. These results are summarised in the tables below, and were calculated using a script in python that I wrote to calculate the colour factors for the four families:

3 particles				
su(n)	so(n)	sp(n)	$\frac{su(2)}{so(3)}$	$\frac{sp(4)}{so(5)}$
$2(n^3 - n)$	$\frac{1}{2}(n^3 - 3n^2 + 2n)$	$\frac{1}{2}(-n^3 - 3n^2 - 2n)$	4	-2

Table 6.1: Colour factor for 3 particle scattering

From these results we see that the 3 particle results differs by a factor of 2 for the case $so(5) \cong sp(4)$, this is as expected since the Coxeter number is the same but the contraction differs by a factor of 2. We see that this is actually a pattern that continues for scattering amplitudes with

4 particles				
su(n)	so(n)	sp(n)	$\frac{su(2)}{so(3)}$	$\frac{sp(4)}{so(5)}$
$4n^4 - 4n^2$	$\frac{1}{2}n^4 - \frac{5}{2}n^3 + 4n^2 - 2n$	$\frac{1}{2}n^4 + \frac{5}{2}n^3 + 4n^2 + 2n$	16	4
$2n^4 - 2n^2$	$\frac{1}{4}n^4 - \frac{5}{4}n^3 + 2n^2 - n$	$\frac{1}{4}n^4 + \frac{5}{4}n^3 + 2n^2 + n$	16	4

Table 6.2: Colour factors for 4 particle scattering

5 particles				
su(n)	so(n)	sp(n)	$\frac{su(2)}{so(3)}$	$\frac{sp(4)}{so(5)}$
$8n^5 - 8n^3$	$\frac{1}{2}n^5 - \frac{7}{2}n^4 + 9n^3 - 10n^2 + 4n$	$-\frac{1}{2}n^5 - \frac{7}{2}n^4 - 9n^3 - 10n^2 - 4n$	64	-8
$2n^5 - 2n^3$	$\frac{5}{4}n^5 - \frac{7}{8}n^4 + \frac{9}{4}n^3 - \frac{5}{2}n^2 + n$	$-\frac{5}{4}n^5 - \frac{7}{8}n^4 - \frac{9}{4}n^3 - \frac{5}{2}n^2 - n$	64	-8
$2n^5 - 2n^3$	$\frac{5}{4}n^5 - \frac{7}{8}n^4 + \frac{9}{4}n^3 - \frac{5}{2}n^2 + n$	$-\frac{5}{4}n^5 - \frac{7}{8}n^4 - \frac{9}{4}n^3 - \frac{5}{2}n^2 - n$	64	-8
$4n^5 - 4n^3$	$\frac{1}{4}n^5 - \frac{7}{4}n^4 + \frac{9}{2}n^3 - 5n^2 + 2n$	$-\frac{1}{4}n^5 - \frac{7}{4}n^4 - \frac{9}{2}n^3 - 5n^2 - 2n$	64	-8
0	0	0	/	/
$4n^5 - 4n^3$	$\frac{1}{4}n^5 - \frac{7}{4}n^4 + \frac{9}{2}n^3 - 5n^2 + 2n$	$-\frac{1}{4}n^5 - \frac{7}{4}n^4 - \frac{9}{2}n^3 - 5n^2 - 2n$	64	-8

Table 6.3: Colour factors for 5 particle scattering

more particles. For four particles we have two contractions and thus we have an extra factor of 2 so we in total ends up with a factor of 4. This pattern is sustained for every permutation of the lines in the diagrams from eq. (4.21) and for each extra particle we have an extra contraction and an extra factor of 2, where the sign changing is caused by the contracting of the skew symmetric invariant of $sp(n)$. For the case $su(2) \cong so(3)$, the dual Coxeter number differs by a factor of 2, so we would expect to see the same pattern as with the previous case (without the sign changing). However what we see is that there is a factor of 4 for each extra particle in the scattering factor, this is due to the adjoint casimir for $so(3)$ still not giving twice the dual Coxeter number, even though the normalization from the Cartan algebra has it as the normalization. This is a limitation of the birdtrack method, but luckily for us it only applies to calculations with the algebras of $so(3)$ and $so(4)$.

6 particles				
su(n)	so(n)	sp(n)	$\frac{su(2)}{so(3)}$	$\frac{sp(4)}{so(5)}$
$16n^6 - 16n^4$	$\frac{1}{2}n^6 - \frac{9}{2}n^5 + 16n^4 - 28n^3 + 24n^2 - 8n$	$\frac{1}{2}n^6 + \frac{9}{2}n^5 + 16n^4 + 28n^3 + 24n^2 + 8n$	256	16
$2n^6 + 22n^4 - 24n^2$	$\frac{1}{16}n^6 - \frac{3}{16}n^5 + \frac{83}{16}n^4 - \frac{71}{4}n^3 + \frac{107}{4}n^2 - 13n$	$\frac{1}{16}n^6 + \frac{3}{16}n^5 + \frac{83}{16}n^4 + \frac{71}{4}n^3 + \frac{107}{4}n^2 + 13n$	256	16
$24n^4 - 24n^2$	$-\frac{3}{16}n^5 + \frac{51}{16}n^4 - \frac{57}{4}n^3 + \frac{93}{4}n^2 - 12n$	$\frac{3}{16}n^5 + \frac{51}{16}n^4 + \frac{57}{4}n^3 + \frac{93}{4}n^2 + 12n$	256	16
$4n^6 - 4n^4$	$\frac{1}{8}n^6 - \frac{9}{8}n^5 + 4n^4 - 7n^3 + 6n^2 - 2n$	$\frac{1}{8}n^6 + \frac{9}{8}n^5 + 4n^4 + 7n^3 + 6n^2 + 2n$	256	16
$4n^6 - 4n^4$	$\frac{1}{8}n^6 - \frac{9}{8}n^5 + 4n^4 - 7n^3 + 6n^2 - 2n$	$\frac{1}{8}n^6 + \frac{9}{8}n^5 + 4n^4 + 7n^3 + 6n^2 + 2n$	256	16
$4n^6 - 4n^4$	$\frac{1}{8}n^6 - \frac{9}{8}n^5 + 4n^4 - 7n^3 + 6n^2 - 2n$	$\frac{1}{8}n^6 + \frac{9}{8}n^5 + 4n^4 + 7n^3 + 6n^2 + 2n$	256	16
0	0	0	/	/
0	0	0	/	/
$4n^6 - 4n^4$	$\frac{1}{8}n^6 - \frac{9}{8}n^5 + 4n^4 - 7n^3 + 6n^2 - 2n$	$\frac{1}{8}n^6 + \frac{9}{8}n^5 + 4n^4 + 7n^3 + 6n^2 + 2n$	256	16
0	0	0	/	/
0	0	0	/	/
$4n^6 - 4n^4$	$\frac{1}{8}n^6 - \frac{9}{8}n^5 + 4n^4 - 7n^3 + 6n^2 - 2n$	$\frac{1}{8}n^6 + \frac{9}{8}n^5 + 4n^4 + 7n^3 + 6n^2 + 2n$	256	16
$8n^6 - 8n^4$	$\frac{1}{4}n^6 - \frac{9}{4}n^5 + 8n^4 - 14n^3 + 12n^2 - 4n$	$\frac{1}{4}n^6 + \frac{9}{4}n^5 + 8n^4 + 14n^3 + 12n^2 + 4n$	256	16
$24n^4 - 24n^2$	$-\frac{3}{16}n^5 + \frac{51}{16}n^4 - \frac{57}{4}n^3 + \frac{93}{4}n^2 - 12n$	$+\frac{3}{16}n^5 + \frac{51}{16}n^4 + \frac{57}{4}n^3 + \frac{93}{4}n^2 + 12n$	256	16
$24n^4 - 24n^2$	$-\frac{3}{16}n^5 + \frac{51}{16}n^4 - \frac{57}{4}n^3 + \frac{93}{4}n^2 - 12n$	$+\frac{3}{16}n^5 + \frac{51}{16}n^4 + \frac{57}{4}n^3 + \frac{93}{4}n^2 + 12n$	256	16
$8n^6 - 8n^4$	$\frac{1}{4}n^6 - \frac{9}{4}n^5 + 8n^4 - 14n^3 + 12n^2 - 4n$	$\frac{1}{4}n^6 + \frac{9}{4}n^5 + 8n^4 + 14n^3 + 12n^2 + 4n$	256	16
$8n^6 - 8n^4$	$\frac{1}{4}n^6 - \frac{9}{4}n^5 + 8n^4 - 14n^3 + 12n^2 - 4n$	$\frac{1}{4}n^6 + \frac{9}{4}n^5 + 8n^4 + 14n^3 + 12n^2 + 4n$	256	16
$2n^6 - 2n^4$	$\frac{1}{16}n^6 - \frac{9}{16}n^5 + 2n^4 - \frac{7}{2}n^3 + 3n^2 - n$	$\frac{1}{16}n^6 + \frac{9}{16}n^5 + 2n^4 + \frac{7}{2}n^3 + 3n^2 + n$	256	16
$24n^4 - 24n^2$	$-\frac{3}{16}n^5 + \frac{51}{16}n^4 - \frac{57}{4}n^3 + \frac{93}{4}n^2 - 12n$	$+\frac{3}{16}n^5 + \frac{51}{16}n^4 + \frac{57}{4}n^3 + \frac{93}{4}n^2 + 12n$	256	16
$2n^6 - 2n^4$	$\frac{1}{16}n^6 - \frac{9}{16}n^5 + 2n^4 - \frac{7}{2}n^3 + 3n^2 - n$	$\frac{1}{16}n^6 + \frac{9}{16}n^5 + 2n^4 + \frac{7}{2}n^3 + 3n^2 + n$	256	16
0	0	0	/	/
0	0	0	/	/
$2n^6 - 2n^4$	$\frac{1}{16}n^6 - \frac{9}{16}n^5 + 2n^4 - \frac{7}{2}n^3 + 3n^2 - n$	$\frac{1}{16}n^6 + \frac{9}{16}n^5 + 2n^4 + \frac{7}{2}n^3 + 3n^2 + n$	256	16
$2n^6 - 2n^4$	$\frac{1}{16}n^6 - \frac{9}{16}n^5 + 2n^4 - \frac{7}{2}n^3 + 3n^2 - n$	$\frac{1}{16}n^6 + \frac{9}{16}n^5 + 2n^4 + \frac{7}{2}n^3 + 3n^2 + n$	256	16

Table 6.4: Colour factors for 6 particle scattering

 G_2

With our recent success for the four families we will proceed right away with our first treatment of one of the exceptional groups. This chapter is based on chapter 16 in [3]. The group we will focus on here is G_2 , which is defined as having two primitive invariants. The first is a symmetric two index invariant g^{ab} , that is exactly like the invariant we saw in our treatment of $so(n)$. We will draw this invariant as:

$$\begin{aligned}
 g^{ab} &= a \text{-----} b \\
 g^{ab} &= g^{ba}
 \end{aligned}
 \tag{7.1}$$

It acts as a metric on the group so that it can raise and lower indices at will, and therefore we will omit any arrows on our lines. The second invariant we have in our group is a three index

anti-symmetric invariant defined as:

$$l^{abc} = \begin{array}{c} a \\ | \\ \bullet \\ / \quad \backslash \\ b \quad c \end{array} \quad (7.2)$$

$$l^{abc} = -l^{acb} \quad \Rightarrow \quad \begin{array}{c} | \\ \bullet \\ / \quad \backslash \\ \quad \quad \end{array} = - \begin{array}{c} | \\ \bullet \\ \backslash \quad / \\ \quad \quad \end{array}$$

From the primitiveness assumption we know that if we contract two indices on a l^{abc} tensor with two indices on another l^{cbd} tensor it should be proportional to a sum of the possible combinations of 2-index tree-invariants. Since there only exists one such tree invariant, we can immediately fix the normalization of the 3 index invariant. This is drawn as:

$$\begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \bullet \end{array} = \alpha \text{ --- } \quad (7.3)$$

Throughout this chapter we set $\alpha = 1$. As with the families our task is to decompose the space $V \otimes V$ into a sum of tree invariant subspaces. This we can write in a general form as:

$$0 = A \begin{array}{c} \text{---} \\ \text{---} \end{array} + B \begin{array}{c} \text{---} \\ \text{---} \end{array} + C \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} + D \begin{array}{c} \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + E \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} + F \begin{array}{c} \text{---} \\ \diagdown \quad \diagup \\ \text{---} \end{array} \quad (7.4)$$

However instead of going directly to the characteristic equation like we did with the the families it is easier this time to first find some relations among the subspaces. First we start by using the anti-symmetric subspace from the right which gives:

$$0 = A' \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} + E \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} + F' \begin{array}{c} \text{---} \\ | \\ \text{---} \end{array} \quad (7.5)$$

If we refer to the subspaces by the constant in front of them, we see that B disappears naturally when anti-symmetrized, and the subspace E just returns itself by using the anti-symmetric property from eq. (7.2). A and C are combined, and thus becomes a new constant given by $C' = A - C$. D and F are combined in a similar manner and the new constant is given by $F' = D - F$. If instead we anti-symmetrize 3 indices we see that A , B and C vanish for the same reason that B vanishes in the first example. We will treat the rest of the diagrams in the birdtrack notation, so that we

get:

$$\begin{aligned}
 0 &= D \text{ [diagram]} + E \text{ [diagram]} + F \text{ [diagram]} = D \text{ [diagram]} + E \text{ [diagram]} - F \text{ [diagram]} \\
 &= F' \text{ [diagram]} + E \text{ [diagram]} = F' \text{ [diagram]} + E \text{ [diagram]} \\
 &= -F' \text{ [diagram]} + E \text{ [diagram]} = (E - F') \text{ [diagram]}
 \end{aligned}
 \tag{7.6}$$

This can be shortened to:

$$0 = (E - F') \text{ [diagram]}
 \tag{7.7}$$

This relation differs from the relation in [3] by a minus sign instead of a plus between E and F' , but as it turns out this minus sign is actually needed in order to produce the later results, so this must be a notational error in [3]. If $E \neq F'$ the diagram must be the one that equals zero, in that case we actually get an expression that equals the Jacobi relation, and if $A' \neq 0$ in eq. (7.5) we actually recover the group $su(3)$. A proof of this is given in [3], likewise if $A' \neq 0$ we actually only have the Jacobi-relation to go on from, and any simple-Lie group is a solution to this relation. The proof of this is also in [3]. The only relation we have not exposed is then the case $E = F'$. If that is the case we can rewrite eq. (7.5) into:

$$\text{[diagram]} + \text{[diagram]} = A'' \text{ [diagram]}
 \tag{7.8}$$

Tracing the upper line we can get an expression for A'' which turns out to be:

$$\begin{aligned}
 &\text{[diagram]} + \text{[diagram]} = A'' \text{ [diagram]} \\
 \Rightarrow &\text{[diagram]} + \frac{1}{2} \text{[diagram]} - \frac{1}{2} \text{[diagram]} = A'' \frac{(n-1)}{2} \text{[diagram]} \\
 \Rightarrow &\text{[diagram]} \frac{3}{2} = A'' \frac{(n-1)}{2} \Rightarrow A'' = \frac{3}{(n-1)}
 \end{aligned}
 \tag{7.9}$$

Which makes us reach our first relation in our quest to find the adjoint projection operator for G_2 . The second relation can be found by symmetrizing the two upper indices in eq. (7.8) and rotating

the diagrams by 90 degrees, which gives:

$$\begin{aligned}
 & \text{Y-junction} + \frac{1}{2} \left\{ \text{parallel lines} - \text{cross} \right\} = \frac{3}{2(n-1)} \left\{ \text{parallel lines} - \text{cross} \right\} \\
 \rightarrow & \text{rotated Y-junction} + \frac{1}{2} \left\{ \text{rotated parallel lines} - \text{rotated cross} \right\} = \frac{3}{2(n-1)} \left\{ \text{rotated parallel lines} - \text{rotated cross} \right\} \\
 \text{rotate:} & \\
 \rightarrow & \text{vertical parallel lines} + \frac{1}{2} \left\{ - \text{vertical cross} \right\} = \frac{3}{2(n-1)} \left\{ \text{vertical parallel lines} - \text{vertical cross} \right\} \rightarrow \\
 \frac{3}{2} \text{vertical parallel lines} &= \frac{3}{2(n-1)} \left\{ \text{curved lines} - \text{vertical parallel lines} \right\} \rightarrow \text{vertical parallel lines} = \frac{1}{(n-1)} \left\{ \text{curved lines} - \text{vertical parallel lines} \right\}
 \end{aligned} \tag{7.10}$$

Adding together these two relations we get:

$$\text{Y-junction} + \text{rotated Y-junction} = \frac{1}{n-1} \left\{ \text{parallel lines} - \text{cross} + \text{curved lines} \right\} \tag{7.11}$$

From these two relations we see that the diagrams from D and F , are reducible on both the symmetric and anti-symmetric subspaces, and that from eq. (7.11) it can be expressed in terms of E , whereas E is only reducible on the anti-symmetric subspace. Because they are related we can choose which of these two diagrams we want to keep, so since E is easier to work with we keep that diagram. So now we have the spaces A , B , C and E left that we need to find projection operators for. To do this we start our like we did for $so(n)$ and inserts the flip space C into eq. (5.26) where we find the eigenvalues to be $\lambda_1 = 1$ and $\lambda_2 = -1$. We insert this in eq. (5.27) from which we recover the same two projection operators as we had for $so(n)$:

$$\begin{aligned}
 P_1 &= \frac{1}{2} \left\{ \text{parallel lines} + \text{cross} \right\} = \text{vertical parallel lines} \\
 P_2 &= \frac{1}{2} \left\{ \text{parallel lines} - \text{cross} \right\} = \text{thick vertical parallel lines}
 \end{aligned} \tag{7.12}$$

From this we see that the projection operator P_1 acts as the identity on diagram B (the trace), so again like for $so(n)$ this can be decomposed into two further projection operators. The new case that differs from $so(n)$, which we have to deal with here, is that the P_2 projection operator now acts as the identity on diagram E , so decomposing this space we first finds the eigenvalues from

So putting this back into the last term in eq. (7.16) we get:

$$\begin{array}{c} \text{Y-shape} \\ \text{---} \\ \text{---} \end{array} - \frac{2}{n-1} \begin{array}{c} \text{Y-shape} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \quad (7.18)$$

This relation can be rotated and the bottom anti-symmetrization can be expanded in so that we can make the diagram look like:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} = \frac{2}{n-1} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} - \frac{2}{3} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} + \frac{1}{3} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \quad (7.19)$$

The first two of the bottom lines is anti-symmetric in the left diagram, and the last two bottom lines are also anti-symmetric. This can only be realised on the left side if we fix $n=7$. Thus this reduction relation is only viable for $n = 7$. This relation is paramount in the calculation of our colour factors from eq. (4.21). The reason is that in these calculations we will encounter all sorts of different combinations of the 3-index invariant tensor, so having a general formula for how to reduce 3 connected 3-invariants enables us to solve any calculation regarding G_2 , and thus solves the group for us.

Now that we have the dimension of the fundamental representation we can calculate the dimension of the adjoint representation by tracing the adjoint projection operator:

$$d_{A_{G_2}} = \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} - \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} = \frac{n(n-1)}{2} - n = 14 \quad (7.20)$$

As a quick check we see that this does indeed match the known result from the literature. We can further calculate two extra identities that will help us when calculating colour factors. The first one is obtained by tracing eq. (7.8) from the left with the 3-index invariant, which gives:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} + \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} = \frac{3}{n-1} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} \\
 \begin{array}{c} \text{---} \\ \text{---} \end{array} + \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} = \frac{1}{2} \begin{array}{c} \text{---} \\ \text{---} \end{array} \\
 \begin{array}{c} \text{---} \\ \text{---} \end{array} = \frac{-1}{2} \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad (7.21)$$

From this we get an easy relation for how to contract a loop of 3 3-invariant tensors. The second relation arises from tracing the first two bottom legs in the invariance condition eq. (7.15) with the

3-index primitive invariant, and contracting the adjoint line with a generator. This then gives us:

$$\begin{aligned}
 0 &= \text{[Diagram with thick bar]} = \text{[Diagram 1]} - \text{[Diagram 2]} - \text{[Diagram 3]} + \text{[Diagram 4]} + \text{[Diagram 5]} - \text{[Diagram 6]} \\
 &= 4 \text{[Diagram 7]} + 2 \text{[Diagram 8]} \quad \rightarrow \quad 4 \text{[Diagram 9]} + 2 \text{[Diagram 10]} = 0 \\
 &\quad \rightarrow \text{[Diagram 11]} = \frac{1}{2} \text{[Diagram 12]}
 \end{aligned}
 \tag{7.22}$$

The last equality sign uses that the adjoint projection operators are anti-symmetric with regards to interchange of the two legs in either end. Now that we have all the relations in place and the adjoint projection operator is known, lets see if we can confirm that this approach works as intended. To do so we return to our usual calculation of the 3-particle scattering colour factor, and as always we start from eq. (6.11). Where we use the adjoint projection operator to expand the diagrams, given by:

$$P_{A_{G_2}} = \text{[Diagram 13]} - \text{[Diagram 14]}
 \tag{7.23}$$

alongside the identities from eq. (7.3) and (7.2). For the case of G_2 the first terms yields:

$$\begin{aligned}
 \text{[Diagram 15]} &= \frac{1}{4} \left\{ \text{[Diagram 16]} - \text{[Diagram 17]} - \text{[Diagram 18]} + \text{[Diagram 19]} \right\} \\
 &+ \frac{1}{2} \left\{ - \text{[Diagram 20]} + \text{[Diagram 21]} - \text{[Diagram 22]} + \text{[Diagram 23]} \right\} + \text{[Diagram 24]} \\
 &= \frac{(n-2)}{4} \text{[Diagram 25]} - \frac{1}{4} \text{[Diagram 26]}
 \end{aligned}
 \tag{7.24}$$

And the second term gives:

$$\begin{aligned}
 & \text{Diagram} = \frac{1}{4} \left\{ \text{Diagram 1} - \text{Diagram 2} - \text{Diagram 3} + \text{Diagram 4} \right\} \\
 & + \frac{1}{2} \left\{ \text{Diagram 5} - \text{Diagram 6} - \text{Diagram 7} + \text{Diagram 8} \right\} + \text{Diagram 9} \\
 & = \frac{(2-n)}{4} \text{Diagram 10} + \frac{1}{4} \text{Diagram 11}
 \end{aligned} \tag{7.25}$$

So inserting these two results in eq. (6.11) we will in total get:

$$\begin{aligned}
 & 2 \left\{ \frac{(n-2)}{4} \text{Diagram 10} - \frac{1}{4} \text{Diagram 11} - \frac{(2-n)}{4} \text{Diagram 10} - \frac{1}{4} \text{Diagram 11} \right\} \\
 & = (n-3) \text{Diagram 10} = 56
 \end{aligned} \tag{7.26}$$

Since we are calculating these colour factors for the exceptional groups, we see that the results is just a number. Looking at figure (6.1) we see that the dual coxeter number is 4, and since the dimension is 14, we find that:

$$f^{abc} f_{acb} = h^\vee d_{A_{g_2}} \tag{7.27}$$

Here we again see that the factor of 2 that we see in eq. (6.50) is missing. This also stems from the normalization of the bilinear form as it did with $so(n)$.

E_6

What we have done so far is following the path that [3] laid down for us for each group. That path consisted of starting from the primitive invariants finding invariant matrices and possible relations among the tree invariants one can make of the primitive invariants. The next step is to find projection operators to split different spaces (we have focused on the $V \otimes V$ or $V \otimes \bar{V}$ space) into projection operators which enables one to express everything in terms of lines in the fundamental representation. This method is useful if one's goal is to define and describe the entire group, however this approach has its limitations. If we take G_2 as an example, we saw in the last chapter that there was a relation that enables you to iteratively reduce any combination of three 3-invariants given by eq. (7.19). This identity is what makes us able to say that any diagram we can get as a colour factor from eq. (4.21) is solvable, through repeated reduction. The problem arises when

Which combined with the dimensions of the other 2 representations gives exactly what we need in order to reach the adjoint dimension of e_6 , $35 + 3 + 40 = 78$. So far so good, but how do we turn this dimension into a generators? Well generators in birdtrack notation are essentially just lines of the given representation where one has attached an adjoint line. This is motivated by looking at a generator in the tensor notation $(T_R^i)_b^a$. Where the indices a and b are dependant on the representation R , while the index i is always in the adjoint representation. The next question is, since we have four lines in the new 40 dimensional "special" representation, we need to figure out which line we should connect the adjoint to. Well actually it turns out that we need to add the adjoint line to each of the four lines in the special generator, which gives:

$$(T_s^i)_b^a = \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} \quad (8.3)$$

However now we see that there is a new problem, because it should not matter which of the indices a or b that we chose to interact with, so the fact that we have an anti-symmetric symbol on one side of the adjoint line violates this principle. Therefore we will choose to put another anti-symmetric symbol on the other side of the adjoint line. This will not change anything for our calculation of the dimension of the special space, since the anti-symmetrization operator does not affect an anti-symmetric space. We therefore get:

$$(T_s^i)_b^a = \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} \\ = \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} + 3 \begin{array}{c} i \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} \\ \quad \quad \quad A \quad \quad \quad B \quad (8.4)$$

The last equality follows from the fact that if we change one line in the middle one up with the anti-symmetric symbol in one end it will give a minus sign, but if we give change a line one up in both ends it will give 2 minus sign cancelling each other. This results in the 3 last sub-generators to be identical. We have further given the two remaining sub-generators the name of A and B. So now are we sure these are the correct generators for for e_6 ? Well almost, so far they satisfies the dimension needed so we in total get the dimension of the adjoint representation of e_6 , however it would be nice to make another check to be sure that this is correct. The check we now do is to see if the special generator satisfies the Jacobi relation defined in the birdtrack notation in eq. (5.61). So doing this calculation we find that for the sub-generators A the Jacobi relation is:

$$\begin{array}{c} \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} - \begin{array}{c} \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} \quad (8.5)$$

and expanding the diagram labeled by 2 we find:

$$\begin{aligned}
 & 9 \left[\text{Diagram 2} \right] = 9 \left[\text{Diagram 2} \right] = \\
 & \frac{9}{6} \left\{ \text{Diagram 2a} - \text{Diagram 2b} - \text{Diagram 2c} + \text{Diagram 2d} + \text{Diagram 2e} - \text{Diagram 2f} \right\} \quad (8.9) \\
 & = \frac{9}{6} \left\{ 2 \left[\text{Diagram 2a} \right] - 4 \left[\text{Diagram 2b} \right] \right\}
 \end{aligned}$$

The term 3 trivially expands in the same way as it did for the $su(2)$ line, so that it gives:

$$3 \left[\text{Diagram 3} \right] = 3 \left[\text{Diagram 3a} \right] - 3 \left[\text{Diagram 3b} \right] \quad (8.10)$$

From this we see that the first result in the calculation of diagram 1 and 2 together equals the result from the calculation of diagram 3. Therefore we must have:

$$4 \left[\text{Diagram 4a} \right] = 4 \left[\text{Diagram 4b} \right] \quad (8.11)$$

This can actually be verified fairly easily by interchanging the upper two lines in the left diagram on both anti-symmetrization symbols. This will result in the adjoint line that was connected at the top left part of the diagram is now connected to the top right, and likewise in reverse for the other adjoint line. Which results in them now being directly under the point that they connect to above the special generator. This shows that the two diagrams are identical, and thus cancel each other in the Jacobi calculation.

With these confirmations that we have found the correct generators, lets see if we can do a calculation and hopefully get the correct result. Returning to the usual calculation of the colour factor for 3-particle scattering eq. (6.10), we see that the calculation for e_6 differs compared to other groups since we no have 4 options for each of the two general adjoint generators in the calculation. Luckily for us though it turns out that not all combinations are valid, so we do not have to calculate $4^2 = 16$ diagrams. The only valid diagrams we can make are those where each line in the diagram connects with another line of the same kind, so that lines with the same colour only meet lines with a matching colour, and lines in different representations only meet lines in the same representation. Taking care that we satisfy these requirements we end up with the following 4 diagrams for the

colour factor of the 3-particle scattering amplitude:

$$\begin{array}{cccc}
 \text{Diagram 1} & + & \text{Diagram 2} & + & \text{Diagram 3} & + & 9 \times \text{Diagram 4} \\
 \text{(1)} & & \text{(2)} & & \text{(3)} & & \text{(4)}
 \end{array} \tag{8.12}$$

Diagram 1 and 2 is just the result we got from eq. (6.14), with $n = 2$ and $n = 6$ respectively. Diagram 3 can be calculated using the adjoint projection operator for $su(n)$ (6.8) and the formula for contraction a fully anti-symmetric tensor with itself eq. (5.19). The diagram thus gives:

$$\text{Diagram 3} = \text{Diagram 3} - \frac{1}{n} \text{Diagram 3} = 4 \cdot \frac{6!}{3!^2} - \frac{2}{2} \cdot \frac{6!}{3!^2} = 80 - 20 = 60 \tag{8.13}$$

Using the same identities for diagram 4 we get:

$$9 \times \text{Diagram 4} = 9 \times \text{Diagram 4} - \frac{9}{n} \times \text{Diagram 4} = 9 \left(\frac{4}{3}\right)^2 \times \text{Diagram 4} - \frac{9}{6} \times \text{Diagram 4} = 18 \left(\frac{4}{3}\right)^2 \frac{6!}{2!6!} - \frac{18}{6} \frac{6!}{3!3!} = 420 \tag{8.14}$$

One thing that we have to be careful of is that in diagram 1 and 2, we can not distinguish the legs in the diagram from each other, however this is not the case for the legs in diagrams 3 and 4, where one of the legs is in the adjoint representation and the other two legs is in either the 2 or 20 dimensional representation. Because of the legs in diagram 3 and 4 being distinguishable from each other, we have to sum each possible legs combination together. The different combinations though all give the same result so we simply just have to multiply the results of diagram 3 and 4 with a factor of 3. Thus in total we get:

$$2(6^3 - 6) + 2(2^3 - 2) + 3 \cdot 60 + 3 \cdot 420 = 1872 \tag{8.15}$$

Which is very lucky for us since looking at the table of dual Coxeter numbers (6.1) we see that the dual Coxeter number of E_6 is $h^\vee = 12$, so dividing our previous result for the 3 particle colour factor by the adjoint dimension of e_6 we find:

$$\frac{1872}{78} = 24 = 2 \cdot h_{E_6}^\vee \tag{8.16}$$

Which is exactly what we expect since we express e_6 in terms of subgroups of $su(2)$ and $su(6)$. This method of calculating diagrams of e_6 in the adjoint representation is completely generalisable to every diagram that can occur in the colour factors from eq. (4.21), which means that we have solved e_6 regarding any calculation we can encounter.

Likewise from our calculations in the chapter on e_6 diagram 1 gives:

$$16 \left[\text{Diagram 1} \right] = \frac{16}{24} \left\{ 6 \left[\text{Diagram 2} \right] - 18 \left[\text{Diagram 3} \right] \right\} \quad (9.6)$$

and the calculations for diagram 2 gives:

$$16 \left[\text{Diagram 2} \right] = \frac{16}{24} \left\{ 6 \left[\text{Diagram 4} \right] - 18 \left[\text{Diagram 5} \right] \right\} \quad (9.7)$$

substituting these result back into eq. (9.5) we see that the two diagrams with the front factor of 6 actually reproduces the the Jacobi relation. So what is left to show is that the diagrams with the front factor of 18 cancels each other. This is done by showing that they are actually equal, so when they are subtracted it just gives zero:

$$\begin{aligned} \left[\text{Diagram 3} \right] &= \left[\text{Diagram 5} \right] \\ \left[\text{Diagram 4} \right] &= - \left[\text{Diagram 5} \right] \\ \left[\text{Diagram 2} \right] &= \left[\text{Diagram 5} \right] \\ \left[\text{Diagram 1} \right] &= \left[\text{Diagram 5} \right] \end{aligned} \quad (9.8)$$

So from this we see that the generator from eq. (9.3) satisfies the Jacobi relation. Moving on with the now very familiar case of calculating the colour factor for 3 particle scattering, we now have 2 different generators, and have to combine them in every possible combination to get all the diagrams we have to calculate. This turns out to be very simple since the two diagrams are unable to combine in the 3-particle case, thus we end up with the following diagrams:

$$\text{Diagram 1} + 16 \text{Diagram 2} \quad (9.9)$$

The first diagram (diagram 1) is the familiar result from our calculation of colour factors for 3-particle scattering amplitudes for the algebra $su(n)$ eq. (6.14) where in this case $n = 8$. The second

diagram (diagram 2) can be calculated using the reduction identity for anti-symmetric tensors given by eq. (5.19). Furthermore we again have to take into account that the legs in the generators for the 70 dimensional representation of $su(8)$ are not all the same, so that we have to sum up all possible configurations of how to order the legs. Luckily all diagrams turns out to be equal to each other, so that we only have to account for this by multiplying with a factor of 3. Therefore we can calculate the value of diagram 2, which gives:

$$\begin{aligned}
 16 \cdot 3 \cdot \text{Diagram 1} &= 16 \cdot 3 \cdot \text{Diagram 2} - \frac{16 \cdot 3}{n} \cdot \text{Diagram 3} \\
 &= 16 \cdot 3 \left(\frac{5}{4}\right)^2 \cdot \text{Diagram 4} - \frac{16 \cdot 3}{8} \cdot \text{Diagram 5} \\
 &= 16 \cdot 3 \left(\frac{5}{4}\right)^2 \frac{8!}{3!5!} - \frac{16 \cdot 3}{8} \frac{8!}{4!4!} = 3780
 \end{aligned} \tag{9.10}$$

Therefore combining the results from the two diagrams, we find the final result to be:

$$3780 + 2(n^3 - n) = 3780 + 2(8^3 - 8) = 4788 \tag{9.11}$$

If we divide this number by the adjoint dimension of E_7 we find:

$$\frac{4788}{133} = 36 = 2h_{E_7}^\vee \tag{9.12}$$

Where the dual Coxeter number is found in table 6.1. This is exactly the result we were looking for. From this we also see that since we have represented e_7 using $su(8)$ subgroups we find the result to be equal to $2h_{E_7}^\vee$ instead of just $h_{E_7}^\vee$. Since the calculations here are easily translated to the general case, we have now solved the algebra e_7 regarding every possible colour factor of a type from eq. (4.21).

F_4 and E_8

Riding on the wave of our results from e_6 and e_7 , we throw ourselves head first into the last two algebras that we need to solve: f_4 and e_8 . In this chapter we will as it turns out solve both of these groups simultaneously. The reason we can do that stems from the fact that for both of these algebras we can decompose the adjoint representation as a sum of two subalgebras, one of $so(n)$ and the other of $spin(n)$. For the case of e_8 , it can be decomposed as a sum of $so(16)$ alongside $spin(16)$, while f_4 can be decomposed as a sum of $so(9)$ plus $spin(9)$, see [20]. The reason that we can solve both of these algebras simultaneously is that we can solve this type of decomposition in the case of a general n . However before we can start to solve this case and check that we get the correct result, we need to first find out how to describe spinors in the Birdtrack notation, we will base this on [3].

Spinors in birdtrack

Spinors are representations on which γ_μ matrices act, where γ_μ satisfy a Clifford algebra:

$$\gamma_\mu \gamma_\nu + \gamma_\nu \gamma_\mu = 2g_{\mu\nu} \mathbf{1} \tag{10.1}$$

Where $\mathbf{1}$ is the identity matrix for the spinor representation and $g_{\mu\nu}$ is the 2 index primitive invariant of $so(n)$. The Gamma matrices are linked to their $so(n)$ counterpart in that their index is an index that lives in the space of $so(n)$. Furthermore it can be shown [3] that the dimension of the gamma matrices depends on the specific n in $so(n)$, where for $so(2r)$ and $so(2r + 1)$ gamma matrices are realised as $[2^r \times 2^r]$ dimensional matrices. This is a typical physicist way of defining spinors. One could instead go a more abstract mathematical way and define a vector space V with a norm N and a product \cdot , where every element $x \in V$ satisfies $x \cdot x = N(x)$. In this way we would eventually be lead to a complete classification of spinors and their representations. However such a treatment of spinors is not needed in our case, and we will instead stick to the above definition.

With this in mind we will define the gamma matrices in the Birdtrack notation as:

$$\begin{aligned} (\gamma^i)_{ab} &= \begin{array}{c} i \\ | \\ a \text{---} \text{---} \text{---} b \end{array} \quad a, b = 1, \dots, 2^{2r} \text{ or } 2^{2r-1} \\ \mathbf{1}_{ab} &= \begin{array}{c} a \text{---} \text{---} \text{---} b \end{array} \quad i = 1, \dots, n \\ \text{tr}(\mathbf{1}_{ab}) &= \text{---} \circ \text{---} \end{array} \tag{10.2}$$

Where the spinor lines are depicted with a dotted line, and the $so(n)$ fundamental representation line is depicted with a solid line. Looking at the definition of the Clifford algebra we see that we can depict it in birdtrack notation as:

$$\begin{array}{c} | \\ \text{---} \text{---} \text{---} \\ | \end{array} = \text{---} \cup \text{---} \tag{10.3}$$

From this we see that we can take a general product of gamma matrices and split them up so it only consists of anti-symmetric products of gamma matrices. This leads to the general formula for how to express a matrix product of an anti-symmetric gamma matrix with a single gamma matrix:

$$\begin{array}{c} 1 \quad 2 \quad 3 \quad \dots \quad p \\ | \quad | \quad | \quad \dots \quad | \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} = \begin{array}{c} 1 \quad 2 \quad 3 \quad \dots \quad p \\ | \quad | \quad | \quad \dots \quad | \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} + (p-1) \begin{array}{c} 1 \quad 2 \quad 3 \quad \dots \quad p \\ | \quad | \quad | \quad \dots \quad | \\ \text{---} \text{---} \text{---} \text{---} \text{---} \end{array} \tag{10.4}$$

This relation can be derived by looking at the first term on the right hand side. For each time the line 1 choose to go somewhere, there are $(p - 1)!$ different ways for the other $(p - 1)$ lines to organise themselves. For each of these diagrams there exists a corresponding diagram where the only difference is that line number 1 is interchanged with the line that sits in the first place. Taking these two diagrams and adding the corresponding two diagrams where these lines have the opposite front factor is going to give a diagram where the first line is line number 1, since the opposite sign cancels the first. This could be achieved by adding a diagram similar to the one on the left hand

side where the first two lines have been symmetrized. However since fixing two lines in the mixed diagram results in there being $(p - 2)!$ diagrams left, we need to multiply with a factor of $(p - 1)$ in order to have enough diagrams to cancel all the diagrams from the totally anti-symmetric gamma product. Lastly we can exchange symmetric operator using the definition of the Clifford algebra eq. (10.1) to reach eq. (10.4). To see how this helps us, we will now show how to decompose general products of gamma matrices, where we start of with a product of two gamma matrices:

$$\begin{array}{c} | \quad | \\ \hline \hline \end{array} = \begin{array}{c} \boxed{} \\ | \quad | \\ \hline \hline \end{array} + \begin{array}{c} \blacksquare \\ | \quad | \\ \hline \hline \end{array} = \begin{array}{c} \blacksquare \\ | \quad | \\ \hline \hline \end{array} + \begin{array}{c} \cup \\ \hline \hline \end{array} \quad (10.5)$$

Then we can use that result to get the decomposition for the product of 3 gamma matrices:

$$\begin{aligned} \begin{array}{c} | \quad | \quad | \\ \hline \hline \hline \end{array} &= \begin{array}{c} | \quad \blacksquare \\ | \quad | \\ \hline \hline \hline \end{array} + \begin{array}{c} | \quad \cup \\ | \quad | \\ \hline \hline \hline \end{array} \\ &= \begin{array}{c} \blacksquare \\ | \quad | \quad | \\ \hline \hline \hline \end{array} + \begin{array}{c} \cup \\ | \quad | \\ \hline \hline \hline \end{array} - \begin{array}{c} \Psi \\ | \quad | \\ \hline \hline \hline \end{array} + \begin{array}{c} | \quad \cup \\ | \quad | \\ \hline \hline \hline \end{array} \end{aligned} \quad (10.6)$$

From this we see that only a product of fully anti-symmetric gamma matrices can not be reduced further. Therefore these fully anti-symmetric tensors provide a complete basis for expanding products of gamma matrices. We define these tensors as:

$$\begin{aligned} \Gamma^{(0)} &= \begin{array}{c} \\ \hline \hline \end{array} = \begin{array}{c} | \quad 0 \\ \hline \hline \end{array} \\ \Gamma^{(1)} &= \begin{array}{c} | \\ \hline \hline \end{array} = \begin{array}{c} | \quad 1 \\ \hline \hline \end{array} \\ \Gamma^{(2)} &= \begin{array}{c} \blacksquare \\ | \quad | \\ \hline \hline \end{array} = \begin{array}{c} | \quad 2 \\ \hline \hline \end{array} \\ \Gamma^{(a)} &= \begin{array}{c} \blacksquare \\ | \quad | \quad \dots \\ \hline \hline \hline \end{array} = \begin{array}{c} | \quad a \\ \hline \hline \end{array} \end{aligned} \quad (10.7)$$

We can further use the birdtrack definition of the Clifford algebra to get a recursive formula for the product of p gamma matrices, given by:

$$\begin{aligned} \begin{array}{c} | \quad | \quad | \quad | \\ \hline \hline \hline \hline \end{array} &= 2 \begin{array}{c} \cup \\ | \quad | \quad | \\ \hline \hline \hline \end{array} - \begin{array}{c} \times \\ | \quad | \quad | \\ \hline \hline \hline \end{array} \\ &= 2 \begin{array}{c} \cup \\ | \quad | \quad | \\ \hline \hline \hline \end{array} - 2 \begin{array}{c} \Psi \\ | \quad | \quad | \\ \hline \hline \hline \end{array} + (-1)^2 \begin{array}{c} \times \\ | \quad | \quad | \\ \hline \hline \hline \end{array} \\ \Rightarrow \frac{1}{2} \left\{ \begin{array}{c} | \quad | \quad \dots \quad | \\ \hline \hline \hline \hline \end{array} + (-1)^p \begin{array}{c} \text{crossing} \\ | \quad | \quad \dots \quad | \\ \hline \hline \hline \hline \end{array} \right\} &= \begin{array}{c} \cup \\ | \quad | \quad \dots \quad | \\ \hline \hline \hline \hline \end{array} + \dots + (-1)^p \begin{array}{c} \text{crossing} \\ | \quad | \quad \dots \quad | \\ \hline \hline \hline \hline \end{array} \end{aligned} \quad (10.8)$$

From this we see that if we take the trace of the spinor line and p is an even number we get a formula that relates traces of p gamma matrices to traces of $p - 2$ gamma matrices. for the trace of 2 and 4 gamma matrices this yields:

$$\begin{aligned}
 & \text{---} \bigcirc \text{---} = \bigcirc \text{---} \\
 & \text{---} \bigcirc \text{---} = \bigcirc \left\{ \text{---} \text{---} - \text{---} \text{---} + \text{---} \text{---} \right\} \quad (10.9)
 \end{aligned}$$

This formula for spinor traces of p gamma matrices will give $(p - 1)!!$ different diagrams to calculate, so if p is sufficiently large this will absolutely blow up, and give a completely unnecessary amount of diagrams to evaluate. Luckily for us the solution for this is right around the corner, for as it will turn out later, the only thing we need to consider is spinor traces of anti-symmetric gamma matrices, so let's see how that is calculated.

First let's start off by noticing that if we have a set of anti-symmetric $so(n)$ lines, and we then contract two of them with g^{ab} , the diagram will equal zero:

$$\begin{array}{c} \text{|||} \\ \text{|||} \end{array} \text{---} = 0 \quad (10.10)$$

We can use this to show that the trace of two $\Gamma^{(a)}$ matrices multiplied together is orthogonal. Since if one of the gamma matrices have more lines than the other, then those lines will have to connect back to itself, and we just saw that a self "contacting" g^{ab} line under an anti-symmetric operator equals zero, which proves the point.

$$\text{---} \bigcirc \text{---} = \delta_{ab} a! \bigcirc \text{---} \quad (10.11)$$

The factor $a!$ stems from all the diagrams that survive being being anti-symmetrized from the formula from eq. (10.9). Taking this one step further one can extrapolate the formula for a trace of 3 anti-symmetric gamma matrices to be:

$$\begin{aligned}
 & \begin{array}{c} \text{---} \\ \text{---} \end{array} \bigcirc \begin{array}{c} \text{---} \\ \text{---} \end{array} = \bigcirc \frac{a!b!c!}{s!t!u!} \begin{array}{c} \text{---} \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ \text{---} \end{array} \\
 & s = \frac{1}{2}(c + b - a) \quad t = \frac{1}{2}(c + a - b) \quad u = \frac{1}{2}(a + b - c) \quad (10.12)
 \end{aligned}$$

Because the Γ 's constitute a complete basis we know that we can express a product of gamma matrices as a sum over Γ 's and g^{ab} 's. A product of $\Gamma^{(a)}$ and $\Gamma^{(b)}$, needs to preserve all the outer

that there are 3 distinct ways to do this, however these combinations all give the same result, so we can account for this by simply multiplying with a factor of 3.

The first diagram is just the familiar diagram of $so(n)$ which we solved in eq. (6.31), the second diagram is the spinor diagram which we can now solve:

$$\begin{aligned}
 3 \text{ (dashed circle)} &= \frac{3}{16} \text{ (dashed circle with two internal loops)} = \frac{3}{16} \text{ (dashed circle)} \left\{ \begin{array}{l} \text{(two circles)} \\ - \text{(infinity symbol)} \\ + \text{(figure-eight)} \end{array} \right\} \\
 &= \frac{3}{16} \text{ (dashed circle)} (n^2 - n)
 \end{aligned} \tag{10.17}$$

adding together the result for the adjoint spinor generators and the adjoint $so(n)$ generators we find:

$$\begin{array}{c}
 \textcircled{1} = \frac{n(n-1)(n-2)}{2} + \frac{3}{16} \text{ (dashed circle)} (n^2 - n) \\
 \hline
 \begin{array}{l}
 E_8: \\
 n = 16 \quad \text{ (dashed circle)} = 128 \\
 \text{So therefore:} \\
 \textcircled{1} = 7440
 \end{array}
 \quad \Bigg| \quad
 \begin{array}{l}
 F_4: \\
 n = 9 \quad \text{ (dashed circle)} = 16 \\
 \text{So therefore:} \\
 \textcircled{1} = 468
 \end{array}
 \end{array} \tag{10.18}$$

Looking at figure (6.1), we find the dual Coxeter number for both E_8 and f_4 . Since the dual Coxeter number for e_8 is $h^\vee = 30$ and the adjoint dimension is 248, we see that:

$$7440 = 248 \cdot h_{e_8}^\vee \tag{10.19}$$

Which is exactly what we would expect for a group described by $so(n)$ (the result is missing the factor 2 from eq. (6.51)). For f_4 we know the dual Coxeter number to be $h^\vee = 9$, so looking at the result we see:

$$468 = 52 \cdot h_{f_4}^\vee \tag{10.20}$$

Which again is exactly what we would expect from a group described by subgroups of $so(n)$. So it turns out amazingly that we have solved both e_8 and f_4 simultaneously, what a time to be alive in!

Summary

So lets recap what has been achieved in this thesis. In the chapter on the DDM basis the standard colour ordered scattering amplitudes given by (2.58) was rewritten into a basis where the colour factors are expressed as products of structure constants, which is generators in the adjoint representation, instead of having the generators in their fundamental representation. This basis was then transformed into another basis in the chapter on the Zeppenfeld basis, where the sum over the

kinematic part is summed over irreducible representations of the symmetric group and the sum over the Lie algebra structure constants is hidden in the $C^{(\alpha;jl)}$ factors defined in eq. (4.20). Since the structure of the $C^{(\alpha;jl)}$ factors does not depend on the Lie algebra we use in the scattering amplitude, this enables us to easily compare the result of amplitude calculations where we use different Lie algebras. What was then left was the question of how to calculate the colour factors from eq. (4.21) for each of the simple compact Lie algebras.

To answer this question we used the birdtrack formalism[3] to express group theoretic formulas and results in a pictorial way allowing us to doodle us to our answers. The method of decomposing the fundamental spaces $V \otimes \bar{V}$ or $V \otimes V$ into irreducible subspaces was then formulated in the birdtrack notation, where the subspaces were expressed as projection operators. The adjoint projection operator could then be singled out using the invariance condition defined in eq. (5.51). The adjoint projection operators then enables us to calculate the diagrams from eq. (4.21) after using the trace based identity for the structure constants defined in eq. (5.60). This enabled us to tackle the algebras of $su(n)$, $so(n)$, $sp(n)$ and g_2 . For g_2 we saw that the exceptional groups had primitive invariants with more than two indices, which meant that we need extra reduction identities in order to calculate our birdtrack diagrams.

Because of the need for extra reduction identities for the exceptional Lie groups we found ourselves unable to solve the rest of the exceptional algebras using the method described above. Therefore for these algebras we represented their adjoint representation using their sub-algebras which were all made up of the Lie algebras we had already solved. Using the Jacobi identity eq. (5.61), we could verify that we had found the correct generators, and along with introducing spinors in birdtrack, the rest of the exceptional Lie algebras (e_6 , e_7 , e_8 and f_4) were solved.

So what is the next step from here? Well a couple of questions immediately arises, where the first one is, that since these calculations have all been performed exclusively on tree amplitudes what happens when we want to calculate colour factors to any loop order? One of the issues that comes to mind here is that in order to calculate the colour factors for the exceptional algebras e_6 , e_7 and f_4 we had to take advantage of them being expressed in terms of the adjoint representation. This was ensured by transforming the generators in colour ordered amplitudes into the structure constants in the DDM basis, but for the general loop level case it is not certain that one can express the colour factors in the adjoint representation. Therefore it could be interesting to investigate if it is possible to express the colour factors to all loop orders in terms of their adjoint representation (notice that since the adjoint representation of e_8 is the same as the fundamental representation of e_8 , we can still calculate colour factors for this algebra), or if it is possible to find a way to express the fundamental representation in a way where we can use birdtrack to calculate diagrams.

Another interesting question to arise from this is the appearance of the symmetric group in the Zeppenfeld basis. An open question here is why this group appears and if it is possible to find new relations or symmetries based on this appearance. One possible way of attacking this problem could be by using the Brauer algebra [16]. Lastly we can ask why the birdtrack method fails by a factor of 2 to correctly calculate the adjoint quadratic Casimir for $so(3)$, and $so(4)$. Even though this might not seem of the utmost importance it is nevertheless worrying that the birdtrack method gives the wrong answer here, when it seem to give the correct answer at all other instances, so maybe there is something to uncover here. But all these things aside lets end it here and rejoice in the fact that in the end we did catch 'em all.

References

- [1] V.E. Barnes et al. “Observation of a Hyperon with Strangeness Minus Three”. In: *Phys. Rev. Lett.* 12 (1964), pp. 204–206. DOI: 10.1103/PhysRevLett.12.204.
- [2] E. Cremmer and B. Julia. “The N=8 Supergravity Theory. 1. The Lagrangian”. In: *Phys. Lett. B* 80 (1978). Ed. by A. Salam and E. Sezgin, p. 48. DOI: 10.1016/0370-2693(78)90303-9.
- [3] Predrag Cvitanovic. *Group theory: Birdtracks, Lie’s and exceptional groups*. 2008.
- [4] Vittorio Del Duca, Lance J. Dixon, and Fabio Maltoni. “New color decompositions for gauge amplitudes at tree and loop level”. In: *Nucl. Phys. B* 571 (2000), pp. 51–70. DOI: 10.1016/S0550-3213(99)00809-3. arXiv: hep-ph/9910563.
- [5] P. Di Francesco, P. Mathieu, and D. Senechal. *Conformal Field Theory*. Graduate Texts in Contemporary Physics. New York: Springer-Verlag, 1997. ISBN: 978-0-387-94785-3, 978-1-4612-7475-9. DOI: 10.1007/978-1-4612-2256-9.
- [6] Henriette Elvang and Yu-tin Huang. “Scattering Amplitudes”. In: (Aug. 2013). arXiv: 1308.1697 [hep-th].
- [7] Robert Feger and Thomas W. Kephart. “LieART—A Mathematica application for Lie algebras and representation theory”. In: *Comput. Phys. Commun.* 192 (2015), pp. 166–195. DOI: 10.1016/j.cpc.2014.12.023. arXiv: 1206.6379 [math-ph].
- [8] J. Fuchs and C. Schweigert. *Symmetries, Lie algebras and representations: A graduate course for physicists*. Cambridge University Press, Oct. 2003. ISBN: 978-0-521-54119-0.
- [9] Murray Gell-Mann. “A Schematic Model of Baryons and Mesons”. In: *Phys. Lett.* 8 (1964), pp. 214–215. DOI: 10.1016/S0031-9163(64)92001-3.
- [10] Murray Gell-Mann. “The Eightfold Way: A Theory of strong interaction symmetry”. In: (Mar. 1961). DOI: 10.2172/4008239.
- [11] Ilya G Kaplan. *The Pauli exclusion principle: origin, verifications and applications*. See appendix B. Hoboken, NJ: John Wiley & Sons, 2017. URL: <https://cds.cern.ch/record/2199117>.
- [12] Killing. “Die Zusammensetzung der stetigen endlichen Transformationsgruppen. Vierter Theil (Schluss).” In: *Mathematische Annalen* 36 (1890), pp. 161–189. URL: <http://eudml.org/doc/157490>.
- [13] Tetsuji Kimura. “Explicit Description of the Zassenhaus Formula”. In: *PTEP* 2017.4 (2017), 041A03. DOI: 10.1093/ptep/ptx044. arXiv: 1702.04681 [math-ph].
- [14] Alexander Kirillov Jr. *An introduction to Lie groups and Lie algebras*. Vol. 113. Cambridge University Press, 2008.
- [15] Ronald Kleiss and Hans Kuijf. “Multi - Gluon Cross-sections and Five Jet Production at Hadron Colliders”. In: *Nucl. Phys. B* 312 (1989), pp. 616–644. DOI: 10.1016/0550-3213(89)90574-9.
- [16] Steffen König and Changchang Xi. “A Characteristic Free Approach to Brauer Algebras”. In: *Transactions of the American Mathematical Society* 353.4 (2001), pp. 1489–1505. ISSN: 00029947. URL: <http://www.jstor.org/stable/221866>.

-
- [17] Takahiro Morimoto et al. “ Z_3 symmetry-protected topological phases in the SU(3) AKLT model”. In: *Physical Review B* 90 (Sept. 2014), p. 235111. DOI: 10.1103/PhysRevB.90.235111.
- [18] Matthew D. Schwartz. *Quantum Field Theory and the Standard Model*. Cambridge University Press, Mar. 2014. ISBN: 978-1-107-03473-0, 978-1-107-03473-0.
- [19] Assaf Shomer. “A Pedagogical explanation for the non-renormalizability of gravity”. In: (Sept. 2007). arXiv: 0709.3555 [hep-th].
- [20] R. Slansky. “Group Theory for Unified Model Building”. In: *Phys. Rept.* 79 (1981), pp. 1–128. DOI: 10.1016/0370-1573(81)90092-2.
- [21] M. Srednicki. *Quantum field theory*. Cambridge University Press, Jan. 2007. ISBN: 978-0-521-86449-7, 978-0-511-26720-8.
- [22] Steven Weinberg. “A Model of Leptons”. In: *Phys. Rev. Lett.* 19 (1967), pp. 1264–1266. DOI: 10.1103/PhysRevLett.19.1264.
- [23] Kenneth G. Wilson. “Confinement of Quarks”. In: (Feb. 1974). Ed. by J.C. Taylor, pp. 45–59. DOI: 10.1103/PhysRevD.10.2445.
- [24] Edward Witten. “Quantum Field Theory and the Jones Polynomial”. In: *Commun. Math. Phys.* 121 (1989). Ed. by Asoke N. Mitra, pp. 351–399. DOI: 10.1007/BF01217730.
- [25] Chen-Ning Yang and Robert L. Mills. “Conservation of Isotopic Spin and Isotopic Gauge Invariance”. In: *Phys. Rev.* 96 (1954). Ed. by Jong-Ping Hsu and D. Fine, pp. 191–195. DOI: 10.1103/PhysRev.96.191.
- [26] A. Zee. *Group Theory in a Nutshell for Physicists*. USA: Princeton University Press, 2016. ISBN: 978-0-691-16269-0, 978-0-691-16269-0, 978-1-4008-8118-5.
- [27] D. Zeppenfeld. “Diagonalization of Color Factors”. In: *Int. J. Mod. Phys. A* 3 (1988), pp. 2175–2179. DOI: 10.1142/S0217751X88000916.
- [28] Jean-Bernard Zuber. “Generalized Dynkin diagrams and root systems and their folding”. In: *Taniguchi Symposium on Topological Field Theory, Primitive Forms and Related Topics*. July 1997, pp. 453–493. arXiv: hep-th/9707046.

Appendix - Python Code

This is the Python code used to calculate the colour factors shown in tables: (6.1), (6.2), (6.3) and (6.4). It draws the diagrams and calculates them using the Birdtrack method.

```
import networkx as nx
import time
import matplotlib as plt
import numpy as np
import itertools
import multiprocessing as mp
from sympy import Symbol
from fractions import Fraction

# To use the code to calculate the value of a diagram use the following functions:

# For so_n, sp_n or su_n use the function - complete_diagram(n_p,n_g,group,S_n)
# where n_p is the number of particles, n_g is the "n" in su_n (and also for so_n and sp_n)
# group can be either 'so_n', 'sp_n' or 'su_n'
# S_n is the permutation of the diagram, eq. for 5 particles we might have [0,1,2], which
# is equivalent to the identity or [2,0,1] which is some permutation. Since we are
# working
# with S_(n-2) there are 2 fewer numbers than there are particles.

def make_structure_constant(A,B):
    # This function creates each structure-constant f^{abc}, as a closed circle of 8
    # connected nodes.
    # The node numbers that connect to other nodes is [1,4,6].

    # the variable A is either 'upper' or 'lower' which refers to the row it is eventually
    # going to belong to. B a number that distinguishes the different structure
    # constants in each row.

    G=nx.Graph()
    nodes = [A+'_'+str(B)+'_'+str(i) for i in range(1,9)]
    for n in nodes:
        G.add_node(n)
    for l, r in zip(nodes, nodes[1:]+[nodes[0]]):
        G.add_edge(l,r,length=1)
    return G

def row_of_structure_constants(L,position):
    # This function creates a list containing all the structure constants that exists in a
    # row. A row is the same as the expression T^{\pi} in Zeppenfelds paper.
    # ie. a product of structure constants in a certain permutation.

    # The variable 'L' is the number of structure constants in the row, and the variable
    # 'position' indicates if it is an upper or lower row.
    listen = np.zeros(L,dtype=object)
```

```

for i in range(L):
    listen[i] = make_structure_constant(position,i)
return listen

def connect_structure_constants_row(listen,parity,position):
    # This function connects all the structure constants in a row. A row is the same as
    # the expression  $T^{\{\pi\}}$  in Zeppenfelds paper.
    # The variable 'Listen' is a list of structure constants to be conected in a row.
    # The variable 'parity' indicates which of the two terms in the expansion of the
    # structure constants  $f^{\{abc\}} = \text{tr}(t^a T^b T^c) - \text{tr}(t^a T^c T^b)$  the is used in this
    # example.
    # The variable position indicates weather it is an upper or lower row.

    # in this part of the code the diagrams are merged together into a single diagram.
    F = nx.compose(listen[0],listen[1])
    for i in range(len(listen)-2):
        F = nx.compose(F,listen[i+2])

    # In this part of the code the structure constants are connected based on their parity.
    for i in range(len(listen)-1):
        if parity[i] == 1:
            if parity[i+1] == 1:
                F.add_edge(position+'_'+str(i)+'_6',position+'_'+str(i+1)+'_4',length=1)
            if parity[i+1] == -1:
                F.add_edge(position+'_'+str(i)+'_6',position+'_'+str(i+1)+'_6',length=1)
        if parity[i] == -1:
            if parity[i+1] == 1:
                F.add_edge(position+'_'+str(i)+'_4',position+'_'+str(i+1)+'_4',length=1)
            if parity[i+1] == -1:
                F.add_edge(position+'_'+str(i)+'_4',position+'_'+str(i+1)+'_6',length=1)
    return F

def create_diagrams(n, S_n):
    # This function creates the diagrams that will occure after all the structure
    # constants have been expanded withe relation  $f^{\{abc\}} = \text{tr}(t^a T^b T^c) - \text{tr}(t^a T^c T^b)$ .

    # The variable 'n' is the number of particles.
    # The variable 'S_n' is the specific permutation of the product of structure constants
    #  $T^{\{\pi\}}$  in Zeppenfelds paper.

    # Here the parity for the upper and lower row is created.
    combinations_upper = np.array(list(itertools.product([1, -1], repeat=n-2)))
    combinations_lower = np.array(list(itertools.product([1, -1], repeat=n-2)))

    # Here the array containing the diagrams is prelocated
    diagrams = np.zeros((2**(2*(n-2)),2),dtype=object)
    k=0

    for i, j in itertools.product(combinations_upper, combinations_lower): # This loop is
        # over the permutations in both the upper and lower row.

```

```

upper_diagram =
    connect_structure_constants_row(row_of_structure_constants(n-2,'upper'),i,'upper')
lower_diagram =
    connect_structure_constants_row(row_of_structure_constants(n-2,'lower'),j,'lower')
diagram = nx.compose(upper_diagram,lower_diagram)

# Here the rows are connected with respect to the specific permutation in the list
  S_n
for uy in range(n-2):
    diagram.add_edge('lower_'+str(uy)+'_1','upper_'+str(S_n[uy])+'_1')

# The first structure constant in the lower row has two connections to the upper
  row. This block of code deals with the second connection.
if i[0] == 1:
    if j[0] == 1:
        diagram.add_edge('upper_0_4','lower_0_4',length=1)
    if j[0] == -1:
        diagram.add_edge('upper_0_4','lower_0_6',length=1)
if i[0] == -1:
    if j[0] == 1:
        diagram.add_edge('upper_0_6','lower_0_4',length=1)
    if j[0] == -1:
        diagram.add_edge('upper_0_6','lower_0_6',length=1)

# The last structure constant in the lower row has two connections to the upper
  row. This block of code deals with the second connection.
if i[-1] == 1:
    if j[-1] == 1:
        diagram.add_edge('upper_'+str(n-3)+'_6','lower_'+str(n-3)+'_6',length=1)
    if j[-1] == -1:
        diagram.add_edge('upper_'+str(n-3)+'_6','lower_'+str(n-3)+'_4',length=1)
if i[-1] == -1:
    if j[-1] == 1:
        diagram.add_edge('upper_'+str(n-3)+'_4','lower_'+str(n-3)+'_6',length=1)
    if j[-1] == -1:
        diagram.add_edge('upper_'+str(n-3)+'_4','lower_'+str(n-3)+'_4',length=1)

# After each diagram is created it is saved to a list, along with the product of
  the parity (to keep track of the sign)
diagrams[k,0] = diagram
diagrams[k,1] = np.product(i)*np.product(j)
k+=1
return diagrams

def calculate_diagram(diagram,n_p,group,n_g):
    # This script takes in a single diagram and expand all the adjoint contractions with
      the adjoint projection operator. Then it calculates the value of all those
      expanded diagrams.

    # The variable 'diagram' is a diagram that needs to be expanded.

```

```

# The variable 'n_p' is the number of particles.
# The variable 'group' indicates the Lie group we are dealing with here (eq. su(n))
# The variable 'n_g' is the n in su(n). It is the value of each closed loop in a
  birdtrack diagram.

# Here we initialize n_po which is the number of terms in the adjoint projection
  operator
if group=='su_n':
  n_po=2
elif group=='so_n':
  n_po=2
elif group=='sp_n':
  n_po=2

# Here the different arrays are preallocated. Values are the final values of the
  diagrams, and permutations tells with term in the adjoint projection operator to
  use thi stime.
values = np.zeros(n_po**(n_p+2*(n_p-3)),dtype=object)
permutations = list(itertools.product(np.linspace(1,n_po,n_po).astype(int),
  repeat=n_p+2*(n_p-3)))
diagram_list = np.zeros(len(values),dtype = object)

# The diagrams are copied and duplicated before they are expanded.
for i in range(len(diagram_list)):
  diagram_list[i] = diagram.copy()

# Here each adjoint line in the diagrams is expanded with respect to their
  permutation, and then value of each diagram is calculated.
for i in range(len(values)):
  temp_diagram = deformation(n_p,permutations[i],diagram_list[i],group)
  values[i] = value_diagram(n_g,permutations[i],temp_diagram,group)

# The value of each diagram is summed together to give the final value.
value = sum(values)

# Since the groups 'so(n)' and 'sp(n)' have an overall factor of 1/2 in their adjoint
  projection operator, a quick tjeck is performed to see if the overall value should
  be multiplied with this factor.
if group in ('so_n','sp_n'):
  value = value*((1/2)**(n_p+2*(n_p-3)))
return value

def deformation(n_p,permutation,diagram, group):

# This function is the first of two functions that expands all adjoint lines in a
  diagram.

# the variable 'n_p' is the number of particles.

```

```

# The variable 'permutation' is a list of numbers (one for each adjoint line) that
# indicates which term in the adjoint projection operator to use for each adjoint
# line.
# The variable 'diagram' is the diagram to be deformed.
# the variable 'group' indicates the Lie group we are dealing with here.

k = 0
for i in range(n_p-2):
    # The loop is over the number of structure constants in a row. Since the number of
    # structure constants is the same in the upper and lower row, the upper and lower
    # row are looped over simultaneously. For each adjoint line a specific ordering of
    # the of the two nodes is determined adn then is sent through to the function
    # 'deform_group' which determines how to connect the neighbors to the two
    # forementioned nodes.

    bottom = 'lower_'+str(i)+'_1'
    neighbors = list(diagram.neighbors(bottom))
    for j in range(len(neighbors)):
        if neighbors[j][0:5]=='upper':
            top = neighbors[j]

    diagram = deform_group(top,bottom,permutation[k],diagram,group)
    k+=1
    try:
        bottom = 'lower_'+str(i)+'_4'
        neighbors = list(diagram.neighbors(bottom))
        for j in range(len(neighbors)):
            if neighbors[j][0:5]=='upper':
                top = neighbors[j]
            if int(neighbors[j][6:7])!=int(bottom[6:7]):
                top = neighbors[j]
        diagram = deform_group(top,bottom,permutation[k],diagram,group)
        k+=1
    except:
        1+1
    try:
        bottom = 'lower_'+str(i)+'_6'
        neighbors = list(diagram.neighbors(bottom))
        for j in range(len(neighbors)):
            if neighbors[j][0:5]=='upper':
                top = neighbors[j]
            if int(neighbors[j][6:7])!=int(bottom[6:7]):
                top = neighbors[j]
        diagram = deform_group(top,bottom,permutation[k],diagram,group)
        k+=1
    except:
        1+1
    try:
        top = 'upper_'+str(i)+'_4'

```

```

        neighbors = list(diagram.neighbors(top))
        for j in range(len(neighbors)):
            if neighbors[j][0:5]=='lower':
                bottom = neighbors[j]
            if int(neighbors[j][6:7])!=int(top[6:7]):
                bottom = neighbors[j]
        diagram = deform_group(top,bottom,permutation[k],diagram,group)
        k+=1
    except:
        1+1
    try:
        top = 'upper_'+str(i)+'_6'
        neighbors = list(diagram.neighbors(top))
        for j in range(len(neighbors)):
            if neighbors[j][0:5]=='lower':
                bottom = neighbors[j]
            if int(neighbors[j][6:7])!=int(top[6:7]):
                bottom = neighbors[j]
        diagram = deform_group(top,bottom,permutation[k],diagram,group)
        k+=1
    except:
        1+1

# after a diagram has been deformed it s returned.
return diagram

def deform_group(top,bottom,permutation,diagram,group):

# This function determines the specific ordering of the nodes to be connected when the
# adjoint line between the 'top' and 'bottom' nodes is expanded.
# The variable 'top' is a node in the diagram that constitutes one part of the adjoint
# line to be expanded.
# The variable 'bottom' is a node in the diagram that constitutes the other part of
# the adjoint line to be expanded.
# The variable 'permutation' is the specific permutation for this adjoint line, that
# determines which term in the adjoint projection operator to use.
# The variable 'diagram' is the diagram from which the adjoint line will be expanded.
# The variable 'group' indicates the Lie group we are dealing with here.

# To start with the neighbors of the top and bottom node is found, these neighbors are
# the ones that should be connected to each
# other, in a way that depends on the group.
list1 = list(diagram.neighbors(bottom))
list2 = list(diagram.neighbors(top))
top = top.split('_')
bottom = bottom.split('_')

# Here the bottom node is deleted from the list containing the neighbors of the top
# node, and vice versa for the top node in the list containing
# the neighbors of the bottom node.

```

```
for i in range(len(list2)):
    if '_' .join(bottom) == list2[i]:
        delete1 = i
    if '_' .join(top) == list1[i]:
        delete2 = i
try:
    del list1[delete2]
except:
    pass
try:
    del list2[delete1]
except:
    pass
for i in range(len(list2)):
    list1[i] = list1[i].split('_')
    list2[i] = list2[i].split('_')

# Here the specific ordering of the nodes is determined. I found it difficult to come
# up with a compact way of doing it, so it is basically just checking
# all the possible combinations.
if top[0]=='upper' and bottom[0]=='lower':
    if int(list1[0][2])>int(list1[1][2]):
        A = list1[0]
        B = list1[1]
    else:
        B = list1[0]
        A = list1[1]
    if int(list2[0][2])>int(list2[1][2]):
        C = list2[0]
        D = list2[1]
    else:
        D = list2[0]
        C = list2[1]
if top[0]=='lower' and bottom[0]=='upper':
    if int(list1[0][2])>int(list1[1][2]):
        A = list2[0]
        B = list2[1]
    else:
        B = list2[0]
        A = list2[1]
    if int(list2[0][2])>int(list2[1][2]):
        C = list1[0]
        D = list1[1]
    else:
        D = list1[0]
        C = list1[1]

if top[0]=='upper' and bottom[0]=='upper':
    if int(top[1])>int(bottom[1]):
        if int(list1[0][2])>int(list1[1][2]):
```

```
        A = list1[1]
        B = list1[0]
    else:
        B = list1[1]
        A = list1[0]
    if int(list2[0][2])>int(list2[1][2]):
        C = list2[0]
        D = list2[1]
    else:
        D = list2[0]
        C = list2[1]
elif int(top[1])<int(bottom[1]):
    if int(list1[0][2])>int(list1[1][2]):
        A = list2[1]
        B = list2[0]
    else:
        B = list2[1]
        A = list2[0]
    if int(list2[0][2])>int(list2[1][2]):
        C = list1[0]
        D = list1[1]
    else:
        D = list1[0]
        C = list1[1]

if top[0]=='lower' and bottom[0]=='lower':
    if int(top[1])>int(bottom[1]):
        if int(list1[0][2])>int(list1[1][2]):
            A = list1[0]
            B = list1[1]
        else:
            B = list1[0]
            A = list1[1]
        if int(list2[0][2])>int(list2[1][2]):
            C = list2[1]
            D = list2[0]
        else:
            D = list2[1]
            C = list2[0]
    elif int(top[1])<int(bottom[1]):
        if int(list1[0][2])>int(list1[1][2]):
            A = list2[0]
            B = list2[1]
        else:
            B = list2[0]
            A = list2[1]
        if int(list2[0][2])>int(list2[1][2]):
            C = list1[1]
            D = list1[0]
        else:
```

```

        D = list1[1]
        C = list1[0]

# Here the expression used to determine the ordering is combined again, so that they
# represents nodes in the diagram.
A = '_' .join(A)
B = '_' .join(B)
C = '_' .join(C)
D = '_' .join(D)
top = '_' .join(top)
bottom = '_' .join(bottom)
diagram.remove_node(top)
diagram.remove_node(bottom)

# Finally the nodes are connected based on the group used.
if group=='su_n':
    if permutation == 1:
        diagram.add_edges_from([(A,C), (B,D)])
    if permutation == 2:
        diagram.add_edges_from([(A,B), (C,D)])

if group == 'so_n':
    if permutation == 1:
        diagram.add_edges_from([(A,C), (B,D)])
    if permutation == 2:
        diagram.add_edges_from([(A,D), (B,C)])

if group == 'sp_n':
    if permutation == 1:
        diagram.add_edges_from([(A,C), (B,D)])
    if permutation == 2:
        diagram.add_edges_from([(A,D), (B,C)])

return diagram

def value_diagram(n_g,permutation,temp_diagram,group):

# Here the value of a diagram is calculated.
# The variable 'n_g' is the value of a closed fundamental line.
# The variable 'permutation' is a list of numbers containing the specific permutation
# for each expanded adjoint line.
# The variable 'temp_diagram' is the diagram for which the value is to be calculated.
# The variable 'group' indicates the Lie group we are dealing with here.

# This function uses the fact that after the diagram has been deformed, there are only
# closed loops left, so it just have to count how many
# closed loops there are. To do this networkx has a function (nx.cycle_basis) that
# gives exactly this.
# exponent1 is how many times n_g is multiplied with itself.
# since the adjoint projection operator for each group has a negative term exponent 2

```

```

    is the total number of negative terms in this diagram.

if group == 'su_n':
    exponent1 = len(nx.cycle_basis(temp_diagram))
    exponent2 = sum(np.array(permutation)==2)
    return (n_g**(exponent1)/(n_g**(exponent2)))*((-1)**(exponent2))

if group == 'so_n':
    exponent1 = len(nx.cycle_basis(temp_diagram))
    exponent2 = sum(np.array(permutation)==2)
    return (n_g**(exponent1))*((-1)**(exponent2))

if group == 'sp_n':
    exponent1 = len(nx.cycle_basis(temp_diagram))
    exponent2 = sum(np.array(permutation)==2)
    return (n_g**(exponent1))*((-1)**(exponent2))*((-1)**(exponent1))

def complete_diagram(n_p,n_g,group,S_n):

    # This function calculates a whole diagram.

    # The variable 'n_p' is the number of particles.
    # The variable 'n_g' is the value of a closed fundamental line.
    # The variable 'group' indicates the Lie group we are dealing with here.
    # The variable 'S_n' is the specific permutation of the product of structure constants
    #  $T^{\pi}$  in Zeppenfelds paper.

    # Here all the diagrams are created
    diagrams = create_diagrams(n_p, S_n)
    result = 0

    # Here all the diagrams are looped over and their value is calculated
    for i in range(len(diagrams)):
        asdf = calculate_diagram(diagrams[i][0],n_p,group,n_g)
        result = result+asdf*diagrams[i][1]
        print(i)
    return result

def intermediate_calculation(n_p,n_g,group,little_n,fundamental,j):

    # This function is used in combination with the irreducible representations of the
    # permutation group. It is not used to calculate an diagram, but instead to
    # calculate the C's from Zeppenfelds paper.

    S_n = np.array(np.transpose(np.matmul(np.matrix(fundamental[j][0]),
        np.transpose(np.matrix([np.linspace(1,little_n,little_n).astype(int)]))))[0].astype(int)-1
    return fundamental[j][1]*complete_diagram(n_p,n_g,group,S_n)

```
