

Modern Techniques for One-Loop Calculations

J. C. Romão

Departamento de Física and CFTP, Instituto Superior Técnico

Avenida Rovisco Pais 1, 1049-001 Lisboa, Portugal

`jorge.romao@tecnico.ulisboa.pt`

May 14, 2020

Abstract

We review the techniques used for one-loop calculations with emphasis on practical applications. QED is used as an example but the methods can be used in any theory. The aim is to teach how to use modern techniques, like the symbolic package `FeynCalc` for `Mathematica` and the numerical package `LoopTools` for `Fortran` or `C++`, in one-loop calculations.

Note Added:

This is a new version of an old text that I wrote mostly for my personal use and of my students. Some years ago it went into an appendix of my text on *Advanced Quantum Field Theory*[1]. From that moment all corrections were done in the appendix and not in this text, which made the texts to diverge. To avoid that I decided to synchronize them, so now they are completely equal. I did not delete this text as it appears if you make a search on Google.

Lisbon, 14/05/2020

Jorge C. Romão

Contents

1	μ parameter	4
2	Feynman parameterization	5
3	Wick Rotation	8
4	Scalar integrals in dimensional regularization	10
5	Tensor integrals in dimensional regularization	12
6	Γ function and useful relations	14
7	Explicit formulæ for the <i>1-loop</i> integrals	15
7.1	<i>Tadpole</i> integrals	16
7.2	<i>Self-Energy</i> integrals	16
7.3	<i>Triangle</i> integrals	17
7.4	<i>Box</i> integrals	17
8	Divergent part of <i>1-loop</i> integrals	18
8.1	<i>Tadpole</i> integrals	18
8.2	<i>Self-Energy</i> integrals	18
8.3	<i>Triangle</i> integrals	19
8.4	<i>Box</i> integrals	19
9	Passarino-Veltman Integrals	19
9.1	The general definition	19
9.2	The divergences	22
9.3	Useful results for PV integrals	22
9.3.1	Explicit expression for A_0	23
9.3.2	Explicit expressions for the B functions	23
9.3.3	Explicit expressions for the C functions	26
9.3.4	The package PVzem	29
9.3.5	Explicit expressions for the D functions	32
10	Examples of <i>1-loop</i> calculations with PV functions	32
10.1	Vacuum Polarization in QED	32
10.2	Electron Self-Energy in QED	34
10.3	QED Vertex	37

11 Modern techniques in a real problem: $\mu \rightarrow e\gamma$	41
11.1 Neutral scalar charged fermion loop	41
11.2 Charged scalar neutral fermion loop	51

1 μ parameter

The reason for the μ parameter introduced in section 10.1 is the following. In dimension $d = 4 - \epsilon$, the fields A_μ and ψ have dimensions given by the kinetic terms in the action,

$$\int d^d x \left[-\frac{1}{4}(\partial_\mu A_\nu - \partial_\nu A_\mu)^2 + i\bar{\psi}\gamma \cdot \partial\psi \right] \quad (1)$$

We have therefore

$$\begin{aligned} 0 &= -d + 2 + 2[A_\mu] \Rightarrow [A_\mu] = \frac{1}{2}(d - 2) = 1 - \frac{\epsilon}{2} \\ 0 &= -d + 1 + 2[\psi] \Rightarrow [\psi] = \frac{1}{2}(d - 1) = \frac{3}{2} - \frac{\epsilon}{2} \end{aligned} \quad (2)$$

Using these dimensions in the interaction term

$$S_I = \int d^d x e\bar{\psi}\gamma_\mu\psi A^\mu \quad (3)$$

we get

$$\begin{aligned} [S_I] &= -d + [e] + 2[\psi] + [A] \\ &= -4 + \epsilon + [e] + 3 - \epsilon + 1 - \frac{\epsilon}{2} \\ &= [e] - \frac{\epsilon}{2} \end{aligned} \quad (4)$$

Therefore, if we want the action to be dimensionless (remember that we use the system where $\hbar = c = 1$), we have to set

$$[e] = \frac{\epsilon}{2} \quad (5)$$

We see then that in dimensions $d \neq 4$ the coupling constant has dimensions. As it is more convenient to work with a dimensionless coupling constant we introduce a parameter μ with dimensions of a mass and in $d \neq 4$ we will make the substitution

$$e \rightarrow e\mu^{\frac{\epsilon}{2}} \quad (\epsilon = 4 - d) \quad (6)$$

while keeping e dimensionless.

2 Feynman parameterization

The most general form for a 1 -loop is ¹

$$\hat{T}_n^{\mu_1 \dots \mu_p} \equiv \int \frac{d^d k}{(2\pi)^d} \frac{k^{\mu_1} \dots k^{\mu_p}}{D_0 D_1 \dots D_{n-1}} \quad (7)$$

where

$$D_i = (k + r_i)^2 - m_i^2 + i\epsilon \quad (8)$$

and the momenta r_i are related with the external momenta (all taken to be incoming) through the relations,

$$\begin{aligned} r_j &= \sum_{i=1}^j p_i \quad ; \quad j = 1, \dots, n-1 \\ r_0 &= \sum_{i=1}^n p_i = 0 \end{aligned} \quad (9)$$

as indicated in Fig. (1). In these expressions there appear in the denominators products

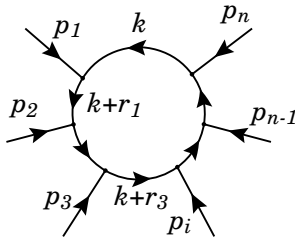


Figure 1: Conventions for the momenta in the loop.

of the denominators of the propagators of the particles in the loop. It is convenient to combine these products in just one common denominator. This is achieved by a technique due to Feynman. Let us exemplify with two denominators.

$$\frac{1}{ab} = \int_0^1 \frac{dx}{[ax + b(1-x)]^2} \quad (10)$$

The proof is trivial. In fact

$$\int dx \frac{1}{[ax + b(1-x)]^2} = \frac{x}{b[(a-b)x + b]} \quad (11)$$

and therefore Eq. (10) immediately follows. Taking successive derivatives with respect to a and b we get

$$\frac{1}{a^p b^q} = \frac{\Gamma(p+q)}{\Gamma(p)\Gamma(q)} \int_0^1 dx \frac{x^{p-1}(1-x)^{q-1}}{[ax + b(1-x)]^{p+q}} \quad (12)$$

¹We introduce here the notation \hat{T} to distinguish from a more standard notation that will be explained in subsection 9.

and using induction we obtain a general formula

$$\frac{1}{a_1 a_2 \cdots a_n} = \Gamma(n) \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \cdots \int_0^{1-x_1-\cdots-x_{n-2}} \frac{dx_{n-1}}{[a_1 x_1 + a_2 x_2 + \cdots + a_n (1 - x_1 - \cdots - x_{n-1})]^n} \quad (13)$$

Complement 2.1

Let us take a closer look at Eq. (13) and derive it in a different way that will make more clear the range of variation of the Feynman parameters. We follow closely the argument of Gross [2].

We start with the definition of the Γ function,

$$\Gamma(\alpha) = \int_0^\infty dt t^{\alpha-1} e^{-t} \quad (14)$$

Making a change of variables we also get

$$\frac{\Gamma(\alpha)}{a^\alpha} = \int_0^\infty dt t^{\alpha-1} e^{-t a} \quad (15)$$

We consider first the case of two denominators using Eq. (15) with $\alpha = 1$. We get

$$\frac{1}{a b} = \int_0^\infty \int_0^\infty dt_1 dt_2 e^{-(t_1 a + t_2 b)} \quad (16)$$

Now we introduce 1 in the form

$$1 = \int_0^\infty dt \delta(t - t_1 - t_2) \quad (17)$$

in Eq. (16) to get

$$\frac{1}{a b} = \int_0^\infty \int_0^\infty \int_0^\infty dt dt_1 dt_2 \delta(t - t_1 - t_2) e^{-(t_1 a + t_2 b)} \quad (18)$$

To continue we scale the variables $t_1 = t x_1$ and $t_2 = t x_2$. We then get

$$\frac{1}{a b} = \int_0^\infty \int_0^\infty dx_1 dx_2 \delta(1 - x_1 - x_2) \int_0^\infty dt t e^{-t(x_1 a + x_2 b)} \quad (19)$$

Now we use the definition in Eq. (15) to obtain

$$\begin{aligned} \frac{1}{a b} &= \Gamma(2) \int_0^\infty \int_0^\infty dx_1 dx_2 \delta(1 - x_1 - x_2) \frac{1}{[x_1 a + x_2 b]^2} \\ &= \int_0^1 dx_1 \frac{1}{[x_1 a + (1 - x_1) b]^2} \end{aligned} \quad (20)$$

in agreement with Eq. (10). The nice thing about this procedure is that it can be generalized easily to obtain

$$\begin{aligned} \frac{1}{a_1 a_2 \cdots a_n} &= \Gamma(n) \int_0^\infty dx_1 \cdots \int_0^\infty dx_n \frac{\delta(1 - x_1 - \cdots - x_n)}{[a_1 x_1 + a_2 x_2 + \cdots + a_n x_n]^n} \\ &= \Gamma(n) \int_0^1 dx_1 \cdots \int_0^{1-x_1 \cdots x_{n-1}} dx_{n-1} \frac{1}{[a_1 x_1 + a_2 x_2 + \cdots + a_n (1 - x_1 \cdots x_{n-1})]^n} \end{aligned} \quad (21)$$

where the limits in the last equation can be understood by the fact that the delta function defines an hyperplane that constrains the variables. For instance consider the case of $n = 3$. One gets the condition that defines a plane in the 3 dimensional space,

$$1 - x_1 - x_2 - x_3 = 0 , \quad (22)$$

as can be seen in Fig. 2. As the x_i are positive, we immediately see that they obey, for the case of

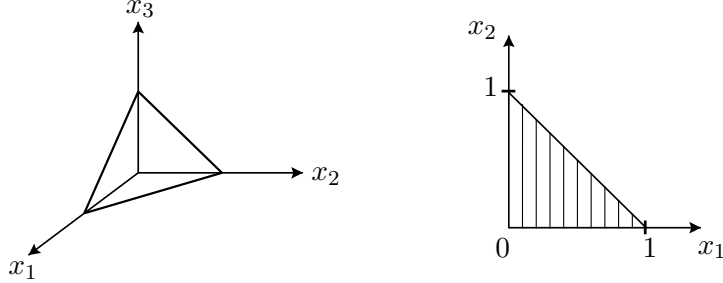


Figure 2: Graphical representation of the constraint of Eq. (22) on the Feynman parameters. On the right panel the projection on the x_1x_2 plane.

n denominators,

$$x_1 < 1, \quad x_2 < 1 - x_1, \quad x_3 < 1 - x_1 - x_2, \quad \dots, \quad x_{n-1} < 1 - x_1 - \dots - x_{n-2} \quad (23)$$

Before closing the section let us give an example that will be useful in the self-energy case. Consider the situation with the kinematics described in Fig. (3).

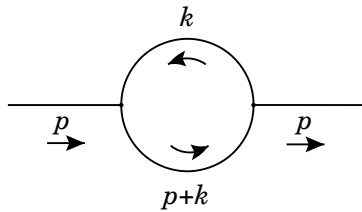


Figure 3: Kinematics for the self-energy in ϕ^3 .

We get

$$\begin{aligned} I &= \int \frac{d^d k}{(2\pi)^d} \frac{1}{[(k+p)^2 - m_1^2 + i\epsilon] [k^2 - m_2^2 + i\epsilon]} \\ &= \int_0^1 dx \int \frac{d^d k}{(2\pi)^d} \frac{1}{[k^2 + 2p \cdot k x + p^2 x - m_1^2 x - m_2^2 (1-x) + i\epsilon]^2} \end{aligned}$$

$$\begin{aligned}
&= \int_0^1 dx \int \frac{d^d k}{(2\pi)^d} \frac{1}{[k^2 + 2P \cdot k - M^2 + i\epsilon]^2} \\
&= \int_0^1 dx \int \frac{d^d k}{(2\pi)^d} \frac{1}{[(k + P)^2 - P^2 - M^2 + i\epsilon]^2}
\end{aligned} \tag{24}$$

where in the last line we have completed the square in the term with the loop momenta k . The quantities P and M^2 are, in this case, defined by

$$P = xp \tag{25}$$

and

$$M^2 = -xp^2 + m_1^2 x + m_2^2 (1 - x) \tag{26}$$

They depend on the masses, external momenta and Feynman parameters, but not in the loop momenta. Now changing variables $k \rightarrow k - P$ we get rid of the linear terms in k and finally obtain

$$I = \int_0^1 dx \int \frac{d^d k}{(2\pi)^d} \frac{1}{[k^2 - C + i\epsilon]^2} \tag{27}$$

where C is independent of the loop momenta k and it is given by

$$C = P^2 + M^2 \tag{28}$$

Notice that the $i\epsilon$ factors will add correctly and can all be put as in Eq. (27).

3 Wick Rotation

From the example of the last section we can conclude that all the scalar integrals can be reduced to the form

$$I_{r,m} = \int \frac{d^d k}{(2\pi)^d} \frac{k^{2r}}{[k^2 - C + i\epsilon]^m} \tag{29}$$

It is also easy to realize that also all the tensor integrals can be obtained from the scalar integrals. For instance

$$\begin{aligned}
&\int \frac{d^d k}{(2\pi)^d} \frac{k^\mu}{[k^2 - C + i\epsilon]^m} = 0 \\
&\int \frac{d^d k}{(2\pi)^d} \frac{k^\mu k^\nu}{[k^2 - C + i\epsilon]^m} = \frac{1}{d} g^{\mu\nu} \int \frac{d^d k}{(2\pi)^d} \frac{k^2}{[k^2 - C + i\epsilon]^m}
\end{aligned} \tag{30}$$

and so on. Therefore the integrals $I_{r,m}$ are the important quantities to evaluate. We will consider that $C > 0$. The case $C < 0$ can be done by analytical continuation of the final formula for $C > 0$.

To evaluate the integral $I_{r,m}$ we will use integration in the complex plane of the variable k^0 as described in Fig. 4. We can then write

$$I_{r,m} = \int \frac{d^{d-1} k}{(2\pi)^d} \int dk^0 \frac{k^{2r}}{[k_0^2 - |\vec{k}|^2 - C + i\epsilon]^m} \tag{31}$$

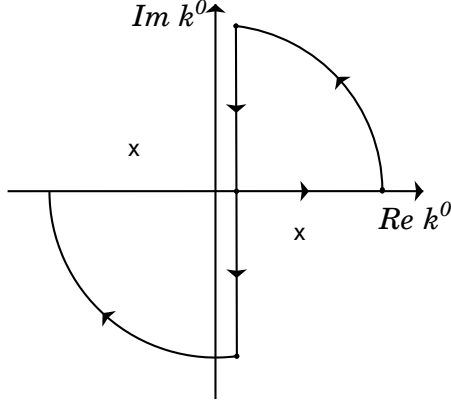


Figure 4: Integration contour path for the Wick rotation.

The function under the integral has poles for

$$k^0 = \pm \left(\sqrt{|\vec{k}|^2 + C} - i\epsilon \right) \quad (32)$$

as shown in Fig. 4. Using the properties of functions of complex variables (Cauchy theorem) we can deform the contour, changing the integration from the real to the imaginary axis plus the two arcs at infinity. This can be done because in deforming the contour we do not cross any pole. Notice the importance of the $i\epsilon$ prescription to be able to do this. The contribution from the arcs at infinity vanishes in dimension sufficiently low for the integral to converge, as we assume in dimensional regularization (see the details below in Complement 3.1). This means that

$$\int_{-\infty}^{+\infty} dk^0 + \int_{+i\infty}^{-i\infty} dk^0 = 0 \implies \int_{-\infty}^{+\infty} dk^0 = \int_{-i\infty}^{+i\infty} dk^0 \quad (33)$$

We can then change the integration along the real axis into an integration along the imaginary axis in the plane of the complex variable k^0 . If we write

$$k^0 = ik_E^0 \quad \text{com} \quad \int_{-\infty}^{+\infty} dk^0 \rightarrow i \int_{-\infty}^{+\infty} dk_E^0 \quad (34)$$

and $k^2 = (k^0)^2 - |\vec{k}|^2 = -(k_E^0)^2 - |\vec{k}|^2 \equiv -k_E^2$, where $k_E = (k_E^0, \vec{k})$ is an euclidean vector. By this we mean that we calculate the scalar product using the euclidean metric $\text{diag}(+, +, +, +)$,

$$k_E^2 = (k_E^0)^2 + |\vec{k}|^2 \quad (35)$$

We can then write

$$I_{r,m} = i(-1)^{r-m} \int \frac{d^d k_E}{(2\pi)^d} \frac{k_E^{2r}}{[k_E^2 + C]^m} \quad (36)$$

where we do not need the $i\epsilon$ because the denominator is strictly positive ($C > 0$). This procedure is known as *Wick Rotation*. We note that the Feynman prescription for the

propagators that originated the $i\epsilon$ rule for the denominators is crucial for the Wick rotation to be possible.

Complement 3.1

In the argument that allowed for the Wick rotation it was claimed that the integrals over the circles at infinite vanish. Let us be more careful on this point. We just start with the simplest integral,

$$I_{0,m} = \int \frac{d^d k}{(2\pi)^d} \frac{1}{[k^2 - C + i\epsilon]^m} \quad (37)$$

We begin by using the following representation for the denominator,

$$\frac{1}{[k^2 - C + i\epsilon]} = (-i) \int_0^\infty dz e^{-iz(C - k^2 - i\epsilon)} \quad (38)$$

which can be verified by direct integration noticing the crucial role of the $i\epsilon$ prescription. This representation is related to the Schwinger proper time method [3]. Now we differentiate both sides with respect to C to obtain,

$$\frac{1}{[k^2 - C + i\epsilon]^m} = \frac{(-i)^m}{\Gamma(m)} \int_0^\infty dz z^{m-1} e^{-iz(C - k^2 - i\epsilon)} \quad (39)$$

Now introduce this in Eq. (37) and separate the integral in k^0 . We get,

$$\begin{aligned} I_{0,m} &= \int \frac{d^{d-1} k}{(2\pi)^d} \int dk^0 \frac{1}{[k^2 - C + i\epsilon]^m} \\ &= \int \frac{d^{d-1} k}{(2\pi)^d} \int dk^0 \frac{(-i)^m}{\Gamma(m)} \int_0^\infty dz z^{m-1} e^{-iz(C - k^2 - i\epsilon)} \\ &= \int \frac{d^{d-1} k}{(2\pi)^d} \frac{(-i)^m}{\Gamma(m)} \int_0^\infty dz z^{m-1} \int dk^0 e^{-iz(C - (k^0)^2 + \vec{k} \cdot \vec{k} - i\epsilon)} \end{aligned} \quad (40)$$

We now go to the plane of complex $k^0 = |k^0|(\cos \theta + i \sin \theta)$. Therefore

$$(k^0)^2 = |k^0|^2 (\cos 2\theta + i \sin 2\theta) \quad (41)$$

and the integral in k^0 will be

$$\int dk^0 e^{-iz(C - (k^0)^2 + \vec{k} \cdot \vec{k} - i\epsilon)} = e^{-iz(C + \vec{k} \cdot \vec{k} - i\epsilon)} \int dk^0 e^{-z|k^0|^2 \sin 2\theta} e^{-iz|k^0|^2 \cos 2\theta} \quad (42)$$

and it will vanish in the circle at infinity for any value of θ . This shows that for $I_{0,m}$ we can perform the Wick rotation. This is also true for the general case of $I_{r,m}$ as the exponential vanishes faster than any power. This concludes the proof that we are allowed to perform the Wick rotation that lead to Eq. (36). We also note that the integration on the circles also vanish for finite values of $|k^0|$, as they are equal and with opposite signs.

4 Scalar integrals in dimensional regularization

We have seen in the last section that the scalar integrals to be calculated with dimensional regularization had the general form of Eq. (36). We are now going to find a general formula

for $I_{r,m}$. We begin by writing

$$\int d^d k_E = \int d\bar{k} \bar{k}^{d-1} d\Omega_{d-1} \quad (43)$$

where $\bar{k} = \sqrt{(k_E^0)^2 + |\vec{k}|^2}$ is the length of the vector k_E in the euclidean space in d dimensions and $d\Omega_{d-1}$ is the solid angle that generalizes spherical coordinates in that euclidean space. The angles are defined by

$$k_E = \bar{k}(\cos \theta_1, \sin \theta_1 \cos \theta_2, \sin \theta_1 \sin \theta_2, \sin \theta_1 \sin \theta_2 \cos \theta_3, \dots, \sin \theta_1 \cdots \sin \theta_{d-1}) \quad (44)$$

We can then write

$$\int d\Omega_{d-1} = \int_0^\pi \sin \theta_1^{d-2} d\theta_1 \cdots \int_0^{2\pi} d\theta_{d-1} \quad (45)$$

Using now

$$\int_0^\pi \sin \theta^m d\theta = \sqrt{\pi} \frac{\Gamma(\frac{m+1}{2})}{\Gamma(\frac{m+2}{2})} \quad (46)$$

where $\Gamma(z)$ is the gamma function (see section 6) we get

$$\int d\Omega_{d-1} = 2 \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2})} \quad (47)$$

The integration in \bar{k} is done using the result

$$\int_0^\infty dx \frac{x^p}{(x^2 + C)^m} = \frac{\Gamma(\frac{p+1}{2}) C^{\frac{1}{2}(p-2m+1)} \Gamma(-\frac{p}{2} + m - \frac{1}{2})}{2 \Gamma(m)}. \quad (48)$$

and we finally get

$$I_{r,m} = i C^{r-m+\frac{d}{2}} \frac{(-1)^{r-m}}{(4\pi)^{\frac{d}{2}}} \frac{\Gamma(r+\frac{d}{2})}{\Gamma(\frac{d}{2})} \frac{\Gamma(m-r-\frac{d}{2})}{\Gamma(m)} \quad (49)$$

Before ending the section we note that the integral representation for $I_{r,m}$, Eq. (29), is valid only for $d < 2(m-r)$ to ensure convergence when $\bar{k} \rightarrow \infty$. However the final form in Eq. (49) can be analytically continued for all values of d except for those where the function $\Gamma(m-r-d/2)$ has poles, that is for (see section 6),

$$m-r-\frac{d}{2} \neq 0, -1, -2, \dots \quad (50)$$

For the application in dimensional regularization it is convenient to rewrite Eq. (49) using the relation $d = 4 - \epsilon$. We get

$$I_{r,m} = i \frac{(-1)^{r-m}}{(4\pi)^2} \left(\frac{4\pi}{C}\right)^{\frac{\epsilon}{2}} C^{2+r-m} \frac{\Gamma(2+r-\frac{\epsilon}{2})}{\Gamma(2-\frac{\epsilon}{2})} \frac{\Gamma(m-r-2+\frac{\epsilon}{2})}{\Gamma(m)} \quad (51)$$

5 Tensor integrals in dimensional regularization

We are frequently faced with the task of evaluating the tensor integrals of the form of Eq. (7),

$$\hat{T}_n^{\mu_1 \dots \mu_p} \equiv \int \frac{d^d k}{(2\pi)^d} \frac{k^{\mu_1} \dots k^{\mu_p}}{D_0 D_1 \dots D_{n-1}} \quad (52)$$

The first step is to reduce to one common denominator using the Feynman parameterization technique. The result is,

$$\begin{aligned} \hat{T}_n^{\mu_1 \dots \mu_p} &= \Gamma(n) \int_0^1 dx_1 \dots \int_0^{1-x_1-\dots-x_{n-2}} dx_{n-1} \int \frac{d^d k}{(2\pi)^d} \frac{k^{\mu_1} \dots k^{\mu_p}}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^n} \\ &= \Gamma(n) \int_0^1 dx_1 \dots \int_0^{1-x_1-\dots-x_{n-2}} dx_{n-1} I_n^{\mu_1 \dots \mu_p} \end{aligned} \quad (53)$$

where we have defined

$$I_n^{\mu_1 \dots \mu_p} \equiv \int \frac{d^d k}{(2\pi)^d} \frac{k^{\mu_1} \dots k^{\mu_p}}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^n} \quad (54)$$

that we call, from now on, the tensor integral. In principle all these integrals can be written in terms of scalar integrals. It is however convenient to have a general formula for them. We start with the result,

$$\begin{aligned} I_{0,n} &= \int \frac{d^d k}{(2\pi)^d} \frac{1}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^n} \\ &= \frac{i}{(4\pi)^{d/2}} (-1)^n \frac{\Gamma(n - d/2)}{\Gamma(n)} \left(\frac{1}{C} \right)^{n-d/2} \end{aligned} \quad (55)$$

where we used the result in Eq. (49) and use the definition of the Γ function,

$$\left(\frac{1}{C} \right)^z = \frac{1}{\Gamma(z)} \int_0^\infty dt t^{z-1} e^{-tC} \quad (56)$$

to write

$$\int \frac{d^d k}{(2\pi)^d} \frac{1}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^n} = \frac{i}{(4\pi)^{d/2}} (-1)^n \frac{1}{\Gamma(n)} \int_0^\infty dt t^{n-1-d/2} e^{-tC} \quad (57)$$

Now we use

$$\frac{\partial}{\partial P^\mu} \frac{1}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^n} = -n \frac{2k_\mu}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^{n+1}} \quad (58)$$

to show that

$$\frac{k^{\mu_1} \dots k^{\mu_p}}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^n} = \frac{(-1)^p \Gamma(n-p)}{2^p \Gamma(n)} \frac{\partial}{\partial P_{\mu_1}} \dots \frac{\partial}{\partial P_{\mu_p}} \frac{1}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^{n-p}} \quad (59)$$

We then use Eq. (57) to write

$$\begin{aligned} \int \frac{d^d k}{(2\pi)^d} \frac{1}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^{n-p}} &= \frac{i}{(4\pi)^{d/2}} (-1)^{n-p} \frac{1}{\Gamma(n-p)} \int_0^\infty dt t^{n-p-1-d/2} e^{-tC} \\ &= \frac{i}{16\pi^2} (-1)^{n-p} \frac{(4\pi)^{\epsilon/2}}{\Gamma(n-p)} \int_0^\infty dt t^{n-p-3+\epsilon/2} e^{-tC} \end{aligned} \quad (60)$$

Inserting Eq. (59) and Eq. (60) into Eq. (54) we finally get the result

$$I_n^{\mu_1 \dots \mu_p} = \frac{i}{16\pi^2} \frac{(4\pi)^{\epsilon/2}}{\Gamma(n)} (-1)^n \int_0^\infty \frac{dt}{(2t)^p} t^{n-3+\epsilon/2} \frac{\partial}{\partial P_{\mu_1}} \dots \frac{\partial}{\partial P_{\mu_p}} e^{-tC} \quad (61)$$

where $C = P^2 + M^2$. After doing the derivatives the remaining integrals can be done using the properties of the Γ function (see section 6). Notice that P , M^2 and therefore also C depend not only in the Feynman parameters but also in the exterior momenta. The advantage of having a general formula is that it can be programmed [4] and all the integrals can then be obtained automatically.

Complement 5.1

The steps that lead to Eq. (59) and Eq. (60) might pose some questions when $n \leq p$, as for this case the Gamma function has poles. The other question is how are these results related to those of section 4? We will just give an example that illustrates this relation and shows that the final result in Eq. (61) is correct.

Consider, in the notation of Eq. (54), the integral

$$I_2^{\mu\nu} \equiv \int \frac{d^d k}{(2\pi)^d} \frac{k^\mu k^\nu}{[k^2 + 2k \cdot P - M^2 + i\epsilon]^2} \quad (62)$$

that is $n = p = 2$. With the method of section 4 we complete the square and shift the integration momentum $k \rightarrow k - P$. Then

$$I_2^{\mu\nu} = \int \frac{d^d k}{(2\pi)^d} \frac{k^\mu k^\nu}{[k^2 - C + i\epsilon]^2} + \int \frac{d^d k}{(2\pi)^d} \frac{P^\mu P^\nu}{[k^2 - C + i\epsilon]^2} \quad (63)$$

where we have used the fact that the odd terms in k vanish. We obtain therefore,

$$I_2^{\mu\nu} = \frac{1}{d} g^{\mu\nu} I_{1,2} + P^\mu P^\nu I_{0,2} \quad (64)$$

Now we use Eq. (51) and the properties of the Γ function (see section 6) to obtain

$$I_{0,2} = \frac{i}{16\pi^2} [\Delta_\epsilon - \ln C + O(\epsilon)], \quad \frac{1}{d} I_{1,2} = \frac{i}{16\pi^2} \frac{C}{2} [\Delta_\epsilon + 1 - \ln C + O(\epsilon)] \quad (65)$$

where

$$\Delta_\epsilon = \frac{2}{\epsilon} - \gamma + \ln 4\pi \quad (66)$$

Putting everything together we finally obtain,

$$I_2^{\mu\nu} = \frac{i}{16\pi^2} \frac{1}{2} [C g^{\mu\nu} (\Delta_\epsilon + 1 - \ln C) + 2(\Delta_\epsilon - \ln C) P^\mu P^\nu] + O(\epsilon) \quad (67)$$

We now use Eq. (61) that for our case reads

$$I_2^{\mu\nu} = \frac{i}{16\pi^2} \frac{(4\pi)^{\epsilon/2}}{\Gamma(2)} \int_0^\infty \frac{dt}{(2t)^2} t^{-1+\epsilon/2} \frac{\partial}{\partial P_\mu} \frac{\partial}{\partial P_\nu} e^{-tC} \quad (68)$$

Now

$$\frac{\partial}{\partial P_\mu} \frac{\partial}{\partial P_\nu} e^{-tC} = [(-2t)g^{\mu\nu} + (-2t)^2 P^\mu P^\nu] e^{-tC} \quad (69)$$

and therefore

$$\begin{aligned} I_2^{\mu\nu} &= \frac{i}{16\pi^2} (4\pi)^{\epsilon/2} \left[-\frac{1}{2} g^{\mu\nu} \int_0^\infty dt t^{-2+\epsilon/2} e^{-tC} + P^\mu P^\nu \int_0^\infty dt t^{-1+\epsilon/2} e^{-tC} \right] \\ &= \frac{i}{16\pi^2} (4\pi)^{\epsilon/2} \left[-\frac{1}{2} g^{\mu\nu} C^{1-\epsilon/2} \Gamma(-1 + \frac{\epsilon}{2}) + P^\mu P^\nu C^{-\epsilon/2} \Gamma(\frac{\epsilon}{2}) \right] \\ &= \frac{i}{16\pi^2} \frac{1}{2} [C g^{\mu\nu} (\Delta_\epsilon + 1 - \ln C) + 2(\Delta_\epsilon - \ln C) P^\mu P^\nu] + O(\epsilon) \end{aligned} \quad (70)$$

where we have used the definition of the Γ function, Eq. (72). This coincides exactly with what we have obtained before in Eq. (67).

6 Γ function and useful relations

The Γ function is defined by the integral

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt \quad (71)$$

or equivalently

$$\int_0^\infty t^{z-1} e^{-\mu t} dt = \mu^{-z} \Gamma(z) \quad (72)$$

The function $\Gamma(z)$ has the following important properties

$$\begin{aligned} \Gamma(z+1) &= z\Gamma(z) \\ \Gamma(n+1) &= n! \end{aligned} \quad (73)$$

Another related function is the logarithmic derivative of the Γ function, with the properties,

$$\psi(z) = \frac{d}{dz} \ln \Gamma(z) \quad (74)$$

$$\psi(1) = -\gamma \quad (75)$$

$$\psi(z+1) = \psi(z) + \frac{1}{z} \quad (76)$$

where γ is the Euler constant. The function $\Gamma(z)$ has poles for $z = 0, -1, -2, \dots$. Near the pole $z = -m$ we have ($\epsilon \rightarrow 0$)

$$\Gamma(-m + \epsilon) = \frac{(-1)^m}{m!} \frac{1}{\epsilon} + \frac{(-1)^m}{m!} \psi(m+1) + O(\epsilon) \quad (77)$$

From this we conclude that when $\epsilon \rightarrow 0$

$$\Gamma\left(\frac{\epsilon}{2}\right) = \frac{2}{\epsilon} + \psi(1) + O(\epsilon) \quad \Gamma\left(-n + \frac{\epsilon}{2}\right) = \frac{(-1)^n}{n!} \left[\frac{2}{\epsilon} + \psi(n+1) \right] \quad (78)$$

For positive integers the function $\Gamma(z)$ has no poles. But as we have to expand everything up to order ϵ , before making $\epsilon \rightarrow 0$, we need the expansion near the positive integers. Using the definition in Eq. (74) we get for a general n , up to order ϵ

$$\Gamma(n + \epsilon) = \Gamma(n) + \Gamma(n)\psi(n)\epsilon \quad (79)$$

giving, in particular,

$$\Gamma\left(1 + \frac{\epsilon}{2}\right) = 1 - \gamma \frac{\epsilon}{2} + O(\epsilon^2) \quad (80)$$

Using these results we can expand our integrals in powers of ϵ and separate the divergent and finite parts. For instance for the one of the integrals of the self-energy,

$$\begin{aligned} I_{0,2} &= \frac{i}{(4\pi)^2} \left(\frac{4\pi}{C}\right)^{\frac{\epsilon}{2}} \Gamma\left(\frac{\epsilon}{2}\right) \\ &= \frac{i}{16\pi^2} \left[\frac{2}{\epsilon} - \gamma + \ln 4\pi - \ln C + O(\epsilon) \right] \\ &= \frac{i}{16\pi^2} [\Delta_\epsilon - \ln C + O(\epsilon)] \end{aligned} \quad (81)$$

where we have introduced the notation

$$\Delta_\epsilon = \frac{2}{\epsilon} - \gamma + \ln 4\pi \quad (82)$$

for a combination that will appear in all expressions. In a similar way,

$$\begin{aligned} I_{1,2} &= \frac{i}{(4\pi)^2} (-1) \left(\frac{4\pi}{C}\right)^{\frac{\epsilon}{2}} C \frac{\Gamma\left(3 - \frac{\epsilon}{2}\right) \Gamma\left(-1 + \frac{\epsilon}{2}\right)}{\Gamma\left(2 - \frac{\epsilon}{2}\right) \Gamma(2)} \\ &= \frac{i}{(4\pi)^2} 2C \left[\Delta_\epsilon + \frac{1}{2} - \ln C \right] + O(\epsilon) \end{aligned} \quad (83)$$

7 Explicit formulæ for the 1-loop integrals

Although we have presented in the previous sections the general formulæ for all the integrals that appear in 1-loop, Eqs. (51) and (61), in practice it is convenient to have

expressions for the most important cases with the expansion on the ϵ already done. The results presented below were generated with the `Mathematica` package `OneLoop` [4] from the general expressions. In these results the integration on the Feynman parameters has still to be done (see Eq. (53)). This is in general a difficult problem and we will present in section 9 an alternative way of expressing these integrals more convenient for a numerical evaluation.

7.1 Tadpole integrals

With the definitions of Eqs. (51) and (61) we get

$$\begin{aligned}
I_{0,1} &= \frac{i}{16\pi^2} C(1 + \Delta_\epsilon - \ln C) \\
I_1^\mu &= 0 \\
I_1^{\mu\nu} &= \frac{i}{16\pi^2} \frac{1}{8} C^2 g^{\mu\nu} (3 + 2\Delta_\epsilon - 2 \ln C)
\end{aligned} \tag{84}$$

where for the *tadpole* integrals

$$P = 0 \quad ; \quad C = m^2 \tag{85}$$

because there are no Feynman parameters and there is only one mass. In this case the above results are final.

7.2 Self-Energy integrals

For the integrals with two denominators we get,

$$\begin{aligned}
I_{0,2} &= \frac{i}{16\pi^2} (\Delta_\epsilon - \ln C) \\
I_2^\mu &= \frac{i}{16\pi^2} (-\Delta_\epsilon + \ln C) P^\mu \\
I_2^{\mu\nu} &= \frac{i}{16\pi^2} \frac{1}{2} \left[C g^{\mu\nu} (1 + \Delta_\epsilon - \ln C) + 2(\Delta_\epsilon - \ln C) P^\mu P^\nu \right] \\
I_2^{\mu\nu\alpha} &= \frac{i}{16\pi^2} \frac{1}{2} \left[-C g^{\mu\nu} (1 + \Delta_\epsilon - \ln C) P^\alpha - C g^{\nu\alpha} (1 + \Delta_\epsilon - \ln C) P^\mu \right. \\
&\quad \left. - C g^{\mu\alpha} (1 + \Delta_\epsilon - \ln C) P^\nu - 2(\Delta_\epsilon - \ln C) P^\alpha P^\mu P^\nu \right]
\end{aligned} \tag{86}$$

where, with the notation and conventions of Fig. (1), we have

$$P^\mu = x r_1^\mu \quad ; \quad C = x^2 r_1^2 + (1-x) m_2^2 + x m_1^2 - x r_1^2 \tag{87}$$

7.3 Triangle integrals

For the integrals with three denominators we get,

$$\begin{aligned}
I_{0,3} &= \frac{i}{16\pi^2} \frac{-1}{2C} \\
I_3^\mu &= \frac{i}{16\pi^2} \frac{1}{2C} P^\mu \\
I_3^{\mu\nu} &= \frac{i}{16\pi^2} \frac{1}{4C} \left[C g^{\mu\nu} (\Delta_\epsilon - \ln C) - 2P^\mu P^\nu \right] \\
I_3^{\mu\nu\alpha} &= \frac{i}{16\pi^2} \frac{1}{4C} \left[C g^{\mu\nu} (-\Delta_\epsilon + \ln C) P^\alpha + C g^{\nu\alpha} (-\Delta_\epsilon + \ln C) P^\mu \right. \\
&\quad \left. + C g^{\mu\alpha} (-\Delta_\epsilon + \ln C) P^\nu + 2P^\alpha P^\mu P^\nu \right] \\
I_3^{\mu\nu\alpha\beta} &= \frac{i}{16\pi^2} \frac{1}{8C} \left[C^2 (1 + \Delta_\epsilon - \ln C) (g^{\mu\alpha} g^{\nu\beta} + g^{\mu\beta} g^{\nu\alpha} + g^{\alpha\beta} g^{\mu\nu}) \right. \\
&\quad + 2C (\Delta_\epsilon - \ln C) (g^{\mu\nu} P^\alpha P^\beta + g^{\nu\beta} P^\alpha P^\mu + g^{\nu\alpha} P^\beta P^\mu + g^{\mu\alpha} P^\beta P^\nu \\
&\quad \left. + g^{\mu\beta} P^\alpha P^\nu + g^{\alpha\beta} P^\mu P^\nu) - 4P^\alpha P^\beta P^\mu P^\nu \right] \tag{88}
\end{aligned}$$

where

$$\begin{aligned}
P^\mu &= x_1 r_1^\mu + x_2 r_2^\mu \\
C &= x_1^2 r_1^2 + x_2^2 r_2^2 + 2x_1 x_2 r_1 \cdot r_2 + x_1 m_1^2 + x_2 m_2^2 \\
&\quad + (1 - x_1 - x_2) m_3^2 - x_1 r_1^2 - x_2 r_2^2 \tag{89}
\end{aligned}$$

7.4 Box integrals

$$\begin{aligned}
I_{0,4} &= \frac{i}{16\pi^2} \frac{1}{6C^2} \\
I_4^\mu &= \frac{i}{16\pi^2} \frac{-1}{6C^2} P^\mu \\
I_4^{\mu\nu} &= \frac{i}{16\pi^2} \frac{-1}{12C^2} \left[C g^{\mu\nu} - 2P^\mu P^\nu \right] \\
I_4^{\mu\nu\alpha} &= \frac{i}{16\pi^2} \frac{1}{12C^2} \left[C (g^{\mu\nu} P^\alpha + g^{\nu\alpha} P^\mu + g^{\mu\alpha} P^\nu) - 2P^\alpha P^\mu P^\nu \right]
\end{aligned}$$

$$\begin{aligned}
I_4^{\mu\nu\alpha\beta} &= \frac{i}{16\pi^2} \frac{1}{24C^2} \left[C^2 (\Delta_\epsilon - \ln C) \left(g^{\mu\alpha} g^{\nu\beta} + g^{\mu\beta} g^{\nu\alpha} + g^{\alpha\beta} g^{\mu\nu} \right) \right. \\
&\quad - 2C \left(g^{\mu\nu} P^\alpha P^\beta + g^{\nu\beta} P^\alpha P^\mu + g^{\nu\alpha} P^\beta P^\mu + g^{\mu\alpha} P^\beta P^\nu \right. \\
&\quad \left. \left. + g^{\mu\beta} P^\alpha P^\nu + g^{\alpha\beta} P^\mu P^\nu \right) + 4P^\alpha P^\beta P^\mu P^\nu \right] \quad (90)
\end{aligned}$$

where

$$\begin{aligned}
P^\mu &= x_1 r_1^\mu + x_2 r_2^\mu + x_3 r_3^\mu \\
C &= x_1^2 r_1^2 + x_2^2 r_2^2 + x_3^2 r_3^2 + 2x_1 x_2 r_1 \cdot r_2 + 2x_1 x_3 r_1 \cdot r_3 + 2x_2 x_3 r_2 \cdot r_3 \\
&\quad + x_1 m_1^2 + x_2 m_2^2 + x_3 m_3^2 + (1 - x_1 - x_2 - x_3) m_4^2 \\
&\quad - x_1 r_1^2 - x_2 r_2^2 - x_3 r_3^2 \quad (91)
\end{aligned}$$

8 Divergent part of 1-loop integrals

When we want to study the renormalization of a given theory it is often convenient to have expressions for the divergent part of the one-loop integrals, with the integration on the Feynman parameters already done. We present here the results for the most important cases. These divergent parts were calculated with the help of the package **OneLoop** [4]. The results are for the functions $\hat{T}_n^{\mu, \mu_2, \dots, \mu_n}$ defined in Eq. (52).

8.1 Tadpole integrals

$$\begin{aligned}
\text{Div} \left[\hat{T}_1 \right] &= \frac{i}{16\pi^2} \Delta_\epsilon m^2 \\
\text{Div} \left[\hat{T}_1^\mu \right] &= 0 \\
\text{Div} \left[\hat{T}_1^{\mu\nu} \right] &= \frac{i}{16\pi^2} \frac{1}{4} \Delta_\epsilon m^4 g^{\mu\nu} \quad (92)
\end{aligned}$$

8.2 Self-Energy integrals

$$\begin{aligned}
\text{Div} \left[\hat{T}_2 \right] &= \frac{i}{16\pi^2} \Delta_\epsilon \\
\text{Div} \left[\hat{T}_2^\mu \right] &= \frac{i}{16\pi^2} \left(-\frac{1}{2} \right) \Delta_\epsilon r_1^\mu \\
\text{Div} \left[\hat{T}_2^{\mu\nu} \right] &= \frac{i}{16\pi^2} \frac{1}{12} \Delta_\epsilon \left[(3m_1^2 + 3m_2^2 - r_1^2) g^{\mu\nu} + 4r_1^\mu r_1^\nu \right]
\end{aligned}$$

$$\begin{aligned} \text{Div} \left[\hat{T}_2^{\mu\nu\alpha} \right] &= \frac{i}{16\pi^2} \left(-\frac{1}{24} \right) \Delta_\epsilon \left[(4m_1^2 + 2m_2^2 - r_1^2) (g^{\mu\nu} r_1^\alpha + g^{\nu\alpha} r_1^\mu + g^{\mu\alpha} r_1^\nu) \right. \\ &\quad \left. + 6 r_1^\alpha r_1^\mu r_1^\nu \right] \end{aligned} \quad (93)$$

8.3 Triangle integrals

$$\begin{aligned} \text{Div} \left[\hat{T}_3 \right] &= 0 \\ \text{Div} \left[\hat{T}_3^\mu \right] &= 0 \\ \text{Div} \left[\hat{T}_3^{\mu\nu} \right] &= \frac{i}{16\pi^2} \frac{1}{4} \Delta_\epsilon g^{\mu\nu} \\ \text{Div} \left[\hat{T}_3^{\mu\nu\alpha} \right] &= \frac{i}{16\pi^2} \left(-\frac{1}{12} \right) \Delta_\epsilon \left[g^{\mu\nu} (r_1^\alpha + r_2^\alpha) + g^{\nu\alpha} (r_1^\mu + r_2^\mu) + g^{\mu\alpha} (r_1^\nu + r_2^\nu) \right] \\ \text{Div} \left[\hat{T}_3^{\mu\nu\alpha\beta} \right] &= \frac{i}{16\pi^2} \frac{1}{48} \Delta_\epsilon \left[(2m_1^2 + 2m_2^2 + 2m_3^2) \left(g^{\mu\alpha} g^{\nu\beta} + g^{\alpha\beta} g^{\mu\nu} + g^{\mu\beta} g^{\nu\alpha} \right) \right. \\ &\quad + g^{\alpha\beta} \left[2r_1^\mu r_1^\nu + r_1^\mu r_2^\nu + (r_1 \leftrightarrow r_2) \right] + g^{\mu\beta} \left[2r_1^\alpha r_1^\nu + r_1^\alpha r_2^\nu + (r_1 \leftrightarrow r_2) \right] \\ &\quad + g^{\nu\beta} \left[2r_1^\alpha r_1^\mu + r_1^\alpha r_2^\mu + (r_1 \leftrightarrow r_2) \right] + g^{\mu\nu} \left[2r_1^\alpha r_1^\beta + r_1^\alpha r_2^\beta + (r_1 \leftrightarrow r_2) \right] \\ &\quad + g^{\mu\alpha} \left[2r_1^\beta r_1^\nu + r_1^\beta r_2^\nu + (r_1 \leftrightarrow r_2) \right] + g^{\nu\alpha} \left[2r_1^\beta r_1^\mu + r_1^\beta r_2^\mu + (r_1 \leftrightarrow r_2) \right] \\ &\quad \left. + (-r_1^2 + r_1 \cdot r_2 - r_2^2) \left(g^{\mu\alpha} g^{\nu\beta} + g^{\alpha\beta} g^{\mu\nu} + g^{\mu\beta} g^{\nu\alpha} \right) \right] \end{aligned} \quad (94)$$

8.4 Box integrals

$$\begin{aligned} \text{Div} \left[\hat{T}_4 \right] &= \text{Div} \left[\hat{T}_4^\mu \right] = \text{Div} \left[\hat{T}_4^{\mu\nu} \right] = \text{Div} \left[\hat{T}_4^{\mu\nu\alpha} \right] = 0 \\ \text{Div} \left[\hat{T}_4^{\mu\nu\alpha\beta} \right] &= \frac{i}{16\pi^2} \frac{1}{24} \Delta_\epsilon \left[g^{\mu\nu} g^{\alpha\beta} + g^{\mu\beta} g^{\alpha\nu} + g^{\mu\alpha} g^{\nu\beta} \right] \end{aligned} \quad (95)$$

9 Passarino-Veltman Integrals

9.1 The general definition

The description of the previous sections works well if one just wants to calculate the divergent part of a diagram or to show the cancellation of divergences in a set of diagrams. If one actually wants to numerically calculate the integrals the task is normally quite

complicated. Except for the *self-energy* type of diagrams the integration over the Feynman parameters is normally quite difficult.

To overcome this problem a scheme was first proposed by Passarino and Veltman [5]. These scheme with the conventions of [6, 7] was latter implemented in the `Mathematica` package `FeynCalc` [7, 8] and, for numerical evaluation, in the `LoopTools` package [9]. The numerical evaluation follows the code developed earlier by van Oldenborgh [10].

We will now describe this scheme. We will write the generic one-loop tensor integral as

$$T_n^{\mu_1 \dots \mu_p} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k \frac{k^{\mu_1} \dots k^{\mu_p}}{D_0 D_1 D_2 \dots D_{n-1}} \quad (96)$$

where we follow for the momenta the conventions of section 2 and Fig. 1 and defined $D_0 \equiv D_n$ and $m_n = m_0$ so that $D_0 = k^2 - m_0^2$ (remember that $r_n \equiv r_0 = 0$). The main difference between this definition and the previous one Eq. (7) is that a factor of $\frac{i}{16\pi^2}$ is taken out. This is because, as we have seen in section 3 these integrals always give that prefactor. So with our new convention that prefactor **has** to included in the end. Factoring out the i has also the convenience of dealing with real functions in many cases.² From all those integrals in Eq. (96) the scalar integrals are, has we have seen, of particular importance and deserve a special notation. It can be shown that there are only four independent such integrals, namely ($4 - d = \epsilon$)

$$A_0(m_0^2) = \frac{(2\pi\mu)^\epsilon}{i\pi^2} \int d^d k \frac{1}{k^2 - m_0^2} \quad (97)$$

$$B_0(r_{10}^2, m_0^2, m_1^2) = \frac{(2\pi\mu)^\epsilon}{i\pi^2} \int d^d k \prod_{i=0}^1 \frac{1}{[(k + r_i)^2 - m_i^2]} \quad (98)$$

$$C_0(r_{10}^2, r_{12}^2, r_{20}^2, m_0^2, m_1^2, m_2^2) = \frac{(2\pi\mu)^\epsilon}{i\pi^2} \int d^d k \prod_{i=0}^2 \frac{1}{[(k + r_i)^2 - m_i^2]} \quad (99)$$

$$D_0(r_{10}^2, r_{12}^2, r_{23}^2, r_{30}^2, r_{20}^2, r_{13}^2, m_0^2, m_1^2, \dots, m_3^2) = \frac{(2\pi\mu)^\epsilon}{i\pi^2} \int d^d k \prod_{i=0}^3 \frac{1}{[(k + r_i)^2 - m_i^2]} \quad (100)$$

where

$$r_{ij}^2 = (r_i - r_j)^2 \quad ; \quad \forall i, j = (0, n-1) \quad (101)$$

Remember that with our conventions $r_0 = 0$ so $r_{i0}^2 = r_i^2$. In all these expressions the $i\epsilon$ part of the denominator factors is suppressed. The general one-loop tensor integrals are not independent. Their decomposition is not unique. We follow the conventions of [7, 9] to write

$$B^\mu \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu \prod_{i=0}^1 \frac{1}{[(k + r_i)^2 - m_i^2]} \quad (102)$$

$$B^{\mu\nu} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu k^\nu \prod_{i=0}^1 \frac{1}{[(k + r_i)^2 - m_i^2]} \quad (103)$$

²The one loop functions are in general complex, but in some cases they can be real. These cases correspond to the situation where cutting the diagram does not corresponding to a kinematically allowed process.

$$C^\mu \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu \prod_{i=0}^2 \frac{1}{[(k+r_i)^2 - m_i^2]} \quad (104)$$

$$C^{\mu\nu} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu k^\nu \prod_{i=0}^2 \frac{1}{[(k+r_i)^2 - m_i^2]} \quad (105)$$

$$C^{\mu\nu\rho} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu k^\nu k^\rho \prod_{i=0}^2 \frac{1}{[(k+r_i)^2 - m_i^2]} \quad (106)$$

$$D^\mu \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu \prod_{i=0}^3 \frac{1}{[(k+r_i)^2 - m_i^2]} \quad (107)$$

$$D^{\mu\nu} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu k^\nu \prod_{i=0}^3 \frac{1}{[(k+r_i)^2 - m_i^2]} \quad (108)$$

$$D^{\mu\nu\rho} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu k^\nu k^\rho \prod_{i=0}^3 \frac{1}{[(k+r_i)^2 - m_i^2]} \quad (109)$$

$$D^{\mu\nu\rho\sigma} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2} \int d^d k k^\mu k^\nu k^\rho k^\sigma \prod_{i=0}^3 \frac{1}{[(k+r_i)^2 - m_i^2]} \quad (110)$$

These integrals can be decomposed in terms of (reducible) functions in the following way:

$$B^\mu = r_1^\mu B_1 \quad (111)$$

$$B^{\mu\nu} = g^{\mu\nu} B_{00} + r_1^\mu r_1^\nu B_{11} \quad (112)$$

$$C^\mu = r_1^\mu C_1 + r_2^\mu C_2 \quad (113)$$

$$C^{\mu\nu} = g^{\mu\nu} C_{00} + \sum_{i=1}^2 r_i^\mu r_j^\nu C_{ij} \quad (114)$$

$$C^{\mu\nu\rho} = \sum_{i=1}^2 (g^{\mu\nu} r_i^\rho + g^{\nu\rho} r_i^\mu + g^{\rho\mu} r_i^\nu) C_{00i} + \sum_{i,j,k=1}^2 r_i^\mu r_j^\nu r_k^\rho C_{ijk} \quad (115)$$

$$D^\mu = \sum_{i=1}^3 r_i^\mu D_i \quad (116)$$

$$D^{\mu\nu} = g^{\mu\nu} D_{00} + \sum_{i,j=1}^3 r_i^\mu r_j^\nu D_{ij} \quad (117)$$

$$D^{\mu\nu\rho} = \sum_{i=1}^3 (g^{\mu\nu} r_i^\rho + g^{\nu\rho} r_i^\mu + g^{\rho\mu} r_i^\nu) D_{00i} + \sum_{i,j,k=1}^3 r_i^\mu r_j^\nu r_k^\rho D_{ijk} \quad (118)$$

$$\begin{aligned} D^{\mu\nu\rho\sigma} &= (g_{\mu\nu} g_{\rho\sigma} + g_{\mu\rho} g_{\nu\sigma} + g_{\mu\sigma} g_{\nu\rho}) D_{0000} \\ &+ \sum_{i,j=1}^3 \left(g^{\mu\nu} r_i^\rho r_j^\sigma + g^{\nu\rho} r_i^\mu r_j^\sigma + g^{\mu\rho} r_i^\nu r_j^\sigma + g^{\mu\sigma} r_i^\nu r_j^\rho \right. \\ &\quad \left. + g^{\nu\sigma} r_i^\mu r_j^\rho + g^{\rho\sigma} r_i^\mu r_j^\nu \right) D_{00ij} \end{aligned} \quad (119)$$

$$+ \sum_{i,j,k,l=1}^3 r_i^\mu r_j^\nu r_k^\rho r_l^\sigma D_{ijkl} \quad (120)$$

All coefficient functions have the same arguments as the corresponding scalar functions and are totally symmetric in their indices. In the `FeynCalc` [11] package one generic notation is used,

$$\text{PaVe} [i, j, \dots, \{r_{10}^2, r_{12}^2, \dots\}, \{m_0^2, m_1^2, \dots\}] \quad (121)$$

for instance

$$B_{11}(r_{10}^2, m_0^2, m_1^2) = \text{PaVe} [1, 1, \{r_{10}^2\}, \{m_0^2, m_1^2\}] \quad (122)$$

All these coefficient functions are not independent and can be reduced to the scalar functions. `FeynCalc` provides the command `PaVeReduce[...]` to accomplish that. This is very useful if one wants to check for cancellation of divergences or for gauge invariance where a number of diagrams have to cancel.

9.2 The divergences

The package `LoopTools` provides ways to numerically check for the cancellation of divergences. However it is useful to know the divergent part of the Passarino-Veltman integrals. Only a small number of these integrals are divergent. They are

$$\text{Div} [A_0(m_0^2)] = \Delta_\epsilon m_0^2 \quad (123)$$

$$\text{Div} [B_0(r_{10}^2, m_0^2, m_1^2)] = \Delta_\epsilon \quad (124)$$

$$\text{Div} [B_1(r_{10}^2, m_0^2, m_1^2)] = -\frac{1}{2} \Delta_\epsilon \quad (125)$$

$$\text{Div} [B_{00}(r_{10}^2, m_0^2, m_1^2)] = \frac{1}{12} \Delta_\epsilon (3m_0^2 + 3m_1^2 - r_{10}^2) \quad (126)$$

$$\text{Div} [B_{11}(r_{10}^2, m_0^2, m_1^2)] = \frac{1}{3} \Delta_\epsilon \quad (127)$$

$$\text{Div} [C_{00}(r_{10}^2, r_{12}^2, r_{20}^2, m_0^2, m_1^2, m_2^2)] = \frac{1}{4} \Delta_\epsilon \quad (128)$$

$$\text{Div} [C_{001}(r_{10}^2, r_{12}^2, r_{20}^2, m_0^2, m_1^2, m_2^2)] = -\frac{1}{12} \Delta_\epsilon \quad (129)$$

$$\text{Div} [C_{002}(r_{10}^2, r_{12}^2, r_{20}^2, m_0^2, m_1^2, m_2^2)] = -\frac{1}{12} \Delta_\epsilon \quad (130)$$

$$\text{Div} [D_{0000}(r_{10}^2, \dots, m_0^2, \dots)] = \frac{1}{24} \Delta_\epsilon \quad (131)$$

$$(132)$$

These results were obtained with the package `LoopTools`, after reducing to the scalar integrals with the command `PaVeReduce`, but they can be verified by comparing with our results of section 8, after factoring out the $i/(16\pi^2)$.

9.3 Useful results for PV integrals

Although the PV approach is intended primarily to be used numerically there are situations where one wants to have explicit results. These can be useful to check cancellation of

divergences or because in some simple cases the integrals can be done analytically. We note that as our conventions for the momenta are the same in sections 9 and 7 one can read immediately the integral representation of the PV in terms of the Feynman parameters just by comparing both expressions, not forgetting to take out the $i/(16\pi^2)$ factor. For instance, from Eq. (114) for $C^{\mu\nu}$ and Eq. (88) for $I_3^{\mu\nu}$ we get

$$C_{12}(r_1^2, r_{12}^2, r_2^2, m_0^2, m_1^2, m_2^2) = -\Gamma(3)\frac{2}{4} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \frac{x_1 x_2}{C} \quad (133)$$

with

$$C = x_1^2 r_1^2 + x_2^2 r_2^2 + x_1 x_2 (r_1^2 + r_2^2 - r_{12}^2) + x_1 m_1^2 + x_2 m_2^2 \\ + (1 - x_1 - x_2) m_0^2 - x_1 r_1^2 - x_2 r_2^2 \quad (134)$$

9.3.1 Explicit expression for A_0

This integral is trivial. There is no Feynman parameter and the integral can be read from Eq. (84). We get, after factoring out the $i/(16\pi^2)$,

$$A_0(m^2) = m^2 \left(\Delta_\epsilon + 1 - \ln \frac{m^2}{\mu^2} \right) \quad (135)$$

9.3.2 Explicit expressions for the B functions

Function B_0

The general form of the integral $B_0(p^2, m_1^2, m_2^2)$ can be read from Eq. (86). We obtain

$$B_0(p^2, m_0^2, m_1^2) = \Delta_\epsilon - \int_0^1 dx \ln \left[\frac{-x(1-x)p^2 + xm_1^2 + (1-x)m_0^2}{\mu^2} \right] \quad (136)$$

From this expression one can easily get the following results,

$$B_0(0, m_0^2, m_1^2) = \Delta_\epsilon + 1 - \frac{m_0^2 \ln \frac{m_0^2}{\mu^2} - m_1^2 \ln \frac{m_1^2}{\mu^2}}{m_0^2 - m_1^2} \quad (137)$$

$$B_0(0, m_0^2, m_1^2) = \frac{A_0(m_0^2) - A_0(m_1^2)}{m_0^2 - m_1^2} \quad (138)$$

$$B_0(0, m^2, m^2) = \Delta_\epsilon - \ln \frac{m^2}{\mu^2} = \frac{A_0(m^2)}{m^2} - 1 \quad (139)$$

$$B_0(m^2, 0, m^2) = \Delta_\epsilon + 2 - \ln \frac{m^2}{\mu^2} = \frac{A_0(m^2)}{m^2} + 1 \quad (140)$$

$$B_0(0, 0, m^2) = \Delta_\epsilon + 1 - \ln \frac{m^2}{\mu^2} = \frac{A_0(m^2)}{m^2} \quad (141)$$

Function B'_0

The derivative of the B_0 function with respect to p^2 appears many times. From Eq. (136) one can derive an integral representation,

$$B'_0(p^2, m_0^2, m_1^2) = \int_0^1 dx \frac{x(1-x)}{-p^2x(1-x) + xm_1^2 + (1-x)m_0^2} \quad (142)$$

An important particular case corresponds to $B'_0(m^2, m_0^2, m^2)$ that appears in the self-energy of the electron. In this case m is the electron mass and $m_0 = \lambda$ is the photon mass that one has to introduce to regularize the IR divergent integral. The integral in this case reduces to

$$\begin{aligned} B'_0(m^2, \lambda^2, m^2) &= \int_0^1 dx \frac{x(1-x)}{m^2x^2 + (1-x)\lambda^2} \\ &= -\frac{1}{m^2} - \frac{1}{2m^2} \ln \frac{\lambda^2}{m^2} \end{aligned} \quad (143)$$

It is clear that in the limit $\lambda \rightarrow 0$ this integral diverges. Another limit that it is useful (for instance is needed in the vacuum polarization, see section 10.1), is

$$B'_0(0, m^2, m^2) = \frac{1}{6m^2} \quad (144)$$

that can be easily obtained from Eq. (142).

Function B_1

The explicit expression can be read from Eq. (86). We have

$$B_1(p^2, m_0^2, m_1^2) = -\frac{1}{2}\Delta_\epsilon + \int_0^1 dx x \ln \left[\frac{-x(1-x)p^2 + xm_1^2 + (1-x)m_0^2}{\mu^2} \right] \quad (145)$$

For $p^2 = 0$ this integral can be easily evaluated to give

$$B_1(0, m_0^2, m_1^2) = -\frac{1}{2}\Delta_\epsilon + \frac{1}{2} \ln \left(\frac{m_0^2}{\mu^2} \right) + \frac{-3 + 4t - t^2 - 4t \ln t + 2t^2 \ln t}{4(-1+t)^2} \quad (146)$$

where we defined

$$t = \frac{m_1^2}{m_0^2} \quad (147)$$

From Eq. (146) one can shown that even for $p^2 = 0$ B_1 is **not** a symmetric function of the masses,

$$B_1(p^2, m_0^2, m_1^2) \neq B_1(p^2, m_1^2, m_0^2) \quad (148)$$

As this might appear strange let us show with one example how the coefficient functions are tied to our conventions about the order of the momenta and Feynman parameters. Let

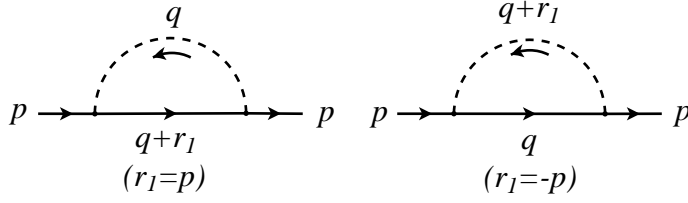


Figure 5:

us consider the contribution to the self-energy of a fermion of mass m_f of the exchange of a scalar with mass m_s . We can consider the two choices in Fig. 5,

Now with the first choice (diagram on the left of Fig. 5) we have

$$\begin{aligned}
-i\Sigma_1 &= \frac{i}{16\pi^2} \left[(\not{p} + m_f) B_0(p^2, m_s^2, m_f^2) + \not{p} B_1(p^2, m_s^2, m_f^2) \right] \\
&= \frac{i}{16\pi^2} \left[\not{p} (B_0(p^2, m_s^2, m_f^2) + B_1(p^2, m_s^2, m_f^2)) + m_f B_0(p^2, m_s^2, m_f^2) \right] \quad (149)
\end{aligned}$$

while with the second choice we have

$$-i\Sigma_2 = \frac{i}{16\pi^2} \left[-\not{p} B_1(p^2, m_f^2, m_s^2) + m_f B_0(p^2, m_f^2, m_s^2) \right] \quad (150)$$

How can these two expressions be equal? The reason has precisely to do with the non symmetry of B_1 with respect to the mass entries. In fact from Eq. (145) we have

$$\begin{aligned}
B_1(p^2, m_0^2, m_1^2) &= -\frac{1}{2}\Delta_\epsilon + \int_0^1 dx x \ln \left[\frac{-x(1-x)p^2 + xm_1^2 + (1-x)m_0^2}{\mu^2} \right] \\
&= -\frac{1}{2}\Delta_\epsilon + \int_0^1 dx (1-x) \ln \left[\frac{-x(1-x)p^2 + (1-x)m_1^2 + xm_0^2}{\mu^2} \right] \\
&= -\frac{1}{2}\Delta_\epsilon + (\Delta_\epsilon - B_0(p^2, m_1^2, m_0^2)) - \left(\frac{1}{2}\Delta_\epsilon + B_1(p^2, m_1^2, m_0^2) \right) \\
&= - (B_0(p^2, m_1^2, m_0^2) + B_1(p^2, m_1^2, m_0^2)) \quad (151)
\end{aligned}$$

where we have changed variables ($x \rightarrow 1-x$) in the integral and used the definitions of B_0 and B_1 . We have then, remembering that $B_0(p^2, m_s^2, m_f^2) = B_0(p^2, m_f^2, m_s^2)$,

$$B_1(p^2, m_f^2, m_s^2) = - (B_0(p^2, m_s^2, m_f^2) + B_1(p^2, m_s^2, m_f^2)) \quad (152)$$

and therefore Eqs. (149) and (150) are equivalent.

9.3.3 Explicit expressions for the C functions

In Eq. (133) we have already given the general form of C_{12} . The other functions are very similar. In the following we just present the results for the particular case of $p^2 = 0$. This case is important in many situations where it is a good approximation to neglect the external momenta in comparison with the masses of the particles in the loop. We also warn the reader that the coefficient functions C_i, C_{ij} obtained from `LoopTools` are not well defined in this limit. Hence there is some utility in given them here.

Function C_0

$$\begin{aligned}
C_0(0, 0, 0, m_0^2, m_1^2, m_2^2) &= -\Gamma(3) \frac{1}{2} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \frac{1}{x_1 m_1^2 + x_2 m_2^2 + (1-x_1-x_2)m_0^2} \\
&= -\frac{1}{m_0^2} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \frac{1}{x_1 t_1 + x_2 t_2 + (1-x_1-x_2)} \\
&= -\frac{1}{m_0^2} \frac{-t_1 \ln t_1 + t_1 t_2 \ln t_1 + t_2 \ln t_2 - t_1 t_2 \ln t_2}{(-1+t_1)(t_1-t_2)(-1+t_2)} \quad (153)
\end{aligned}$$

where

$$t_1 = \frac{m_1^2}{m_0^2} \quad ; \quad t_2 = \frac{m_2^2}{m_0^2} \quad (154)$$

Using the properties of the logarithms one can show that in this limit C_0 is a symmetric function of the masses. This expression is further simplified when two of the masses are equal, as it happens in the $\mu \rightarrow e\gamma$ problem. Then $t = t_1 = t_2$,

$$C_0(0, 0, 0, m_0^2, m_1^2, m_1^2) = -\frac{1}{m_0^2} \frac{-1+t-\ln t}{(-1+t)^2} \quad (155)$$

in agreement with Eq.(20) of [12]. In the case of equal masses for all the loop particles we have

$$C_0(0, 0, 0, m_0^2, m_0^2, m_0^2) = -\frac{1}{2m_0^2} \quad (156)$$

Before we close this section on C_0 there is another particular case when it is useful to have an explicit case for it. This is the case when it is IR divergent as in the QED vertex. The function needed is $C_0(m^2, m^2, 0, m^2, \lambda^2, m^2)$. Using the definition we have

$$\begin{aligned}
C_0(m^2, m^2, 0, m^2, \lambda^2, m^2) &= -\int_0^1 dx_1 \int_0^{1-x_1} dx_2 \frac{1}{m^2(1-2x_1+x_1^2) + x_1 \lambda^2} \\
&= -\int_0^1 dx_1 \frac{1-x_1}{m^2(1-x_1)^2 + x_1 \lambda^2} \\
&= -\int_0^1 dx \frac{x}{m^2 x^2 + (1-x)\lambda^2} \\
&= \frac{1}{2m^2} \ln \frac{\lambda^2}{m^2} = -B'_0(m^2, \lambda^2, m^2) - \frac{1}{m^2} \quad (157)
\end{aligned}$$

We have verified numerically, using `LoopTools`[9, 10], that Eqs. (157), (143) and (144) are verified.

Function C_{00}

$$\begin{aligned}
C_{00}(0, 0, 0, m_0^2, m_1^2, m_2^2) &= \Gamma(3) \frac{1}{4} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \left[\Delta_\epsilon - \ln \left(\frac{C}{\mu^2} \right) \right] \\
&= \frac{1}{4} \Delta_\epsilon - \frac{1}{2} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \ln \left[\frac{x_1 m_1^2 + x_2 m_2^2 + (1-x_1-x_2) m_0^2}{\mu^2} \right] \\
&= \frac{1}{4} \left(\Delta_\epsilon - \ln \frac{m_0^2}{\mu^2} \right) + \frac{3}{8} - \frac{t_1^2}{4(t_1-1)(t_1-t_2)} \ln t_1 \\
&\quad + \frac{t_2^2}{4(t_2-1)(t_1-t_2)} \ln t_2
\end{aligned} \tag{158}$$

where, as before

$$t_1 = \frac{m_1^2}{m_0^2} \quad ; \quad t_2 = \frac{m_2^2}{m_0^2} \tag{159}$$

Using the properties of the logarithms one can show that in this limit C_{00} is a symmetric function of the masses. This expression is further simplified when two of the masses are equal. Then $t = t_1 = t_2$,

$$\begin{aligned}
C_{00}(0, 0, 0, m_0^2, m_1^2, m_1^2) &= \frac{1}{4} \left(\Delta_\epsilon - \ln \frac{m_0^2}{\mu^2} \right) - \frac{-3 + 4t - t^2 - 4t \ln t + 2t^2 \ln t}{8(t-1)^2} \\
&= -\frac{1}{2} B_1(0, m_0^2, m_1^2)
\end{aligned} \tag{160}$$

Functions C_i and C_{ij}

We recall that the definition of the coefficient functions is not unique, it is tied to a particular convention for assigning the loop momenta and Feynman parameters, as shown in Fig. 1. For the particular case of the C functions we show our conventions in Fig. 6.

With the same techniques we obtain,

$$\begin{aligned}
C_1(0, 0, 0, m_0^2, m_1^2, m_2^2) &= \frac{1}{m_0^2} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \frac{x_1}{x_1 t_1 + x_2 t_2 + (1-x_1-x_2)} \\
&= -\frac{1}{m_0^2} \left[\frac{t_1}{2(-1+t_1)(t_1-t_2)} - \frac{t_1(t_1-2t_2+t_1 t_2)}{2(-1+t_1)^2(t_1-t_2)^2} \ln t_1 \right. \\
&\quad \left. + \frac{t_2^2 - 2t_1 t_2^2 + t_1^2 t_2^2}{2(-1+t_1)^2(t_1-t_2)^2(-1+t_2)} \ln t_2 \right]
\end{aligned} \tag{161}$$

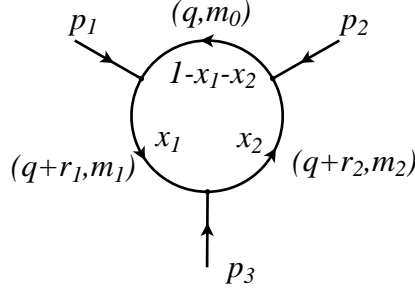


Figure 6:

$$\begin{aligned}
C_2(0, 0, 0, m_0^2, m_1^2, m_2^2) &= \frac{1}{m_0^2} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \frac{x_2}{x_1 t_1 + x_2 t_2 + (1-x_1-x_2)} \\
&= -\frac{1}{m_0^2} \left[-\frac{t_2}{2(t_1-t_2)(-1+t_2)} + \frac{\ln t_1}{2(-1+t_1)(-1+t_2)^2} \right. \\
&\quad \left. + \frac{2t_1 t_2 - 2t_1^2 t_2 - t_2^2 + t_1^2 t_2^2}{2(-1+t_1)(t_1-t_2)^2(-1+t_2)^2} \ln \left(\frac{t_1}{t_2} \right) \right] \quad (162)
\end{aligned}$$

$$C_{ij}(0, 0, 0, m_0^2, m_1^2, m_2^2) = -\frac{1}{m_0^2} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \frac{x_i x_j}{x_1 t_1 + x_2 t_2 + (1-x_1-x_2)} \quad (163)$$

where we have not written explicitly the C_{ij} for $i, j = 1, 2$ because they are rather lengthy. However a simple **Fortran** program can be developed [4] to calculate all the three point functions in the zero external limit case. This is useful because in this case some of the functions from **LoopTools** will fail. Notice that the C_i and C_{ij} functions are not symmetric in their arguments. This a consequence of their non-uniqueness, they are tied to a particular convention. This is very important when ones compares with other results. However using their definition one can get some relations. For instance we can show

$$C_1(0, 0, 0, m_0^2, m_1^2, m_2^2) = C_1(0, 0, 0, m_2^2, m_1^2, m_0^2) \quad (164)$$

$$\begin{aligned}
C_2(0, 0, 0, m_0^2, m_1^2, m_2^2) &= -C_0(0, 0, 0, m_2^2, m_1^2, m_0^2) - C_1(0, 0, 0, m_2^2, m_1^2, m_0^2) \\
&\quad - C_2(0, 0, 0, m_2^2, m_1^2, m_0^2) \quad (165)
\end{aligned}$$

In the limit $m_1 = m_2$ we get the simple expressions,

$$\begin{aligned}
C_1(0, 0, 0, m_0^2, m_1^2, m_1^2) &= C_2(0, 0, 0, m_0^2, m_1^2, m_1^2) \\
&= -\frac{1}{m_0^2} \frac{3 - 4t + t^2 + 2 \ln t}{4(-1+t)^3} \quad (166)
\end{aligned}$$

$$C_{11}(0, 0, 0, m_0^2, m_1^2, m_1^2) = C_{22}(0, 0, 0, m_0^2, m_1^2, m_1^2) = 2 C_{12}(0, 0, 0, m_0^2, m_1^2, m_1^2)$$

$$= -\frac{1}{m_0^2} \frac{-11 + 18t - 9t^2 + 2t^3 - 6 \ln t}{18(-1+t)^4} \quad (167)$$

in agreement with Eqs. (21-22) of [12]. The case of masses equal gives

$$C_1(0, 0, 0, m_0^2, m_0^2, m_0^2) = C_2(0, 0, 0, m_0^2, m_0^2, m_0^2) = \frac{1}{6m_0^2} \quad (168)$$

$$C_{11}(0, 0, 0, m_0^2, m_0^2, m_0^2) = C_{22}(0, 0, 0, m_0^2, m_0^2, m_0^2) = -\frac{1}{12m_0^2} \quad (169)$$

$$C_{12}(0, 0, 0, m_0^2, m_0^2, m_0^2) = -\frac{1}{24m_0^2} \quad (170)$$

9.3.4 The package PVzem

As we said before, in many situations it is a good approximation to neglect the external momenta. In this case, the loop functions are easier to evaluate and one approach is for each problem to evaluate them. However our approach here is more in the direction of automatically evaluating the one-loop amplitudes. If one does that with the use of FeynCalc, as we have been doing, then the result is given in terms of standard functions that can be numerically evaluated with the package LoopTools. However this package has problems with this limit. This is because this limit is unphysical. Let us illustrate this point calculating the functions $C_1(m^2, 0, 0, m_S^2, m_F^2, m_F^2)$ and $C_2(m^2, 0, 0, m_S^2, m_F^2, m_F^2)$ for $m_B = 100$ GeV, $m_F = 80$ GeV and m_2 ranging from 10^{-6} to 100 GeV. To better illustrate our point we show two plots with different scales on the axis.

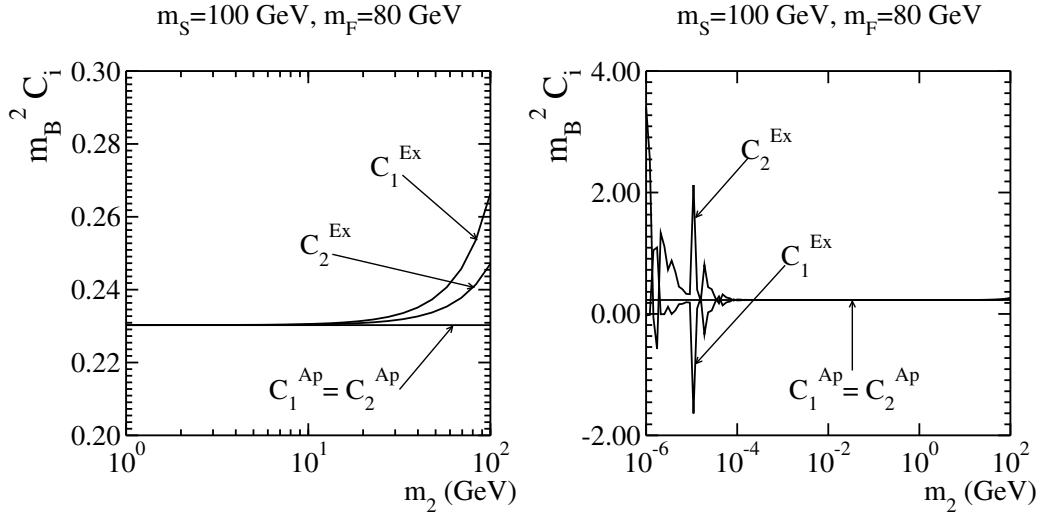


Figure 7:

In these plots, C_i^{Ex} are the exact C_i functions calculated with LoopTools and C_i^{Ap} are the C_i calculated in the zero momenta limit. We can see that only for external momenta (in this case corresponding to the mass m_2) close enough to the masses of the particles in the

loop, the exact result deviates from the approximate one. However for very small values of the external momenta, LoopTools has numerical problems as shown in the right panel of Fig. 7. To overcome this problem I have developed a Fortran package that evaluates all the C functions in the zero external momenta limit. There are no restrictions on the masses being equal or different and the conventions are the same as in FeynCalc and LoopTools, for instance,

$$c12zem(m02, m12, m22) = c0i(cc12, 0, 0, 0, m02, m12, m22) \quad (171)$$

where $c0i(cc12, \dots)$ is the LoopTools notation and $c12zem(\dots)$ is the notation of my package, called PVzem. It can be obtained from the address indicated in Ref.[4]. The approximate functions shown in Fig. 7 were calculated using that package. We include here the Fortran code used to produce that figure.

```

*****
*
*           Program LoopToolsExample
*
*   This program calculates the values used in the plots
*   of Figure 20. For the exact results the LoopTools
*   package was used. The package PVzem was used for the
*   approximate results.
*
*           Version of 14/05/2012
*
*   Author: Jorge C. Romao
*   e-mail: jorge.romao@ist.utl.pt
*****

program LoopToolsExample
implicit none

*
* LoopTools has to be used with FORTRAN programs with the
* extension .F in order to have the header file "looptools.h"
* preprocessed. This file includes all the definitions used
* by LoopTools.
*
* Functions cizem and c2zem are provided by the package PVzem.
*

#include "looptools.h"

integer i
real*8 m2,mF2,mS2,m
real*8 lgmmin,lgmmax,lgm,step
real*8 rc1,rc2
real*8 cizem,c2zem

mS2=100.d0**2
mF2=80.d0**2

*
* Initialize LoopTools. See the LoopTools manual for further

```

```

* details. There you can also learn how to set the scale MU
* and how to handle the UR and IR divergences.
*

    call ltini

    lgmmax=log10(100.d0)
    lgmmin=log10(1.d-6)
    step=(lgmmax-lgmmin)/100.d0
    lgm=lgmmin-step

    open(10,file=plot.dat,status=unknown)

    do i=1,101
        lgm=lgm+step
        m=10.d0**lgm
        m2=m**2
*
* In LoopTools the c0i(...) are complex functions. For the
* kinematics chosen here they are real, so we take the real
* part for comparison.
*

        rc1=dbl(c0i(cc1,m2,0.d0,0.d0,mS2,mF2,mF2))
        rc2=dbl(c0i(cc2,m2,0.d0,0.d0,mS2,mF2,mF2))
        write(10,100)m,rc1*mS2,rc2*mS2,c1zem(mS2,mF2,mF2)*mS2,
&          c2zem(mS2,mF2,mF2)*mS2
        enddo

100  format(5(e22.14))

    call ltexi

    end

***** End of Program LoopToolsExample.F *****

```

When the above program is compiled, the location of the header file `looptools.h` must be known by the compiler. This is best achieved by using a `Makefile`. We give below, as an example, the one that was used with the above program. Depending on the installation details of `LoopTools` the paths might be different.

```

FC          =
LT          = /usr/local/lib/LoopTools
FFLAGS     = -c -O -I$(LT)/include
LDFLAGS    =
LINKER     = $(FC)

LIB        = -L$(LT)/lib
LIBS      = -looptools

.f.o:
    $(FC) $(FFLAGS) $*.F

files     = LoopToolsExample.o PVzem.o

all:      $(files)
          $(LINKER) $(LDFLAGS) -o Example $(files) $(LIB) $(LIBS)

```

9.3.5 Explicit expressions for the D functions

Function D_0

The various D functions can be calculated in a similar way. However they are rather lengthy and have to be handled numerically [4]. Here we just give D_0 for the equal masses case.

$$\begin{aligned}
 D_0(0, \dots, 0, m^2, m^2, m^2, m^2) &= \Gamma(4) \frac{1}{6} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \int_0^{1-x_1-x_2} dx_3 \frac{1}{(m^2)^2} \\
 &= \frac{1}{m^4} \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \int_0^{1-x_1-x_2} dx_3 \\
 &= \frac{1}{6m^4} \tag{172}
 \end{aligned}$$

10 Examples of 1 -loop calculations with PV functions

In this section we will work out in detail a few examples of one-loop calculations using the FeynCalc package and the Passarino-Veltman scheme.

10.1 Vacuum Polarization in QED

We have done this example in section 10.1 using the techniques described in sections 3, 4 and 5. Now we will use FeynCalc. The first step is to write the Mathematica program ³. We list it below:

³One should check which version of Mathematica and FeynCalc is used, as conventions may change. We will indicate in which version these programs were verified. Also the output may change as Mathematica can order the terms differently. We will try to maintain in my web page [13] a version of the programs as updated as possible.


```

(***** Program VacPol.m *****)
(*
Version compatible with FeynCalc 9.2.0

Date: 01/06/2017
Author: Jorge C. Romao
email: jorge.romao@tecnico.ulisboa.pt
*)

(* First input FeynCalc *)
(* Uncomment below if you want to call from this program. If open a new
mathematica notebook and load FeynCalc from there you should not load
it again
*)
(*
<< FeynCalc '
*)

(* Now write the numerator of the Feynman diagram. We define the
constant

C=alpha/(4 pi)

I also use the FCE notation available since FeynCalc 6. See manual
for explanations.

*)

num:= - C Tr[GA[mu] . (GS[q] + m) . GA[nu] . (GS[q]+GS[k]+m)]

(* Set some Options. This changed from previous versions *)

SetOptions[PaVeReduce,A0ToB0->True]
$LimitTo4 = True;

(* Define the amplitude *)

amp:=num * FeynAmpDenominator[PropagatorDenominator[q+k,m], \
PropagatorDenominator[q,m]]

(* Calculate the result *)

res:=(-I / Pi^2) OneLoop[q,amp]
ans=PaVeReduce[res,PaVeAutoReduce->True] //Simplify

(***** End of Program VacPol.m *****)

```

The output from Mathematica is:

```

Out[2]= (4 C (k2 + 6 m2 B0[0, m, m] - 3 (k2 + 2 m2) B0[k, m, m])
        (k g[mu, nu] - k[mu] k[nu])) / (9 k2)

```

Now remembering that,

$$C = \frac{\alpha}{4\pi} \quad (173)$$

and

$$i \Pi_{\mu\nu}(k, \varepsilon) = -i k^2 P_{\mu\nu}^T \Pi(k, \varepsilon) \quad (174)$$

we get

$$\Pi(k, \varepsilon) = \frac{\alpha}{4\pi} \left[-\frac{4}{9} - \frac{8}{3} \frac{m^2}{k^2} B_0(0, m^2, m^2) + \frac{4}{3} \left(1 + \frac{2m^2}{k^2} \right) B_0(k^2, m^2, m^2) \right] \quad (175)$$

To obtain the renormalized vacuum polarization one needs to know the value of $\Pi(0, \varepsilon)$. To do that one has to take the limit $k \rightarrow 0$ in Eq. (175). For that one uses the derivative of the B_0 function

$$B'_0(p^2, m_1^2, m_2^2) \equiv \frac{\partial}{\partial p^2} B_0(p^2, m_1^2, m_2^2) \quad (176)$$

to obtain

$$\Pi(0, \varepsilon) = \frac{\alpha}{4\pi} \left[-\frac{4}{9} + \frac{4}{3} B_0(0, m^2, m^2) + \frac{8}{3} m^2 B'_0(0, m^2, m^2) \right] \quad (177)$$

Using

$$B'_0(0, m^2, m^2) = \frac{1}{6m^2} \quad (178)$$

we finally get

$$\Pi(0, \varepsilon) = -\delta Z_3 = \frac{\alpha}{4\pi} \left[\frac{4}{3} B_0(0, m^2, m^2) \right] \quad (179)$$

and the final result for the renormalized vertex is:

$$\Pi^R(k) = \frac{\alpha}{3\pi} \left[-\frac{1}{3} + \left(1 + \frac{2m^2}{k^2} \right) (B_0(k^2, m^2, m^2) - B_0(0, m^2, m^2)) \right] \quad (180)$$

If we want to compare with our earlier analytical results we need to know that

$$B_0(0, m^2, m^2) = \Delta_\varepsilon - \ln \frac{m^2}{\mu^2} \quad (181)$$

Then Eq. (180) reproduces the result of Eq. (??). The comparison between Eq. (180) and Eq. (??) can be done numerically using the package `LoopTools`[9, 10].

10.2 Electron Self-Energy in QED

In this section we repeat the usual calculation of using the Passarino-Veltman scheme. We start with the `Mathematica` program,

```

(***** Program SelfEnergy.m *****)
(*
Version compatible with FeynCalc 9.2.0

Date: 01/06/2017
Author: Jorge C. Romao
email: jorge.romao@tecnico.ulisboa.pt

```

```

*)
(* First input FeynCalc *)
(* Uncomment below if you want to call from this program. If open a new
  mathematica notebook and load FeynCalc from there you should not load
  it again
*)
(*
  << FeynCalc '
*)

(* Tell FeynCalc to reduce the result to scalar functions *)

SetOptions[PaVeReduce, A0ToB0->False, PaVeAutoReduce->True]
$LimitTo4 = True;

(* Now write the numerator of the Feynman diagram. We define the
  constant

      C= - alpha/(4 pi)

  The minus sign comes from the photon propagator. The factor
  i/(16 pi^2) is already included in this definition.

  I also use the FCE notation available since FeynCalc 6. See manual
  for explanations.
*)

num:= C GA[mu] . (GS[p]+GS[k]+m) . GA[mu]

(* Define the amplitude *)

amp:= num \
FeynAmpDenominator[PropagatorDenominator[p+k,m], \
                    PropagatorDenominator[k]]

(* Calculate the result *)

res:=(-I / Pi^2) OneLoop[k,amp]

ans=-res;

(*
  The minus sign in ans comes from the fact that -i \Sigma = diagram
  *)

(* Calculate the functions A(p^2) and B(p^2) *)

A=Coefficient[ans,DiracSlash[p],0];
B=Coefficient[ans,DiracSlash[p],1];

(* Calculate deltm *)

```

```

delm=A + m B /. ScalarProduct [p,p]->m^2//Simplify

(* Calculate delZ2 *)

Ap2 = A /. ScalarProduct [p,p]->p2
Bp2 = B /. ScalarProduct [p,p]->p2

aux=2 m D[Ap2,p2] + Bp2 \
  + 2 m^2 D[Bp2,p2] /. D[B0 [p2,0,m^2], p2]->DB0 [p2,0,m^2]

aux2= aux /. p2->m^2

aux3= aux2 /. A0 [m^2]->m^2 (B0 [m^2,0,m^2] -1)

delZ2=Simplify [aux3]
(***** End of Program SelfEnergy.m *****)

```

The output from Mathematica is:

```

      2      2
A = C (2 m - 4 m B0 [p , 0, m ])

      2      2      2      2      2      2      2      2
B = -----
      2
      p

delm = -(C m (-1 + B0 [0, 0, m ] + 2 B0 [m , 0, m ]))

      2      2      2      2
delZ2 = C (-1 + B0 [0, 0, m ] - 4 m DB0 [m , 0, m ])

```

We therefore get⁴ (in this case $C = -\frac{\alpha}{4\pi}$)

$$A = \frac{\alpha m}{\pi} \left[-\frac{1}{2} + B_0(p^2, 0, m^2) \right] \quad (184)$$

$$B = \frac{\alpha}{4\pi} \left[1 + \frac{1}{p^2} A_0(m^2) - \left(1 + \frac{m^2}{p^2} \right) B_0(p^2, 0, m^2) \right] \quad (185)$$

$$\delta_m = \frac{3\alpha m}{4\pi} \left[-\frac{1}{3} + \frac{1}{3m^2} A_0(m^2) + \frac{2}{3} B_0(m^2, 0, m^2) \right] \quad (186)$$

⁴One should notice that the PV functions A_0 and B_0 with one or two zero arguments are not independent. Different versions of FeynCalc, or different options, can give the output in different forms. To make the connections the following relations (see Eqs. (138)-(141)) are useful,

$$B_0(0, 0, m^2) = -1 + B_0(m^2, 0, m^2), \quad B_0(0, 0, m^2) = \frac{A_0(m^2)}{m^2}, \quad (182)$$

$$B_0(0, m^2, m^2) = -2 + B_0(m^2, 0, m^2), \quad B_0(0, 0, m^2) = 1 + B_0(0, m^2, m^2) \quad (183)$$

One can check that Eq. (186) is in agreement with Eq. (??). For that one needs the following relations,

$$A_0(m^2) = m^2 (B_0(m^2, 0, m^2) - 1) \quad (187)$$

$$B_0(m^2, 0, m^2) = \Delta_\varepsilon + 2 - \ln \frac{m^2}{\mu^2} \quad (188)$$

$$\int_0^1 dx(1+x) \ln \frac{m^2 x^2}{\mu^2} = -\frac{5}{2} + \frac{3}{2} \ln \frac{m^2}{\mu^2} \quad (189)$$

For δZ_2 we get

$$\delta Z_2 = \frac{\alpha}{4\pi} [2 - B_0(m^2, 0, m^2) + 4m^2 B'_0(m^2, \lambda^2, m^2)] \quad (190)$$

This expression can be shown to be equal to Eq. (??) although this is not trivial. The reason is that B'_0 is IR divergent, hence the parameter λ that controls the divergence.

10.3 QED Vertex

In this section we repeat the usual calculation for the QED vertex using the Passarino-Veltman scheme. The Mathematica program should by now be easy to understand. We just list it here,

```
(***** Program QEDVertex.m *****)
(*
Version compatible with FeynCalc 9.2.0

Date: 01/06/2017
Author: Jorge C. Romao
email: jorge.romao@tecnico.ulisboa.pt
*)

(* First input FeynCalc *)
(* Uncomment below if you want to call from this program. If open a new
   mathematica notebook and load FeynCalc from there you should not load
   it again
*)
(*
<< FeynCalc '
*)

(* Tell FeynCalc to reduce the result to scalar functions *)

SetOptions[PaVeReduce, A0ToB0->True]
$LimitTo4 = True;

(* Useful Function *)

TakeDTo4 = Function[exp, expaux1 = exp /. D -> 4 - eps;
  expaux2 = Normal[Series[expaux1, {eps, 0, 1}]];
  c0 = Coefficient[expaux2, eps, 0]; c1 = Coefficient[expaux2, eps, 1];
  c1div = c1 /. PaVe[0, {z1_}, {z2_, z3_}] -> 2/eps;
```

```

expaux3 = c0 + eps c1div // Simplify;
Simplify[expaux3 /. eps -> 0]]

(* Now write the numerator of the Feynman diagram. We define the
constant

      C=  alpha/(4 pi)

The kinematics is: q = p1 -p2 and the internal momenta is k.

*)

num:=Spinor[p1,m].GA[ro].(GS[p1]-GS[k]+m).GA[mu].(GS[p2]-GS[k]+m).GA[ro].
Spinor[p2,m]

amp:=C num \
FeynAmpDenominator[PropagatorDenominator[k,lbd], \
                    PropagatorDenominator[k-p1,m], \
                    PropagatorDenominator[k-p2,m]]

(* Define the on-shell kinematics *)

onshell={ScalarProduct[p1,p1]->m^2,ScalarProduct[p2,p2]->m^2, \
         ScalarProduct[p1,p2]->m^2-q2/2}

(* Define the divergent part of the relevant PV functions*)

div={PaVe[0,{a_},{b_,c_}]->Div}

res1=(-I / Pi^2) OneLoop[k,amp]

res=res1 /. onshell

auxV1= res /.onshell
auxV2= PaVeReduce[auxV1]
auxV3= PaVeReduce[auxV2] /. div
divV=Simplify[Div*Coefficient[auxV3,Div]]

(* Check that the divergencies do not cancel *)

testdiv:=Simplify[divV]

ans1=res;
var=Select[Variables[ans1],(Head[#]==StandardMatrixElement)&]
Set @@ {var, {ME[1],ME[2],ME[3],ME[4]}}

(* Extract the different Matrix Elements

Mathematica writes the result in terms of 4 Standard Matrix
Elements. To have a simpler result we substitute these elements
by simpler expressions (ME[1],ME[2],ME[3],ME[4]).

PR=GA[6]
PL=GA[7]

```

```

{StandardMatrixElement[u[p1, m1] . PR. u[p2, m2]],
 StandardMatrixElement[u[p1, m1] . PL. u[p2, m2]],
 StandardMatrixElement[u[p1, m1] . ga[mu] . PR . u[p2, m2]],
 StandardMatrixElement[u[p1, m1] . ga[mu] . PL . u[p2, m2]]}
*)
(* We substitute PL and PR by scalar and vector Matrix Elements
ME[5] = StandardMatrixElement[u[p1, m1] . u[p2, m2]]}
ME[6] = StandardMatrixElement[u[p1, m1] . GA[mu]. u[p2, m2]]}
*)
(* We use Gordon Identity *)
ans2=PaVeReduce[PaVeReduce[ans1]]/.
{ME[1]->ME[5]-ME[2],ME[3]->ME[6]-ME[4]}//FCE//Simplify
CE5=Coefficient[ans2, ME[5]]
CE6=Coefficient[ans2, ME[6]]
CE51=Coefficient[CE5,FV[p1,mu]]
CE52=Coefficient[CE5,FV[p2,mu]]
ans3=CE51 (FV[p1,mu]+FV[p2,mu]) ME[5] + CE6 ME[6]
test1:=Simplify[CE51-CE52]
test2:=Simplify[ans2-ans3]
ans4= ans3 /. {(FV[p1,mu]+FV[p2,mu]) ME[5] -> 2 m ME[6] -2m ME[7]}
ans5=TakeDTto4[ans4]
CGamma:=Coefficient[ans5,ME[6]]
CSigmaAux:=Coefficient[ans5,ME[7]]
test3:=Simplify[ans5-CGamma ME[6] -CSigmaAux ME[7]]
F2:=CSigmaAux /. lbd->0//Simplify
delZ1aux= - CGamma /. q2->0 //Simplify
delZ1:= delZ1aux /. lbd->0//Simplify
F1:=CGamma + delZ1 /. lbd->0 //Simplify
(***** End of Program QEDVertex.m *****)

```

From this program we can obtain first the value of δZ_1 . We get

$$\text{delZ1} = C (B0[0, m, m] - 2 (B0[m, 0, m] + 2 m C0[0, m, m, m, 0]))$$

which can be written as

$$\delta Z_1 = \frac{\alpha}{4\pi} [1 - B_0(0, 0, m^2) + 2B_0(0, m^2, m^2) - 2B_0(m^2, 0, m^2) - 4m^2 C_0(m^2, m^2, 0, m^2, \lambda^2, m^2)] \quad (191)$$

where we have introduced a small mass for the photon in the function $C_0(m^2, m^2, 0, m^2, \lambda^2, m^2)$ because it is IR divergent when $\lambda \rightarrow 0$ (see Eq. (157)). Using the results of Eqs. (139), (140), (141) and Eq. (157) we can show the important result

$$\delta Z_1 = \delta Z_2 \quad (192)$$

where δZ_2 was defined in Eq. (190). After performing the renormalization the coefficient $F_1(k^2)$ is finite and given by

```

F1 = 
$$\frac{2 C q^2}{4 m^2 - q^2} + \frac{C (8 m^2 - q^2) B_0[0, m, m]}{4 m^2 - q^2} - \frac{2 C q^2 B_0[m, 0, m]}{4 m^2 - q^2} - \frac{C (8 m^2 - 3 q^2) B_0[q^2, m, m]}{4 m^2 - q^2} - 4 C m^2 C_0[0, m, m, m, m, 0] + 2 C (2 m^2 - q^2) C_0[m, m, q^2, m, 0, m]$$

In[5]:= F1 /. q2->0
Out[5]= 0

```

or, expanding

```

F1 = 
$$C \left( -\frac{q^2}{q^2 - 4 m^2} - \frac{q^2 B_0[0, 0, m]}{q^2 - 4 m^2} + \frac{2 q^2 B_0[0, m, m]}{q^2 - 4 m^2} - \frac{8 m^2 B_0[0, m, m]}{q^2 - 4 m^2} - \frac{3 q^2 B_0[q^2, m, m]}{q^2 - 4 m^2} + \frac{8 m^2 B_0[q^2, m, m]}{q^2 - 4 m^2} \right)$$


```


$$\begin{aligned}
& + \frac{2 q^2 B_0[m^2, 0, m^2] + 4 q^2 m^2 C_0[m^2, m^2, 0, m^2, 0, m^2]}{q^2 - 4 m^2} + \\
& \frac{16 m^4 C_0[m^2, m^2, 0, m^2, 0, m^2] + 2 q^2 m^2 C_0[m^2, m^2, q^2, m^2, 0, m^2]}{q^2 - 4 m^2} + \\
& \frac{12 q^2 m^2 C_0[m^2, m^2, q^2, m^2, 0, m^2] + 16 m^4 C_0[m^2, m^2, q^2, m^2, 0, m^2]}{q^2 - 4 m^2})
\end{aligned}$$

while the coefficient $F_2(q^2)$ does not need renormalization and it is given by,

$$F_2 = \frac{-4 C m^2 (2 + B_0[0, m^2, m^2] - 2 B_0[m^2, 0, m^2] + B_0[q^2, m^2, m^2])}{4 m^2 - q^2}$$

and for $F_2(0)$ we get

$$F_2[0] = -2 C (1 + B_0[0, m^2, m^2] - B_0[m^2, 0, m^2])$$

Using the results of the Appendix (see Eqs. (138)-(141)) we can show that,

$$F_2(0) = \frac{\alpha}{2\pi} \tag{193}$$

a well known result, first obtained by Schwinger even before the renormalization program was fully understood ($F_2(q^2)$ is finite).

11 Modern techniques in a real problem: $\mu \rightarrow e\gamma$

In the previous sections we have redone most of the QED standard textbook examples using the PV decomposition and automatic tools. Here we want to present a more complex example, the calculation of the partial width $\mu \rightarrow e\gamma$ in an arbitrary theory where the charged leptons couple to scalars and fermions, charged or neutral. This has been done in Ref.[12] for fermions and bosons of arbitrary charge Q_F and Q_B , but for simplicity I will consider here separately the cases of neutral and charged scalars.

11.1 Neutral scalar charged fermion loop

We will consider a theory with the following interactions,

$$\begin{array}{c}
\begin{array}{l}
\text{---} l^- \\
\diagup \\
\text{---} S^0 \\
\diagdown \\
\text{---} F^-
\end{array}
\quad i (A_L P_L + A_R P_R)
\quad
\begin{array}{l}
\text{---} F^- \\
\diagup \\
\text{---} S^0 \\
\diagdown \\
\text{---} l^-
\end{array}
\quad i (B_L P_L + B_R P_R)
\end{array}$$

where F^- is a fermion with mass m_F and S^0 a neutral scalar with mass m_S . In fact $B_{L,R}$ are not independent of $A_{L,R}$ but it is easier for our programming to consider them completely general. The Feynman rule for the coupling of the photon with the lepton is $-ieQ_\ell \gamma^\mu$ where e is the positron charge (for an electron $Q_\ell = -1$). ℓ_i^- can be any of the leptons but we will omit all indices in the program, the lepton being identified by its mass and from the assumed kinematics

$$\ell_2(p_2) \rightarrow \ell_1(p_1) + \gamma(k) \quad (194)$$

The diagrams contributing to the process are given in Fig. 8,

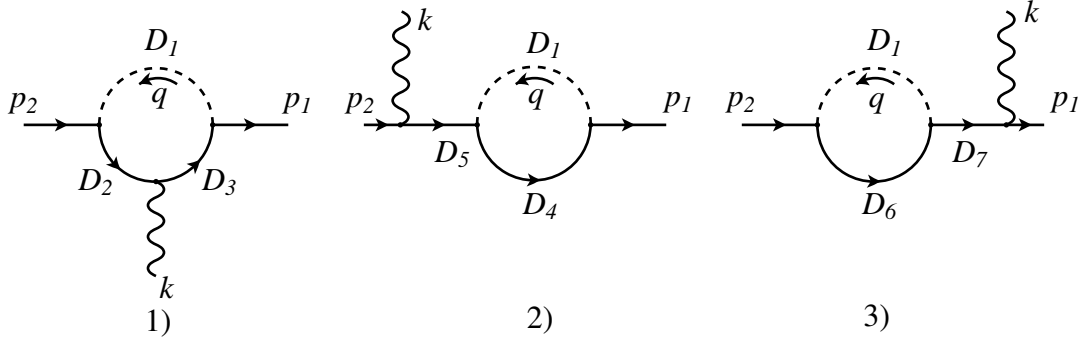


Figure 8:

where

$$D_1 = q^2 - m_S^2 \quad ; \quad D_2 = (p_2 + q)^2 - m_F^2 \quad ; \quad D_3 = (q + p_2 - k)^2 - m_F^2 \quad (195)$$

$$D_4 = D_3 \quad ; \quad D_6 = D_2 \quad ; \quad D_5 = (p_2 - k)^2 - m_2^2 = -2p_2 \cdot k \quad (196)$$

$$D_7 = (p_1 + k)^2 - m_1^2 = 2p_1 \cdot k = -D_5 \quad (197)$$

The amplitudes are

$$\begin{aligned}
iM_1 = & \frac{eQ_\ell}{D_1 D_2 D_3} \bar{u}(p_1) (A_L P_L + A_R P_R) (\not{q} + \not{p}_2 - \not{k} + m_F) \gamma^\mu (\not{q} + \not{p}_2 + m_F) \\
& (B_L P_L + B_R P_R) u(p_2) \varepsilon_\mu(k) \quad (198)
\end{aligned}$$

$$iM_2 = \frac{eQ_\ell}{D_1 D_4 D_5} \bar{u}(p_1) (A_L P_L + A_R P_R) (\not{q} + \not{p}_2 - \not{k} + m_F) (B_L P_L + B_R P_R)$$

$$(p\not{\not{k}}_2 - \not{k}_2 + m_2) \gamma^\mu u(p_2) \varepsilon_\mu(k) \quad (199)$$

$$iM_3 = \frac{e Q_\ell}{D_1 D_6 D_7} \bar{u}(p_1) \gamma^\mu (p\not{\not{k}}_1 + \not{k}_1 + m_F) (A_L P_L + A_R P_R) (\not{k} + \not{p}_2 + m_1) \\ (B_L P_L + B_R P_R) u(p_2) \varepsilon_\mu(k) \quad (200)$$

On-shell the amplitude will take the form (we have $p_1 \cdot k = p_2 \cdot k$)

$$iM = 2p_2 \cdot \varepsilon(k) \left[C_L \bar{u}(p_1) P_L u(p_2) + C_R \bar{u}(p_1) P_R u(p_2) \right] \\ + D_L u(p_1) \not{k} P_L u(p_2) + D_R u(p_1) \not{k} P_R u(p_2) \quad (201)$$

If we write the amplitude as

$$M = M_\mu \varepsilon^\mu(k) \quad (202)$$

then gauge invariance implies

$$M_\mu k^\mu = 0 \quad (203)$$

Imposing this condition on Eq. (201) we get the relations

$$D_L = -m_2 C_R - m_1 C_L \quad (204)$$

$$D_R = -m_1 C_R - m_2 C_L \quad (205)$$

Assuming these relations the amplitude can be written as

$$iM = C_L [2p_2 \cdot \varepsilon(k) \bar{u}(p_1) P_L u(p_2) - m_1 \bar{u}(p_1) \not{k} P_L u(p_2) - m_2 \bar{u}(p_1) \not{k} P_R u(p_2)] \\ + C_R [2p_2 \cdot \varepsilon(k) \bar{u}(p_1) P_R u(p_2) - m_2 \bar{u}(p_1) \not{k} P_L u(p_2) - m_1 \bar{u}(p_1) \not{k} P_R u(p_2)] \quad (206)$$

and the decay width will be

$$\Gamma = \frac{1}{16\pi m_2^3} (m_2^2 - m_1^2)^3 (|C_L|^2 + |C_R|^2) \quad (207)$$

As the coefficient of $p_2 \cdot \varepsilon(k)$ only comes from the 3-point function (amplitude M_1) this justifies the usual procedure of just calculating that coefficient and forgetting about the self-energies (amplitudes M_2 and M_3). However these amplitudes are crucial for the cancellation of divergences and for gauge invariance. Now we will show the power of the automatic `FeynCalc` [11] program and calculate both the coefficients $C_{L,R}$ and $D_{L,R}$, showing the cancellation of the divergences and that the relations, Eqs. (204) and (205) needed for gauge invariance are satisfied. We start by writing the `mathematica` program:

```

(***** Program mueg-ns.m *****)
(*)
This program calculates the COMPLETE (both the 3 point amplitude and
the two self energy type on each external line) amplitudes for
\mu -> e \gamma when the fermion line in the loop is charged and the
neutral line is a scalar. The \mu has momentum p2 and mass m2, the
electron (p1,m1) and the photon momentum k. The momentum in the loop
is q.

The assumed vertices are,

1) Electron-Scalar-Fermion:

    Spinor[p1,m1] (AL P_L + AR P_R) Spinor [pf,mf]

2) Fermion-Scalar-Muon:

    Spinor[pf,mf] (BL P_L + BR P_R) Spinor [p2,m2]
*)

dm[mu_]:=DiracMatrix[mu,Dimension->D]
dm[5]:=DiracMatrix[5]
ds[p_]:=DiracSlash[p]
mt[mu_,nu_]:=MetricTensor[mu,nu]
fv[p_,mu_]:=FourVector[p,mu]
epsilon[a_,b_,c_,d_]:=LeviCivita[a,b,c,d]
id[n_]:=IdentityMatrix[n]
sp[p_,q_]:=ScalarProduct[p,q]
li[mu_]:=LorentzIndex[mu]
L:=dm[7]
R:=dm[6]

(*)
SetOptions[{B0,B1,B00,B11},BReduce->True]
*)

gA:= AL DiracMatrix[7] + AR DiracMatrix[6]
gB:= BL DiracMatrix[7] + BR DiracMatrix[6]

num1:=Spinor[p1,m1] . gA . (ds[q]+ds[p2]-ds[k]+mf) . ds[Polarization[k]]\
      . (ds[q]+ds[p2]+mf) . gB . Spinor[p2,m2]

num2:=Spinor[p1,m1] . gA . (ds[q]+ds[p1]+mf) . gB . (ds[p1]+m2) . \
      ds[Polarization[k]] . Spinor[p2,m2]

num3:=Spinor[p1,m1] . ds[Polarization[k]] . (ds[p2]+m1) . gA . \
      (ds[q]+ds[p2]+mf) . gB . Spinor[p2,m2]

SetOptions[OneLoop,Dimension->D]

```

```

amp1:=num1 \
FeynAmpDenominator [PropagatorDenominator [q+p2-k,mf], \
                    PropagatorDenominator [q+p2,mf], \
                    PropagatorDenominator [q,ms]]
amp2:=num2 \
FeynAmpDenominator [PropagatorDenominator [q+p1,mf], \
                    PropagatorDenominator [p2-k,m2], \
                    PropagatorDenominator [q,ms]]
amp3:=num3 \
FeynAmpDenominator [PropagatorDenominator [p1+k,m1], \
                    PropagatorDenominator [q+p2,mf], \
                    PropagatorDenominator [q,ms]]

(* Define the on-shell kinematics *)

onshell={ScalarProduct [p1,p1]->m1^2,ScalarProduct [p2,p2]->m2^2, \
        ScalarProduct [k,k]->0,ScalarProduct [p1,k]->(m2^2-m1^2)/2,\
        ScalarProduct [p2,k]->(m2^2-m1^2)/2, \
        ScalarProduct [p2,Polarization [k]]->p2epk, \
        ScalarProduct [p1,Polarization [k]]->p2epk}

(* Define the divergent part of the relevant PV functions*)
div={B0[m1^2,mf^2,ms^2]->Div,B0[m2^2,mf^2,ms^2]->Div, \
    B0[0,mf^2,ms^2]->Div,B0[0,mf^2,mf^2]->Div,B0[0,ms^2,ms^2]->Div}

res1:=(-I / Pi^2) OneLoop[q,amp1]
res2:=(-I / Pi^2) OneLoop[q,amp2]
res3:=(-I / Pi^2) OneLoop[q,amp3]
res:=res1+res2+res3 /. onshell

auxT1:= res1 /. onshell
auxT2:= PaVeReduce[auxT1]
auxT3:= auxT2 /. div
divT:=Simplify[Div*Coefficient[auxT3,Div]]

auxS1:= res2 + res3 /. onshell
auxS2:= PaVeReduce[auxS1]
auxS3:= auxS2 /. div
divS:=Simplify[Div*Coefficient[auxS3,Div]]

(* Check cancellation of divergences
   testdiv should be zero because divT=-divS *)

testdiv:=Simplify[divT + divS]

(* Extract the different Matrix Elements

Mathematica writes the result in terms of 8 Standard Matrix Elements.
To have a simpler result we substitute these elements by simpler
expressions (ME[1],...ME[8]). But they are not all independent. The
final result can just be written in terms of 4 Matrix Elements.

{StandardMatrixElement [p2epk u[p1,m1] . ga[6] . u[p2,m2]],
StandardMatrixElement [p2epk u[p1,m1] . ga[7] . u[p2,m2]],

```

```

StandardMatrixElement [p2epk u[p1,m1] . gs[k] . ga[6] . u[p2,m2]],
StandardMatrixElement [p2epk u[p1,m1] . gs[k] . ga[7] . u[p2,m2]],
StandardMatrixElement [u[p1,m1] . gs[ep[k]] . ga[6] . u[p2,m2]],
StandardMatrixElement [u[p1,m1] . gs[ep[k]] . ga[7] . u[p2,m2]],
StandardMatrixElement [u[p1,m1] . gs[k] . gs[ep[k]] . ga[6] . u[p2,m2]],
StandardMatrixElement [u[p1,m1] . gs[k] . gs[ep[k]] . ga[7] . u[p2,m2]]} *)

ans1=res;
var=Select[Variables[ans1],(Head[#]==StandardMatrixElement)&]
Set @@ {var, {ME[1],ME[2],ME[3],ME[4],ME[5],ME[6],ME[7],ME[8]}}
identities={ME[3]->-m1 ME[1] + m2 ME[2], ME[4]->-m1 ME[2] + m2 ME[1],
            ME[7]->-m1 ME[5] - m2 ME[6] + 2 ME[1],
            ME[8]->-m1 ME[6] - m2 ME[5] + 2 ME[2]}

ans2 =ans1 /. identities ;
ans=Simplify[ans2];

CR=Coefficient[ans,ME[1]]/2;
CL=Coefficient[ans,ME[2]]/2;
DR=Coefficient[ans,ME[5]];
DL=Coefficient[ans,ME[6]];

(* Test to see if we did not forget any term *)

test1:=Simplify[ans-2 CR*ME[1]-2 CL*ME[2]-DR*ME[5]-DL*ME[6]]

(* Test that the divergences cancel term by term *)

auxCL=PaVeReduce[CL] /. div ;
testdivCL:=Simplify[Coefficient[auxCL,Div]]

auxCR=PaVeReduce[CR] /. div ;
testdivCR:=Simplify[Coefficient[auxCR,Div]]

auxDL=PaVeReduce[DL] /. div ;
testdivDL:=Simplify[Coefficient[auxDL,Div]]

auxDR=PaVeReduce[DR] /. div ;
testdivDR:=Simplify[Coefficient[auxDR,Div]]

(* Test the gauge invariance relations *)
testGI1:=Simplify[PaVeReduce[(m2^2-m1^2)*CR - DR*m1 + DL*m2]]

testGI2:=Simplify[PaVeReduce[(m2^2-m1^2)*CL + DR*m2 - DL*m1]]

(***** End Program muez-ns.m *****)

```

We first do the tests. The output of mathematica is

```

(***** Mathematica output *****)
In[3]:= << FeynCalc.m

FeynCalc4.1.0.3b Type ?FeynCalc for help or visit
http://www.feyncalc.org

In[4]:= << mueg-ns.m

In[5]:= test1

Out[5]= 0

In[6]:= testdiv

Out[6]= 0

In[7]:= testdivCL

Out[7]= 0

In[8]:= testdivCR

Out[8]= 0

In[9]:= testdivDL

Out[9]= 0

In[10]:= testdivDR

Out[10]= 0

In[11]:= testGI1

Out[11]= 0

In[12]:= testGI2

Out[12]= 0
(***** End of Mathematica output *****)

```

Now we obtain the results for C_L

```

(***** Mathematica output *****)
In[13]:= CL

Out[13]= (-4 AL BL mf C0[0, m2 , m1 , mf , mf , ms ] +
          4 AL BR m2 PaVe[2, {0, m1 , m2 }, {mf , mf , ms }] -
          4 AL BL mf PaVe[2, {0, m1 , m2 }, {mf , mf , ms }] -

```

```

      2      2      2      2      2
4 AR BL m1 PaVe[1, 2, {0, m1 , m2 }, {mf , mf , ms }] +
      2      2      2      2      2
4 AL BR m2 PaVe[1, 2, {0, m1 , m2 }, {mf , mf , ms }] +
      2      2      2      2      2
4 AL BR m2 PaVe[2, 2, {0, m1 , m2 }, {mf , mf , ms }]) / 4

```

and for C_R

```

In[15]:= CR
Out[15]= (-4 AR BR mf C0[0, m2 , m1 , mf , mf , ms ] +
      2      2      2      2      2
4 AR BL m2 PaVe[2, {0, m1 , m2 }, {mf , mf , ms }] -
      2      2      2      2      2
4 AR BR mf PaVe[2, {0, m1 , m2 }, {mf , mf , ms }] -
      2      2      2      2      2
4 AL BR m1 PaVe[1, 2, {0, m1 , m2 }, {mf , mf , ms }] +
      2      2      2      2      2
4 AR BL m2 PaVe[1, 2, {0, m1 , m2 }, {mf , mf , ms }] +
      2      2      2      2      2
4 AR BL m2 PaVe[2, 2, {0, m1 , m2 }, {mf , mf , ms }]) / 4
(***** End of Mathematica output *****)

```

The expressions for $D_{L,R}$ are quite complicated. They are not normally calculated because they can be related to $C_{L,R}$ by gauge invariance. However the power of this automatic program can be illustrated by asking for these functions. As they are very long we calculate them by pieces. We just calculate D_L because one can easily check that $D_R = D_L(L \leftrightarrow R)$.

```

(***** Mathematica output *****)
In[12]:= Coefficient[PaVeReduce[DL],AL BL]
      2      2      2      2      2      2
m1 mf B0[m1 , mf , ms ] m1 mf B0[m2 , mf , ms ]
Out[12]= ----- +
      2      2      2      2
m1 - m2 m1 - m2
      2      2      2      2      2
m1 mf C0[m1 , m2 , 0, mf , ms , mf ]
In[13]:= Coefficient[PaVeReduce[DL],AL BR]
      2      2      2      2

```



```

Out [13]= (mf - ms ) B0[0, mf , ms ]
-----
2 m1 m2

(m1 m2 - m2 mf + m2 ms ) B0[m1 , mf , ms ]
----- +
2 m1 (m1 - m2 )

(m1 m2 - m1 mf + m1 ms ) B0[m2 , mf , ms ]
-----
2 m2 (m1 - m2 )

In [14]:= Coefficient [PaVeReduce [DL],AR BL]

Out [14]= 1 (-2 m1 mf + 2 m1 ms ) B0[m1 , mf , ms ]
----- +
2 2 m1 (m1 - m2 )

(-2 m2 mf + 2 m2 ms ) B0[m2 , mf , ms ]
-----
2 m2 (m1 - m2 )

+ mf C0[m1 , m2 , 0, mf , ms , mf ]

In [15]:= Coefficient [PaVeReduce [DL],AR BR]

Out [15]= m2 mf B0[m1 , mf , ms ] m2 mf B0[m2 , mf , ms ]
----- - -----
2 2 m1 - m2 2 2 m1 - m2

+ m2 mf C0[m1 , m2 , 0, mf , ms , mf ]

(***** End of Mathematica output *****)

```

From these expressions one can immediately verify that the divergences cancel in $D_{L,R}$ and that they are not present in $C_{L,R}$. To finish this section we just rewrite the $C_{L,R}$ in our usual notation. We get

$$C_L = \frac{e Q_\ell}{16\pi^2} \left[A_L B_L m_F \left(-C_0(0, m_2^2, m_1^2, m_F^2, m_S^2) - C_2(0, m_1^2, m_2^2, m_F^2, m_S^2) \right) \right]$$

$$\begin{aligned}
& +A_L B_R m_2 \left(C_2(0, m_1^2, m_2^2, m_F^2, m_F^2, m_S^2) + C_{12}(0, m_1^2, m_2^2, m_F^2, m_F^2, m_S^2) \right. \\
& \quad \left. + C_{22}(0, m_1^2, m_2^2, m_F^2, m_F^2, m_S^2) \right) \\
& \quad \left. + A_R B_L m_1 C_{12}(0, m_1^2, m_2^2, m_F^2, m_F^2, m_S^2) \right] \tag{208}
\end{aligned}$$

$$C_R = C_L(L \leftrightarrow R) \tag{209}$$

These equations are in agreement with Eqs. (32-34) and Eqs. (38-39) of Ref. [12], although some work has to be done in order to verify that⁵. This has to do with the fact that the PV decomposition functions are not independent (see the Appendix for further details on this point). We can however use the power of FeynCalc to verify this. We list below a simple program to accomplish that.

```

(***** Program lavoura-ns.m *****)
(*)
This program tests the results of my program mueg-ns.m against the
results obtained by L. Lavoura (hep-ph/0302221).
*)

(* First load FeynCalc.m and mueg-ns.m *)

<< FeynCalc.m
<< mueg-ns.m

(*)
Now write Lavoura integrals in the notation of FeynCalc. Be careful
with the order of the entries.
*)

c1:=PaVe[1,{m2^2,0,m1^2},{ms^2,mf^2,mf^2}]
c2:=PaVe[2,{m2^2,0,m1^2},{ms^2,mf^2,mf^2}]

d1:=PaVe[1,1,{m2^2,0,m1^2},{ms^2,mf^2,mf^2}]
d2:=PaVe[2,2,{m2^2,0,m1^2},{ms^2,mf^2,mf^2}]

f:=PaVe[1,2,{m2^2,0,m1^2},{ms^2,mf^2,mf^2}]

(* Write Eqs. (32)-(34) of hep-ph/0302221 in our notation *)

k1:=PaVeReduce[m2*(c1+d1+f)]
k2:=PaVeReduce[m1*(c2+d2+f)]
k3:=PaVeReduce[mf*(c1+c2)]

(*)
Now test the results. For this we should use the equivalences:

\rho    -> AL BR
\lambda -> AR BL
\xi     -> AR BR
\nu     -> AL BL

```

⁵An important difference between our conventions and those of Ref.[12] is that p_1 and p_2 (and obviously m_1 and m_2) are interchanged.

```

*)
testCLALBR:=Simplify[PaVeReduce[Coefficient[CL, AL BR]-k1]]
testCLARBL:=Simplify[PaVeReduce[Coefficient[CL, AR BL]-k2]]
testCLALBL:=Simplify[PaVeReduce[Coefficient[CL, AL BL]-k3]]

testCRALBR:=Simplify[PaVeReduce[Coefficient[CR, AL BR]-k2]]
testCRARBL:=Simplify[PaVeReduce[Coefficient[CR, AR BL]-k1]]
testCRARBR:=Simplify[PaVeReduce[Coefficient[CR, AR BR]-k3]]

(***** End of Program lavoura-ns.m *****)

```

One can easily check that the output of the six tests is zero, showing the equivalence between our results. And all this is done in a few seconds.

11.2 Charged scalar neutral fermion loop

We consider now the case of the scalar being charged and the scalar neutral. The general case of both charged [12] can also be easily implemented, but for simplicity we do not consider it here. The couplings are now

$$\begin{array}{c} l^- \\ \swarrow \\ \text{---} S^- \text{---} i(A_L P_L + A_R P_R) \\ \nwarrow \\ F^0 \end{array} \qquad \begin{array}{c} F^0 \\ \swarrow \\ \text{---} S^+ \text{---} i(B_L P_L + B_R P_R) \\ \nwarrow \\ l^- \end{array}$$

and the diagrams contributing to the process are given in Fig. 9, where all the denominators

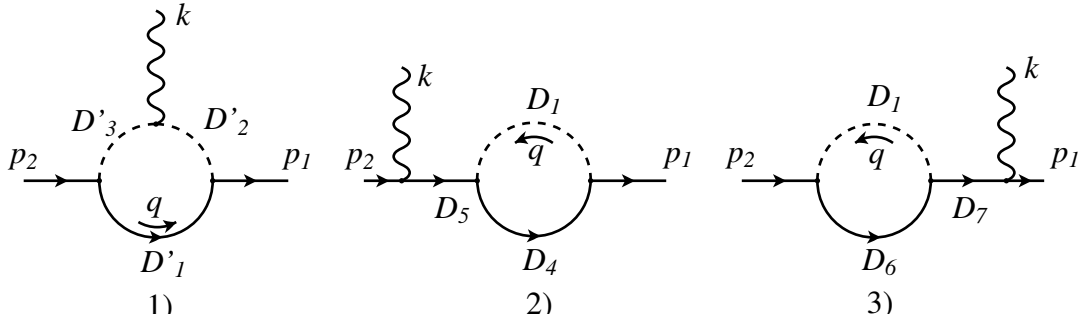


Figure 9:

are as in Eqs. (195)- (197) except that

$$D'_1 = q^2 - m_F^2 \quad ; \quad D'_2 = (q - p_1)^2 - m_S^2 \quad ; \quad D'_3 = (q - p_1 - k)^2 - m_S^2 \quad (210)$$

Also the coupling of the photon to the charged scalar is, in our notation,

$$-ie Q_\ell (-2q + p_1 + p_2)^\mu \quad (211)$$

The procedure is very similar to the neutral scalar case and we just present here the mathematica program and the final result. All the checks of finiteness and gauge invariance can be done as before.

```
(***** Program mueg-cs.m *****)
(*
This program calculates the COMPLETE (both the 3 point amplitude and
the two self energy type on each external line) amplitudes for
\mu -> e \gamma when the fermion line in the loop is neutral and the
charged line is a scalar. The \mu has momentum p2 and mass m2, the
electron (p1,m1) and the photon momentum k. The momentum in the loop
is q.

The assumed vertices are,

1) Electron-Scalar-Fermion:

Spinor[p1,m1] (AL P_L + AR P_R) Spinor [pf,mf]

2) Fermion-Scalar-Muon:

Spinor[pf,mf] (BL P_L + BR P_R) Spinor [p2,m2]
*)

dm[mu_]:=DiracMatrix[mu,Dimension->4]
dm[5]:=DiracMatrix[5]
ds[p_]:=DiracSlash[p]
mt[mu_,nu_]:=MetricTensor[mu,nu]
fv[p_,mu_]:=FourVector[p,mu]
epsilon[a_,b_,c_,d_]:=LeviCivita[a,b,c,d]
id[n_]:=IdentityMatrix[n]
sp[p_,q_]:=ScalarProduct[p,q]
li[mu_]:=LorentzIndex[mu]
L:=dm[7]
R:=dm[6]

(*
SetOptions[{B0,B1,B00,B11},BReduce->True]
*)

gA:= AL DiracMatrix[7] + AR DiracMatrix[6]
gB:= BL DiracMatrix[7] + BR DiracMatrix[6]

num1:= Spinor[p1,m1] . gA . (ds[q]+mf) . gB . Spinor[p2,m2] \
PolarizationVector[k,mu] ( - 2 fv[q,mu] + fv[p1,mu] + fv[p2,mu] )

num11:=DiracSimplify[num1];

num2:=Spinor[p1,m1] . gA . (ds[q]+ds[p1]+mf) . gB . (ds[p1]+m2) . \
ds[Polarization[k]] . Spinor[p2,m2]

num3:=Spinor[p1,m1] . ds[Polarization[k]] . (ds[p2]+m1) . gA . \
(ds[q]+ds[p2]+mf) . gB . Spinor[p2,m2]
```

```

SetOptions [OneLoop, Dimension ->D]

amp1:=num1 \
FeynAmpDenominator [PropagatorDenominator [q,mf],\
                    PropagatorDenominator [q-p1,ms],\
                    PropagatorDenominator [q-p1-k,ms]]

amp2:=num2 \
FeynAmpDenominator [PropagatorDenominator [q+p1,mf], \
                    PropagatorDenominator [p2-k,m2], \
                    PropagatorDenominator [q,ms]]

amp3:=num3 \
FeynAmpDenominator [PropagatorDenominator [p1+k,m1], \
                    PropagatorDenominator [q+p2,mf], \
                    PropagatorDenominator [q,ms]]

(* Define the on-shell kinematics *)

onshell={ScalarProduct [p1,p1]->m1^2, ScalarProduct [p2,p2]->m2^2, \
         ScalarProduct [k,k]->0, ScalarProduct [p1,k]->(m2^2-m1^2)/2, \
         ScalarProduct [p2,k]->(m2^2-m1^2)/2, \
         ScalarProduct [p2,Polarization [k]]->p2epk, \
         ScalarProduct [p1,Polarization [k]]->p2epk}

(* Define the divergent part of the relevant PV functions*)

div={B0 [m1^2,mf^2,ms^2]->Div, B0 [m2^2,mf^2,ms^2]->Div, \
     B0 [0,mf^2,ms^2]->Div, B0 [0,mf^2,mf^2]->Div, B0 [0,ms^2,ms^2]->Div}

res1:=(-I / Pi^2) OneLoop [q, amp1]
res2:=(-I / Pi^2) OneLoop [q, amp2]
res3:=(-I / Pi^2) OneLoop [q, amp3]

res:=res1+res2+res3 /. onshell

auxT1:= res1 /. onshell
auxT2:= PaVeReduce [auxT1]
auxT3:= auxT2 /. div
divT:=Simplify [Div*Coefficient [auxT3,Div]]

auxS1:= res2 + res3 /. onshell
auxS2:= PaVeReduce [auxS1]
auxS3:= auxS2 /. div
divS:=Simplify [Div*Coefficient [auxS3,Div]]

(* Check cancellation of divergences

testdiv should be zero because divT=-divS

*)

```

```

testdiv:=Simplify[divT + divS]

(* Extract the different Matrix Elements

Mathematica writes the result in terms of 6 Standard Matrix Elements.
To have a simpler result we substitute these elements by simpler
expressions (ME[1],...ME[6]). Not all are independent.

{StandardMatrixElement[p2epk u[p1, m1] . ga[6] . u[p2, m2]],
StandardMatrixElement[p2epk u[p1, m1] . ga[7] . u[p2, m2]],
StandardMatrixElement[p2epk u[p1, m1] . gs[k] . ga[6] . u[p2, m2]],
StandardMatrixElement[p2epk u[p1, m1] . gs[k] . ga[7] . u[p2, m2]],
StandardMatrixElement[u[p1, m1] . gs[ep[k]] . ga[6] . u[p2, m2]],
StandardMatrixElement[u[p1, m1] . gs[ep[k]] . ga[7] . u[p2, m2]]}
*)

ans1=res;
var=Select[Variables[ans1],(Head[#]==StandardMatrixElement)&]

Set @@ {var, {ME[1],ME[2],ME[3],ME[4],ME[5],ME[6]}}
identities={ME[3]->-m1 ME[1] + m2 ME[2],ME[4]->-m1 ME[2] + m2 ME[1]}

ans2 =ans1 /. identities ;
ans=Simplify[ans2];

CR=Coefficient[ans,ME[1]]/2;
CL=Coefficient[ans,ME[2]]/2;
DR=Coefficient[ans,ME[5]];
DL=Coefficient[ans,ME[6]];

(* Test to see if we did not forget any term *)

test1:=Simplify[ans-2*CR*ME[1]-2*CL*ME[2]-DR*ME[5]-DL*ME[6]]

(* Test that the divergences cancel term by term *)

auxCL:=PaVeReduce[CL] /. div ;
testdivCL:=Simplify[Coefficient[auxCL,Div]]

auxCR:=PaVeReduce[CR] /. div ;
testdivCR:=Simplify[Coefficient[auxCR,Div]]

auxDL:=PaVeReduce[DL] /. div ;
testdivDL:=Simplify[Coefficient[auxDL,Div]]

auxDR:=PaVeReduce[DR] /. div ;
testdivDR:=Simplify[Coefficient[auxDR,Div]]

(* Test the gauge invariance relations *)

```

```

testGI1:=PaVeReduce[(m2^2-m1^2)*CR - DR*m1 + DL*m2]

testGI2:=PaVeReduce[(m2^2-m1^2)*CL + DR*m2 - DL*m1]

(***** End Program mueg-cs.m *****)

```

Note that although these programs look large, in fact they are very simple. Most of it are comments and tests. The output of this program gives,

```

(***** Mathematica output *****)
In[3]:= CL

Out[3]= (-2 AR BL m1 C0[0, m1^2, m2^2, ms^2, ms^2, mf^2] -

2 AR BL m1 PaVe[1, {m1^2, 0, m2^2}, {mf^2, ms^2, ms^2}] -

4 AR BL m1 PaVe[1, {m1^2, m2^2, 0}, {ms^2, mf^2, ms^2}] -

2 AL BL mf PaVe[1, {m1^2, m2^2, 0}, {ms^2, mf^2, ms^2}] -

2 AL BR m2 PaVe[2, {m1^2, 0, m2^2}, {mf^2, ms^2, ms^2}] -

2 AR BL m1 PaVe[2, {m1^2, m2^2, 0}, {ms^2, mf^2, ms^2}] +

2 AL BR m2 PaVe[2, {m1^2, m2^2, 0}, {ms^2, mf^2, ms^2}] -

2 AR BL m1 PaVe[1, 1, {m1^2, m2^2, 0}, {ms^2, mf^2, ms^2}] -

2 AR BL m1 PaVe[1, 2, {m1^2, m2^2, 0}, {ms^2, mf^2, ms^2}] +

2 AL BR m2 PaVe[1, 2, {m1^2, m2^2, 0}, {ms^2, mf^2, ms^2}]) / 2

(***** End of Mathematica output *****)

```

To finish this section we just rewrite the $C_{L,R}$ in our usual notation. We get

$$\begin{aligned}
C_L = \frac{eQ_\ell}{16\pi^2} & \left[A_L B_L m_F (-C_1(m_1^2, m_2^2, 0, m_S^2, m_F^2, m_S^2)) \right. \\
& + A_L B_R m_2 \left(-C_2(m_1^2, 0, m_2^2, m_F^2, m_S^2, m_S^2) + C_2(m_1^2, m_2^2, 0, m_S^2, m_F^2, m_S^2) \right. \\
& \quad \left. \left. + C_{12}(m_1^2, m_2^2, 0, m_S^2, m_F^2, m_S^2) \right) \right. \\
& \left. + A_R B_L m_1 (-C_0(0, m_1^2, m_2^2, m_S^2, m_S^2, m_F^2) - C_1(m_1^2, 0, m_2^2, m_F^2, m_S^2, m_S^2)) \right]
\end{aligned}$$

$$\begin{aligned}
& -2C_1(m_1^2, m_2^2, 0, m_S^2, m_F^2, m_S^2) - C_2(m_1^2, m_2^2, 0, m_S^2, m_F^2, m_S^2) \\
& -C_{11}(m_1^2, m_2^2, 0, m_S^2, m_F^2, m_S^2) - C_{12}(m_1^2, m_2^2, 0, m_S^2, m_F^2, m_S^2)]
\end{aligned}$$

$$C_R = C_L(L \leftrightarrow R) \tag{212}$$

It is left as an exercise to write a `mathematica` program that proves that these equations are in agreement with Eqs. (35-37) and Eqs. (38-39) of Ref. [12].

References

- [1] J. C. Romão, *Advanced Quantum Field Theory* (IST, 2019), Available online at <http://porthos.tecnico.ulisboa.pt/Public/textos/tca.pdf>.
- [2] F. Gross, *Relativistic quantum mechanics and field theory* (, 1993).
- [3] M. D. Schwartz, *Quantum Field Theory and the Standard Model* (Cambridge University Press, 2014).
- [4] J. C. Romão, *Utilities for One Loop Calculations* (IST, 2004), Available online at <http://porthos.ist.utl.pt/OneLoop/>.
- [5] G. Passarino and M. J. G. Veltman, Nucl. Phys. **B160**, 151 (1979).
- [6] A. Denner, Fortschr. Phys. **41**, 307 (1993).
- [7] R. Mertig, M. Bohm and A. Denner, Comput. Phys. Commun. **64**, 345 (1991), Available at <https://www.feyncalc.org/>.
- [8] V. Shtabovenko, R. Mertig and F. Orellana, Comput. Phys. Commun. **207**, 432 (2016), [1601.01167].
- [9] T. Hahn and M. Perez-Victoria, Comput. Phys. Commun. **118**, 153 (1999), [hep-ph/9807565].
- [10] G. J. van Oldenborgh, Comput. Phys. Commun. **66**, 1 (1991).
- [11] R. Mertig, <http://www.feyncalc.org> .
- [12] L. Lavoura, Eur. Phys. J. **C29**, 191 (2003), [hep-ph/0302221].
- [13] J. C. Romao, <http://porthos.ist.utl.pt/CTQFT/> .