How Effective Is Renormalization

Dirk van Buul, s4500385

Supervisor: Prof. Ronald Kleiss (THEP)

Radboud University, Nijmegen

20 June 2018

Contents

1 Introduction

My research was done in the context of Quantum Field Theory. The objects of interest within this theory were the connected Green's functions, in particular the numerical coefficients thereof. The first objective was to actually find these functions, by solving the Schwinger-Dyson equation. It is the connected Green's functions, or rather the parameters in these functions, that I wanted to renormalize. After this process, I wanted to see how the newly found connected Green's functions behave, or rather improve, after renormalization. The concepts mentioned so far will be explained, but the important thing to note for now is that, before renormalization, the numerical coefficients in the connected Green's functions behave as a divergent series in $\hbar$. I want to know to what extent, if any, the renormalization process improves this.

I want to start with a brief outline of this thesis. I will start off with the necessary theory to explain what has been done and why. This is found in chapter 2. I want to make it clear that this is not my own work, nor is it part of my actual research. The theory is based on [1]. I have simply used the definitions described there to perform this research. After chapter 2, I will explain the content of my own research.

2 Theory

Let us begin by taking a look at the basics from Quantum Field Theory (QFT). In [1], the idea of QFT is built up from the starting point of zero dimensions. Here the basic principles are discussed, like the path integral, the Schwinger-Dyson equation (SDe) and the (connected) Green's functions. Also, Feynman diagrams are linked to the SDe. Additionally, the principles of the procedure of renormalization are demonstrated and finally all these notions are extended to our four-dimensional Minkowski space. We will however, restrict ourselves to zero dimensions. Also Feynman diagrams will not be covered. The essence of my research is the effect of renormalization, so I will explain the necessary parts of the theory up and until that point.

2.1 Quantum Field Theory

In zero dimensions, a quantum field φ is simply an assignment of a single number. So φ can, in principle, take the value of any real number. The most that can be known about φ, since it is a stochastic variable, is the probability density. This probability density contains the action, $S(\varphi)$, and since that is a function of φ, it determines the shape of the probability density, and therefore, one could say the action defines the theory. We will get back to this action shortly. Again, since φ is a random variable, the most that can be known about it, together with the probability density, is the collection of its moments, called the Green's functions. They are defined as follows:

$$G_n \equiv \langle \varphi^n \rangle \equiv N \int \exp\left(-\frac{1}{\hbar} S(\varphi)\right) \varphi^n d\varphi \quad , \quad n = 0, 1, 2, 3, ... \quad . \tag{1}$$

This notation means integration from $-\infty$ to ∞. Here N is just a normalization factor. The factor $\hbar$ is assigned to every closed loop in a Feynman diagram, and will indeed turn out to be the reduced Planck constant in a four-dimensional universe. However, this goes beyond the purpose of the thesis, so I will not further explain it. So the Green's functions contain all the information about the probability density, where by definition we have $G_0 = 1$. It is convenient to express the collection of Green's functions via a generating function. This generating function is also called the path integral and it is defined as:

$$Z(J) = \sum_{n\geq 0} \frac{1}{n!} \left(\frac{J}{\hbar}\right)^n G_n = N \int \exp\left(-\frac{1}{\hbar}(S(\varphi) - J\varphi)\right) d\varphi \quad . \tag{2}$$

In this equation, J is called a source. What this source is exactly, and why it is called as such, also has to do with Feynman diagrams, so it will not further be explained. We will simply use it as a tool. The generating function $Z(J)$ contains all Green's function and therefore all information available about the quantum field φ. Of course, that same information is also contained in the logarithm of the generating function. So we get:

$$W(J) \equiv \ln Z(J) = \sum_{n\geq 1} \frac{1}{n!} \left(\frac{J}{\hbar}\right)^n C_n \quad . \tag{3}$$

The elements C_n in this equation are called connected Green's function and can be regarded as cumulants of the probability density, where always $C_0 = 0$. As mentioned earlier, these functions are the objects of interest for this research, because it is these functions, or rather the parameters inside them, that will be renormalized. Finally the same information about the probability density of the quantum field is contained in the so-called field function, defined as follows:

$$\phi(J) \equiv \hbar \frac{\partial}{\partial J} \ln Z(J) = \sum_{n\geq 0} \frac{1}{n!} \left(\frac{J}{\hbar}\right)^n C_{n+1} \quad . \tag{4}$$

This field function, $\phi(J)$ can be regarded as the expectation value of the quantum field φ, in the presence of sources. Combining all definitions from above, it can be easily seen that the field function can be written as:

$$\begin{aligned} \phi(J) = &\left[\int \exp\left(-\frac{1}{\hbar}(S(\varphi) - J\varphi)\right) \varphi d\varphi\right] \\ &\cdot \left[\int \exp\left(-\frac{1}{\hbar}(S(\varphi) - J\varphi)\right) d\varphi\right]^{-1} \quad . \end{aligned} \tag{5}$$

From this way of writing the field function, it is clear why it was stated that the action defines the theory in a way. Because remember, the quantum field φ was a random number and the only thing that can be known about it, is its probability density (determined by the action) and with that of course, the expectation value.

Now let us get back to the action, $S(\varphi)$. The theory in which I have done my research is called the φ^4 theory, and here the action has the following form:

$$S(\varphi) = \frac{1}{2}\mu\varphi^2 + \frac{1}{4!}\lambda_4\varphi^4 \quad . \tag{6}$$

This theory is the simplest theory apart from the free theory, where the action contains only the first term of equation (6), the quadratic (kinetic) term in φ. The φ^4 theory is the simplest (well-defined) interacting theory, meaning that we can see λ_4 as the coupling constant. Once the path integral $Z(J)$ is known, all information about the quantum field φ can be extracted/derived. Calculating the path integral in the free theory is quite easy to do, however, in the φ^4 theory, this becomes a much more difficult task. That is why we need to resort to perturbation theory. This can be done if it is assumed that the second term in equation (6) is a small perturbation of the free theory, in other words, if it is assumed that λ_4 is a small number. By the way, both μ and λ_4 are assumed to be real and positive (or at least non-negative for λ_4) numbers.

I will not go through the derivations done in [1], to get to the relations that follow. For the perturbation scheme, we can write for the Green's functions:

$$\begin{aligned} G_{2n} &= \frac{H_{2n}}{H_0} \quad , \\ H_{2n} &= \frac{\hbar^n}{\mu^n} \sum_{k\geq 0} \frac{(4k+n)!}{2^{2k+n}(2k+n)!k!} \left(-\frac{\lambda_4\hbar}{24\mu^2}\right)^k \quad , \quad n = 0, 1, 2, 3, ... \quad . \end{aligned} \tag{7}$$

In this theory, φ^4 theory, both G_n and C_n are zero for odd n. For convenience later on, we can with a slightly different definition of H (and in integral form), write equation (7) as follows:

$$\begin{aligned} G_n &= \frac{H_n}{H_0} \quad , \\ H_n &= \int \exp\left(-\frac{1}{\hbar}\left(\frac{\mu}{2}\varphi^2 + \frac{\lambda_4}{24}\varphi^4\right)\right)\varphi^n d\varphi \quad , \quad n = 0, 2, 4, ... \quad . \end{aligned} \tag{8}$$

So in general, the path integral, and therefore the field function, are quite complicated functions of J, but there is a way of finding an equation to

fully describe them, namely the Schwinger-Dyson equation (SDe). We will focus only on the SDe for the field function, where we need the solution to correspond to the perturbative expansion. Again, I will not include the derivation done in [1], I will simply quote the result. For the φ^4 theory, the SDe reads:

$$\phi(J) = \frac{J}{\mu} - \frac{\lambda_4}{6\mu}\left(\phi(J)^3 + 3\hbar\phi(J)\frac{\partial}{\partial J}\phi(J) + \hbar^2\frac{\partial^2}{\partial J^2}\phi(J)\right) \quad . \tag{9}$$

This equation, or rather the solving of this equation, is the starting point of this research. Remember that the connected Green's functions were the object of interest. So by solving the SDe, and using equation (4), the connected Green's functions can be found.

2.2 Why Renormalization

I do not think it is useful to explain what renormalization actually entails, without showing an example. Although it might be handy to mention in a few words what it means. Renormalization is the order-by-order (in perturbation theory) improvement of the parameters. In this paragraph it will be briefly mentioned why renormalization is necessary, but as we go along, the principle itself will become clearer. Chapter 3.2 will then shed further light on the principle of renormalization, as that chapter serves as an example, as well as a summary of the methods used in this research. Let me start off my explanation as to why we are doing renormalization with a quote taken from [1]. After that, I will further comment on this quote.

"In particular, it cannot be stressed often enough that the renormalization procedure is necessary simply because one does perturbation theory, not because loop corrections may contain infinities."

So it is clear from this quote, that perturbation theory is the reason behind the necessity for renormalization. Later on this will be discussed in more detail, but I do want to briefly explain why this makes sense. We are doing perturbation theory, so from equation (7) it can be seen that we only know the Green's functions, and therefore the connected Green's functions, as a truncated series in $\hbar$. If one wants to know the values of the parameters μ and λ_4, one can measure two of those connected Green's functions. And since they

contain the parameters, the values of the parameters can now be calculated. Two measurements is important here, as we need two coupled equations with two unknowns, namely the two parameters. However, it is really important to consider to which order one wants to calculate the parameters. Because once the values of the parameters are found, one can use them to predict the values of other connected Green's functions. But it is not possible to predict the other ones to an arbitrary order. In fact, one can only make an accurate prediction to the same order as to which the parameters have been found.

3 How to Renormalize

The following section is devoted to showing the way the renormalization process is done. It is basically a summary of this research and the methods used.

3.1 Schwinger-Dyson Equation

As stated before, it all starts with solving the Schwinger-Dyson equation, in order to find the field function $\phi(J)$. From equation (4), it is clear that the field function is a polynomial in J, the coefficients of which are the connected Green's functions C_n. So we can write:

$$\phi(J) = C_2\frac{J}{\hbar} + C_4\frac{J^3}{3!\hbar^3} + C_6\frac{J^5}{5!\hbar^5} + ... \quad . \tag{10}$$

The field function is known, once the SDe is solved. With that, the connected Green's functions, the objects of interest, can be extracted. Solving the SDe is done by iterating the assignment (9), using computer software Maple. As a starting point, one needs to set $\phi(J) = 0$. From there, the iteration will lead to the correct form of the field function.

3.1.1 Iteration

There are different ways to iterate the Schwinger-Dyson equation. I will provide part of the source code I made in Maple, showing both these techniques. On the one hand, one can iterate using an auxiliary variable u. In principle, this gives a power series in u, truncated at a certain value of u. This makes iteration easier (although not flawless as explained later), and

after setting $u = 1$ in the output, one arrives at the desired expression for ϕ. Remember that the field function is a polynomial in J and that the connected Green's functions are polynomials in $\hbar$, meaning we are considering two power series simultaneously. In the source code in appendix A, it is shown where in the SDe this auxiliary variable is inserted. The important thing to note, is that the auxiliary variable u, was inserted in the SDe in such a way, that this method of iteration results in a power series where the powers of $\hbar$ are the same as those of J, again, see the appendix. I shall call this method the 'triangle'-method, see figure (1).

On the other hand, one can also directly truncate the series manually, without using an auxiliary variable, at both $\hbar$ and J, at different orders, if one so desires. For reasons that will become apparent later, this method would seem desirable, as we are looking for series containing as much powers of $\hbar$ as possible, but we only need powers of J up to an order of around twelve. This method will be called the 'rectangular'-method, see also the figure below.

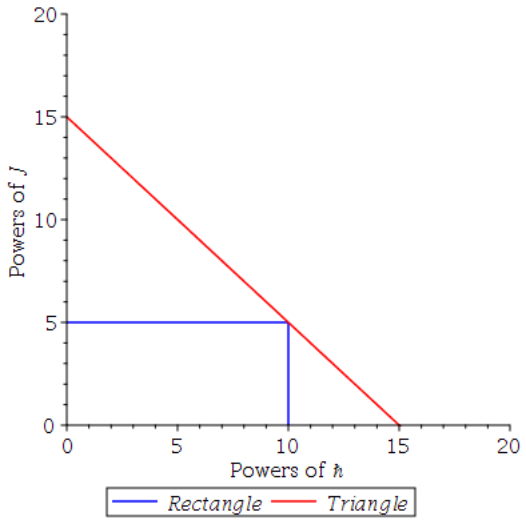

Figure 1: This diagram schematically shows the idea of truncating both power series (in J and $\hbar$) at once.

The last method I explained, works quite a lot faster than the first one, as it throws away the redundant powers of J while calculating. However, it turns out that, with the same number of iterations, the same highest power

of $\hbar$, but a different highest power in J, the two methods do not coincide. In fact, when one increases the value of the highest power of J in the rectangle-method, one finds that the result approaches that of the triangle-method. This means that one cannot simply throw away redundant powers J, mid-calculation, with impunity. The higher powers are actually required for a correct result and therefore the triangle-method is the way to go.
This actually makes sense, when looking at the Schwinger-Dyson equation in relation (9). Consider the fact the we are basically considering two series at once, namely a series in J an a series in $\hbar$. Since the SDe mixes $\hbar$ with derivatives with respect to J, one cannot simply truncate the series at different orders.

So, unfortunately, we are stuck with the first method, which takes a lot of time. In fact, each step just gets slower and slower. And when one finally arrives at the desired result, one can basically discard all terms with powers of J higher than around eleven or twelve. That is because we are only interested in the first couple of connected Green's functions. The order of the connected Green's functions corresponds to the number of external lines of a Feynman diagram, the power $\hbar$ within the connected Green's function to the order of loop corrections for such a diagram. Again, I will not go into further detail as to how exactly all this is related to Feynman diagrams. The higher ones correspond to increasingly complicated Feynman diagrams and one can wonder to what extent they are physically relevant. One can also wonder how many loop corrections are relevant, but the goal of this research is to determine the effect of renormalization, and therefore we require high loop accuracy.

3.1.2 Extracting Connected Green's Functions

Once we have found a proper expression for $\phi(J)$, we can extract the connected Green's functions. As seen in equation (10), these are just the coefficients corresponding to certain powers of J. As stated before, we are doing perturbation theory, therefore we only know the connected Green's functions as a truncated series in $\hbar$, so the idea is to write them as polynomials in $\hbar$. They will be functions of the parameters from the action of our theory, μ and λ_4. The first three non-vanishing connected Green's functions look like this:

$$
\begin{aligned}
C_2 &= \frac{\hbar}{\mu}\left(1 - \frac{\lambda_4\hbar}{2\mu^2} + \frac{2\lambda_4^2\hbar^2}{3\mu^4} - \frac{11\lambda_4^3\hbar^3}{8\mu^6} + ...\right) \quad , \\
C_4 &= -\frac{\lambda_4\hbar^3}{\mu^4}\left(1 - \frac{7\lambda_4\hbar}{2\mu^2} + \frac{149\lambda_4^2\hbar^2}{12\mu^4} - \frac{197\lambda_4^3\hbar^3}{4\mu^6} + ...\right) \quad , \\
C_6 &= \frac{10\lambda_4^2\hbar^5}{\mu^7}\left(1 - \frac{8\lambda_4\hbar}{\mu^2} + \frac{1535\lambda_4^2\hbar^2}{3\mu^4} - \frac{6405\lambda_4^3\hbar^3}{2\mu^6} + ...\right) \quad .
\end{aligned} \tag{11}
$$

The perturbation scheme is quite evident here. All expansions are polynomials of the same variable, which we will discuss later on, and therefore also in $\hbar$.

3.2 Finding Renormalized Parameters

As promised, in this chapter, the necessity for renormalization will become clearer, while it is also explained in more detail what renormalization actually entails. Therefore, this chapter serves as an example, but it is also the way I have performed the renormalization process.

So, let us assume that, hypothetically, the exact values of the first two connected Green's functions, C_2 and C_4, have been experimentally found. Call these values E_2 and E_4. For simplicity, there are no measurement uncertainties whatsoever. These are, in principle, numerical values that cannot change under improvement of the theory. After all, we have measured them with infinite precision and they are not determined by the theory itself. For convenience, we assume the measured values to correspond to the form of the first terms of C_2 and C_4, their lowest order in perturbation theory (and $\hbar$). So we require:

$$
C_2 = E_2 \equiv \frac{\hbar}{\mu_R} \quad \text{and} \quad C_4 = E_4 \equiv -\frac{\lambda_{4R}\hbar^3}{\mu_R^4} \quad . \tag{12}
$$

From there, we can measure/extract the values of the parameters μ and λ_4. They must obviously be some sort of expression of μ_R and λ_{4R} (the R stands for real, although not in the sense one might think, as explained later). Once the parameters are known in terms of μ_R and λ_{4R}, they can then be used to predict the (numerical) values of the other connected Green's functions.

Again, the important thing to note here, is that we can only make predictions about the values of the other connected Green's functions to the same order as to which the parameters have been found. In other words, to which order the coupled set of equations (12) has been solved for μ and λ_4. In fact, this statement also goes for C_2 and C_4. That is because equation (12) must hold, for arbitrary order of perturbation theory. So, given the form we have chosen the experimentally obtained values to have, C_2 and C_4 cannot have higher powers of $\hbar$, after we reinsert the values of μ and λ we are about to calculate.

To first order (perturbation theory in C_2 and C_4) one would be tempted to assume $\mu = \mu_R$ and $\lambda_4 = \lambda_{4R}$. However, when plugging these values back in the first two connected Green's functions, one finds that for higher order perturbation theory, relation (12) is no longer satisfied. The order to which perturbation theory is done, determines how precisely one knows C_2 and C_4 (and the other connected Green's functions), and therefore it determines what equation (12) actually looks like. From this, it follows that our parameters must be refined (order-by-order) as we are doing higher orders in perturbation theory. What this means, is that μ and λ_4 must be functions of μ_R and λ_{4R}. So the R does not mean real in the way one would naively assign values to the parameters such that $\mu = \mu_R$ and $\lambda_4 = \lambda_{4R}$. But we have chosen, simply for convenience, that the experimental values can be expressed in terms of μ_R and λ_{4R}, such that one can write them in the same form as the first terms in C_2 and C_4. So these 'real' values are real, only in the sense that certain combinations of them correspond to the real values E_2 and E_4. The R is just notation.
From the definition (11), we see that the first two connected Green's functions, C_2 and C_4, can be written in the following form:

$$C_2 = \frac{\hbar}{\mu} f_2(x) \quad \text{and} \quad C_4 = -\frac{\lambda_4 \hbar^3}{\mu^4} f_4(x) \quad \text{, with} \quad x \equiv \frac{\hbar \lambda_4}{\mu^2} \quad . \tag{13}$$

So f_2 and f_4 are polynomials in x. We can then define the 'real' value of x to be (again, note that this is only real in the sense that it really does correspond to a measured value; we have already stated that that $\mu \neq \mu_R$ and $\lambda_4 \neq \lambda_{4R}$ for higher orders, so needless to say $x \neq x_R$):

$$x_R \equiv \frac{\lambda_{4R} \hbar}{\mu_R^2} \quad . \tag{14}$$

The way x and x_R are related, or rather, how that relation is determined will be discussed in the next paragraph. With the above definition of x_R, we can write equation (13) in the following way:

$$C_2 = \frac{\hbar}{\mu}\tilde{f}_2(x_R) \quad \text{and} \quad C_4 = -\frac{\lambda_4\hbar^3}{\mu^4}\tilde{f}_4(x_R) \quad . \tag{15}$$

We are trying to find μ and λ_4 as functions of μ_R and λ_{4R} and now we have written the first two connected Green's functions in a convenient way to do just that. Remember that equation (12) must still be true. We do not know the forms of $\tilde{f}_2$ and $\tilde{f}_4$ yet, however we do know exactly what f_2 and f_4 look like. The first thing we must do now, is find x as a function (polynomial) of x_R, let us call it $x \equiv F(x_R)$. We can manipulate the expressions for C_2, C_4, x and x_R in such a way, that we can find x_R as a function (polynomial) of x. This goes as follows:

$$-\frac{C_4}{C_2^2} = x\frac{f_4(x)}{f_2^2(x)} = x_R \quad . \tag{16}$$

Using the computer software 'Maple', we can invert this expression, to find the desired result for x, namely the function F. Now all we have to do is substitute this result in equation (13). Then, using relation (14), we can write equation (15), combined with equation (12) in the following manner:

$$\begin{aligned} C_2 &= \frac{\hbar}{\mu}f_2(F(x_R)) = \frac{\hbar}{\mu}\tilde{f}_2\left(\frac{\lambda_{4R}\hbar}{\mu_R^2}\right) = \frac{\hbar}{\mu_R} \quad \text{and} \\ C_4 &= -\frac{\lambda_4\hbar^3}{\mu^4}f_4(F(x_R)) = -\frac{\lambda_4\hbar^3}{\mu^4}\tilde{f}_4\left(\frac{\lambda_{4R}\hbar}{\mu_R^2}\right) = -\frac{\lambda_{4R}\hbar^3}{\mu_R^4} \quad . \end{aligned} \tag{17}$$

Again, using Maple, we now know what $\tilde{f}_2$ and $\tilde{f}_4$ look like, where it is convenient to express these as polynomials in $\hbar$. Consider first the expression for C_2 in equation (17). This one is easy to solve, as λ_4 and λ_{4R} only appear inside the polynomial. From the last equality, we can immediately see:

$$\mu = \mu_R\tilde{f}_2\left(\frac{\lambda_{4R}\hbar}{\mu_R^2}\right) \quad . \tag{18}$$

Now that we have obtained an expression for μ, let us take a look at the expression for C_4 in equation (17), again at the last equality. After a little bit of rearranging and using the expression (18) for μ, we find that:

$$\lambda_4 = \lambda_{4R} \frac{\tilde{f}_2^4\left(\frac{\lambda_{4R}\hbar}{\mu_R^2}\right)}{\tilde{f}_4\left(\frac{\lambda_{4R}\hbar}{\mu_R^2}\right)} \quad . \tag{19}$$

And that is it. We have now found the renormalized parameters. For reasons that will become apparent in the next chapter, we can write these parameters in the following way:

$$\mu = \mu_R g_1(x_R) \quad \text{and} \quad \lambda = \lambda_R g_2(x_R) \quad . \tag{20}$$

Here we simply used $g_1 \equiv \tilde{f}_2$ and $g_2 \equiv \frac{\tilde{f}_2^4(x_R)}{\tilde{f}_4(x_R)}$. So using Maple to expand these expressions, g_1 and g_2 are also just polynomials in x_R. Without calculating anything, it can actually be derived from the starting point of equation (12), that it is possible to express the parameters is this way. So as some polynomials of the variable x_R. This statement is important for the next paragraph.

3.3 Error check

In order to make sure we have not made any mistakes along the way and to determine to what order the iteration of the SDe still produces accurate results, one has to find a another relation that must be satisfied after doing the renormalization process we just went through. Plugging the renormalized parameters back in the equations for C_2 and C_4, to find that they match the measured values, is not really an honest test. After all, the starting assumption for the derivation of our renormalized parameters μ and λ_4, was that equation (12) must be satisfied. In order to perform a check, we need a consistency relation, for which we need to revisit some of the definitions from chapter 2.

3.3.1 Derivation

Let us start with equations (2) and (3). These definitions, among others will be used to test the correctness of the results calculated so far. Having said that, we will now try to find a relation between the first two (not counting $n = 0$) Green's functions and connected Green's functions. The way to do this, is to combine the expressions above and writing out the sums explicitly, to find:

$$C_2\frac{J^2}{2!}+C_4\frac{J^4}{4!}+...=\ln\left(1+G_2\frac{J^2}{2!}+G_4\frac{J^4}{4!}+...\right) \quad . \tag{21}$$

Next we need to expand the logarithm, so that we can compare the powers of J. This will lead to a relation between G_n and C_n for $n=2$ and $n=4$. Since we are only interested in the first two (connected) Green's functions, we can immediately discard anything with a power of J higher than four throughout the derivation. Conveniently, the expression inside the logarithm has a term 1, since the expansion we will use is $\ln(1+x)=x-\frac{x^2}{2}+\frac{x^3}{3}+...$, where we have expanded around $x=0$. So here x is the rest of the summation of Green's functions with $n>0$. After rewriting this and truncating both series at $J=4$, we find:

$$C_2J^2+C_4\frac{J^4}{12}=G_2J^2+G_4\frac{J^4}{12}-G_2^2\frac{J^4}{4} \quad . \tag{22}$$

So now we can compare the powers of J. Doing this gives us the desired relation, the coupling between the G_n and C_n, which we we will need later, namely:

$$\begin{aligned} G_2 &= C_2 \quad \text{and} \\ G_4 &= C_4+3C_2^2 \quad . \end{aligned} \tag{23}$$

Now consider equation (8). Instead of looking at derivatives of the field function with respect to J, as we have done before, we now look at derivatives of H_n with respect to μ and λ_4. A first order derivative with respect to either of these parameters will result in:

$$\begin{aligned} \frac{\partial}{\partial\mu}H_n &= -\frac{1}{2\hbar}H_{n+2} \quad \text{and} \\ \frac{\partial}{\partial\lambda_4}H_n &= -\frac{1}{24\hbar}H_{n+4} \quad . \end{aligned} \tag{24}$$

Combining these results, along with the expression for G_n in relation (8), means that we can write the first two Green's functions as follows:

$$\begin{aligned} G_2 &= -2\hbar\frac{\partial}{\partial\mu}W \quad \text{and} \\ G_4 &= -24\hbar\frac{\partial}{\partial\lambda_4}W \quad \text{, where} \\ W &\equiv \ln H_0 \quad . \end{aligned} \tag{25}$$

Now evaluate H_{2n} in equation (7), for $n = 0$. With that expression for H_0, it can be stated that, after some algebra and using Stirling's approximation, W should have the following form:

$$W = \log\sqrt{2\pi\hbar} - \frac{1}{2}\ln\mu + L(x) \quad . \tag{26}$$

Here x is the same as defined earlier in relation (13). The exact form of the function $L(x)$ is both unknown and unimportant. The only thing that matters is that it depends on x and therefore on μ and λ_4. The next step is to plug the expression for W back into equation (25). The resulting expressions obviously both contain $L'(x)$. But we know what the measured values for C_2 and C_4 are, or rather, what we defined them to be, see equation (12). We can insert these measured values into equation (23), and equate them to relation (25), where the expression for W is already included. The next step is to express both relations resulting from the previous steps in terms of the 'real' parameters. This is done by using equation (20). All x in the equation can be substituted with the following relation:

$$x = x_R\frac{g_2(x_R)}{g_1^2(x_R)} \quad . \tag{27}$$

Finally, we can equate the two resulting relations. This way, we can get rid of the derivative of $L(x)$ with respect to x (now written like in equation (27)), but at the same time, we can combine and rearrange everything, so that we arrive at a consistency relation that must hold, our check! This expression links the polynomials g_1 and g_2 to each other via x_R. The relation has the following form:

$$g_2 = \frac{6(1-g_1)}{x_R(3-x_R)} \quad . \tag{28}$$

In the derivation we just went through, nothing has actually been calculated. Similar as in the renormalization process, everything mostly follows from definitions, and both use equation (12) as the starting point. To find explicit form of g_1 and g_2, one needs to actually perform the renormalization process. Basically the only actual calculation performed there, was the iteration of the SDe. With that result we finally arrived at an exact form of g_1 and g_2. And now we have a consistency relation linking these power series together, and giving us an idea to which order the results are consistent.

3.3.2 Consistency results

After a lot of experimenting, I finally iterated the SDe to an order of 200, and naturally, the order to which I have expressed the polynomials and inverted the series was also 200. After plugging the renormalized parameters back into the first two connected Green's functions, I found that equation (12) holds up to an order of 200. However, after performing the check we just derived, I found that equation (28) is only valid for an order up to 134. In fact, it turns out that relation (28) only holds for an order of around 2/3 of the order to which one iterates the SDe (and performs the rest of the operations). This means that the the field function itself is only accurate up to powers of J and $\hbar$ of around 2/3 of the number of iterations.
This reinstates the importance of this check. Because in the next step, we will try to analyze the behaviour of the numerical coefficients of the connected Green's functions, before and after renormalization. Since the connected Green's functions are expressed as polynomials in $\hbar$, this check tells us that we can only say something about the coefficients belonging to a power of $\hbar$ that is 2/3 of the order of iteration.

It is hard to make an accurate statement as to why it is apparently 2/3 of the order of iteration, however it does make sense that the higher orders are not accurate anymore. The SDe contains derivatives, and even second derivatives, with respect to J. Furthermore, the SDe is not linear in $\phi(J)$. And since ϕ is a polynomial of J, a derivative of ϕ with respect to J lowers the power of J by one. So truncating the series, or the iteration, at a certain order, can never lead to full accuracy of the higher order terms; the next order result is also needed and so on.

One can solve this by letting the program iterate the SDe a certain number of times more than the highest order of J one desires to obtain. The output will still be a polynomial of the same order as before, only this time, the higher order terms will in fact be accurate. This was not done in this research however. As the results will show, it most likely would not have mattered that much anyway.

4 Results

After we have found the renormalized parameters μ and λ_4, we can plug them back into the connected Green's functions C_n. As stated before, for the first two connected Green's functions C_2 and C_4, this should lead to the expressions in equation (12). This means that they cannot have higher orders in $\hbar$ than their lowest orders, one and three respectively. Since this criterion was also the starting point of the derivation of our re-tuned values for μ and λ_4, the relation is obviously satisfied. But I also want to have a look at the effect of renormalization on the higher order connected Green's functions. In fact, that is precisely the goal of this paper. For simplicity, let us refer to the connected Green's functions where we have inserted the renormalized parameters as $C_{n,R}$.

4.1 Behaviour

For the connected Green's functions C_n, we can say $\mu = 1$ and $\lambda_4 = 1$. Equivalently, for the renormalized ones, $C_{n,R}$, we shall assume $\mu_R = 1$ and $\lambda_{4R} = 1$. The reason we do this, is because now it is easier to analyze the numerical coefficients, which we call $C_n^{(k)}$ and $C_{n,R}^{(k)}$, for C_n and $C_{n,R}$ respectively. We can now write the connected Green's functions as follows:

$$\begin{aligned} C_n(\mu = 1, \lambda_4 = 1) &= \sum_{k \geq n-1} C_n^{(k)} \hbar^k \quad \text{and} \\ C_{n,R}(\mu = 1, \lambda_{4R} = 1) &= \sum_{k \geq n-1} C_{n,R}^{(k)} \hbar^k \quad . \end{aligned} \tag{29}$$

Let us first have a look at the raw behaviour of the coefficients, before and after renormalization. For all C_n I have evaluated, $n = 6$, $n = 8$, $n = 10$ and $n = 12$, the behaviour is very similar. So I will only provide plots for $C_6^{(k)}$ and $C_{6,R}^{(k)}$. This is shown is the figures below. Immediately it is clear that the behaviour has not really changed much. Note that in all the figures, the absolute values of the coefficients are plotted. This is because all the connected Green's functions are alternating series, also after renormalization.

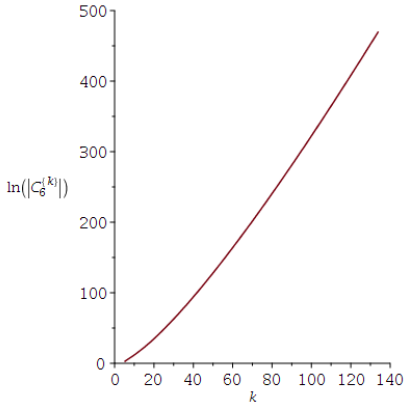

Figure 2: This graph shows behaviour of numerical coefficients of C_6.

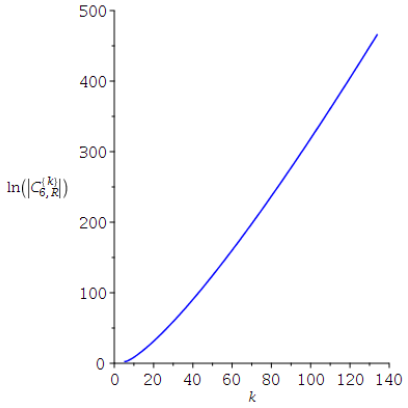

Figure 3: This graph shows behaviour of numerical coefficients of $C_{6,R}$.

It can be derived, using Stirling's approximation, that the k^{th} coefficient of the (connected) Green's functions contains a factor $k!$. When plotting the logarithm of the coefficients divided by $k!$, one can further analyze the behaviour of these coefficients. This is done in the figure below.

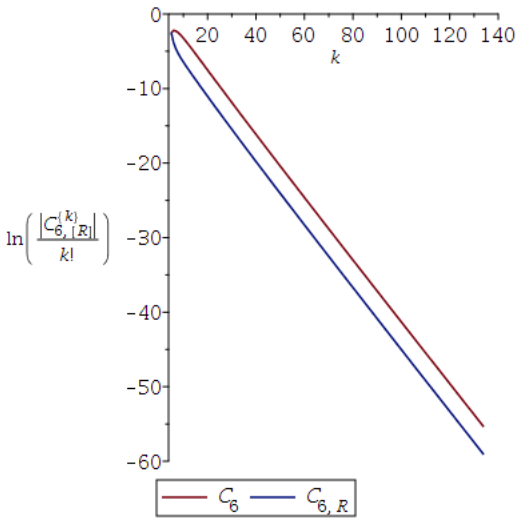

Figure 4: This diagram shows the logarithm of the coefficients of the higher connected Green's functions (before and after renormalization) divided by k factorial.

Obviously, the factorial in the expressions for the numerical coefficients of the connected Green's functions is the dominant factor. The next most dominant part of the expression would then be a constant to the power of k. After plotting the values of the coefficients $C_n^{(k)}$ and $C_{n,R}^{(k)}$ and slightly adjusting the parameters λ_4 and λ_{4R} respectively, it can safely be concluded that both coefficients behave as follows:

$$C_n^{(k)} \sim k!\alpha^k \cdot P(k) \quad \text{and} \quad C_n^{(k)} \sim k!\alpha^k \cdot P_R(k) \quad . \tag{30}$$

Here, P and P_R are polynomials of k, and α is just a constant. Because of the dominance compared to the polynomials (especially after large k), this constant, or its natural logarithm rather, gives approximately the slopes of the graphs in figure (4). These graphs appear to have the exact same slope. Calculating the slope, gives $\alpha \approx 0.66$. So again, for large k, the coefficients are dominated by $k!$, while α^k also has a significant influence. The renormalized coefficients still contain a factorial of k, but it turns out that not even the base α is changed. The only difference between the coefficients before and after renormalization must be in the polynomials P and P_{R2}, whatever their exact form might be. It is not really necessary to compute these polynomials. For large k, they can certainly be neglected and besides, we are dealing with a divergent series regardless of the exact form of the polynomials. So in this case, knowing precisely how the coefficients approach infinity is irrelevant.

4.2 Comparing

By plotting the (natural) logarithm of the ratio of these coefficients versus k, one can quickly see how they compare to each other. It turns out that for larger k, this function seems to approach a constant. This is the case for all the C_n I have plotted.

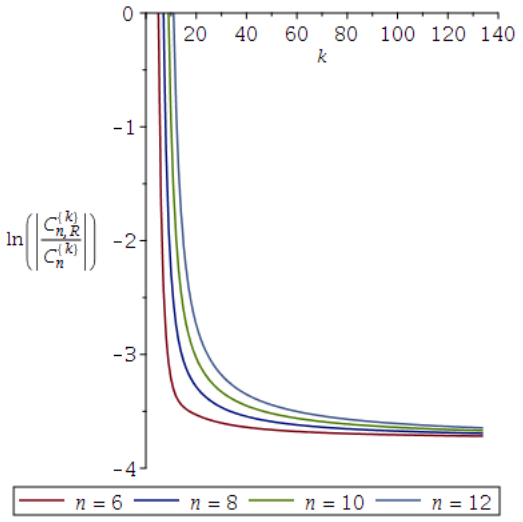

Figure 5: This diagram shows the ratios of the coefficients of four connected Green's functions, before and after renormalization.

I do not really know why all the higher connected Green's functions behave in such a similar way. So, now it is roughly known how the coefficients behave, and given that the ratio seems to approach a constant, we can say (for large k):

$$\frac{C_{n,R}^{(k)}}{C_n^{(k)}} = \frac{P_R(k)}{P(k)} \approx \frac{1}{40} \quad . \tag{31}$$

Note that this only holds for the numerical coefficients of the connected Green's functions, not necessarily for the functions themselves! After all, we have set $\mu = \lambda_4 = \mu_R = \lambda_{4R} = 1$, in order to easily evaluate the numerical coefficients. In reality however, we do not know beforehand what these parameters are. This is only a test to see how the connected Green's functions behave. It can be concluded that they behave as an (extremely) divergent series, also after renormalization, unfortunately.

5 Conclusion

I hoped that the factorial would have disappeared in the behaviour of the coefficients of the renormalized parameters, but it turns out that this is not the case. In fact, it seems that after a while, the renormalized coefficients are simply a constant factor of around 40 smaller than their original counterparts. So while they did get a little bit smaller, they still tend towards infinity at quite an alarming rate. In short, we are still dealing with an extremely divergent series.
Furthermore, the behaviour of the coefficients is, for large k extremely similar. So while the results regarding the improvements after renormalization are quite disappointing, they do raise interesting questions nonetheless. In other words, renormalization, at least in this 0-D toy model, is very ineffective, but nevertheless, still necessary!

References

[1] Kleiss, R.H.P (2018). Pictures Paths Particles Processes. Radboud University.

Appendices

A Source code

Below is the source code I used. The only difference between the two methods, is the way iteration has been done. Since the triangle-method is the one used in the end, I have also included more steps of the renormalization process. From there, plotting the results, or finding the consistency relation is a trivial matter and is therefore not included. The rectangle-method only shows the iteration part.

A.1 Triangle-method

```
restart;
phi:=0:
for k from 1 to 200 do print(k):
  phi:= u*J/mu - lambda[4]/6/mu*(phi^3 + 3*hbar*phi*diff(phi,J)
          + hbar^2*diff(phi,J$2));
  phi:=expand(series(phi,u=0,k+1));
  phi:=convert(phi,polynom);
od:
phi:=subs(u=1,phi):

C[2]:=sort(convert(series((hbar^1)*coeff(phi,J,1)*1!,
           hbar=0,200),polynom),hbar,ascending):
C[4]:=sort(convert(series((hbar^3)*coeff(phi,J,3)*3!,
           hbar=0,200),polynom),hbar,ascending):
C[6]:=sort(convert(series((hbar^5)*coeff(phi,J,5)*5!,
           hbar=0,200),polynom),hbar,ascending):
C[8]:=sort(convert(series((hbar^7)*coeff(phi,J,7)*7!,
           hbar=0,200),polynom),hbar,ascending):
C[10]:=sort(convert(series((hbar^9)*coeff(phi,J,9)*9!,
            hbar=0,200),polynom),hbar,ascending):
C[12]:=sort(convert(series((hbar^11)*coeff(phi,J,11)*11!,
            hbar=0,200),polynom),hbar,ascending):

f[2]:=algsubs(hbar*lambda[4]/mu^2=x, convert(series(mu
              *coeff(phi,J,1)*1!,hbar=0,200),polynom)):
```

```
f[4]:=algsubs(hbar*lambda[4]/mu^2=x, convert(series(
              (-mu^4/lambda[4])*coeff(phi,J,3)*3!,
              hbar=0,200),polynom)):
K:=convert(series(x*f[4]/f[2]^2,x=0,200),polynom):
Order:=200:
S:=solve(series(K,x)=x[R],x):
f[2,X,R]:=expand(series(subs(x=S,f[2]),x[R]=0,200)):
f[2,R]:=subs(x[R]=hbar*lambda[4,R]/mu[R]^2, f[2,X,R]):
f[2,X]:=solve(series(mu[R]*f[2,X,R],x[R])=mu,mu):
MU:=subs(x[R]=hbar*lambda[4,R]/mu[R]^2, f[2,X]):
f[4,X,R]:=expand(series(subs(x=S,f[4]),x[R]=0,200)):
f[4,R]:=subs(x[R]=hbar*lambda[4,R]/mu[R]^2, f[4,X,R]):
f[4,X]:=solve(series(mu[R]^4*f[4,X,R]/lambda[4,R]
              /f[2,X]^4,x[R])=1/lambda[4],lambda[4]):
LAMBDA:=subs(x[R]=hbar*lambda[4,R]/mu[R]^2, f[4,X]):

C[2,R]:=expand(series(subs(mu=MU, lambda[4]=LAMBDA, C[2]),
               hbar=0,200)):
C[4,R]:=expand(series(subs(mu=MU, lambda[4]=LAMBDA, C[4]),
               hbar=0,200)):
C[6,R]:=expand(series(subs(mu=MU, lambda[4]=LAMBDA, C[6]),
               hbar=0,200)):
C[8,R]:=expand(series(subs(mu=MU, lambda[4]=LAMBDA, C[8]),
               hbar=0,200)):
C[10,R]:=expand(series(subs(mu=MU, lambda[4]=LAMBDA, C[10]),
               hbar=0,200)):
C[12,R]:=expand(series(subs(mu=MU, lambda[4]=LAMBDA, C[12]),
               hbar=0,200)):
```

A.2 Rectangle-method

```
restart;
phi:=0:
for k from 1 to 200 do print(k):
  phi:= J/mu - lambda[4]/6/mu*(phi^3 + 3*hbar*phi*diff(phi,J)
        + hbar^2*diff(phi,J$2));
  phi:=convert(series(phi,hbar=0,k+1),polynom);
  phi:=convert(series(phi,J=0,20),polynom);
od:
phi:
```