TTP98-41, BUTP-98/28, hep-ph/9812357
December 1998

Automatic Computation of Feynman Diagrams

R. Harlander[a] and M. Steinhauser[b]

a) Institut für Theoretische Teilchenphysik, Universität Karlsruhe, D-76128 Karlsruhe, Germany
b) Institut für Theoretische Physik, Universität Bern, CH-3012 Bern, Switzerland

Abstract

Quantum corrections significantly influence the quantities observed in modern particle physics. The corresponding theoretical computations are usually quite lengthy which makes their automation mandatory. This review reports on the current status of automatic calculation of Feynman diagrams in particle physics. The most important theoretical techniques are introduced and their usefulness is demonstrated with the help of simple examples. A survey over frequently used programs and packages is provided, discussing their abilities and fields of applications. Subsequently, some powerful packages which have already been applied to important physical problems are described in more detail. The review closes with the discussion of a few typical applications for the automated computation of Feynman diagrams, addressing current physical questions like properties of the Z and Higgs boson, four-loop corrections to renormalization group functions and two-loop electroweak corrections.

Keywords

Feynman diagrams, computer algebra, calculational techniques, radiative corrections, quantum chromodynamics, electroweak physics

Contents

1 Introduction

1.1 Quantum field theory in particle physics

The huge amount of data collected in the recent years mainly at the Large Electron Positron Collider (LEP) at CERN (Geneva) and the Stanford Linear Collider (SLC) as well as at the proton–anti-proton machine TEVATRON at FERMILAB (Chicago) has resulted in an impressive experimental precision for many parameters in particle physics. For example, the mass and width of the Z boson are measured with an accuracy which is of the same order of magnitude as for Fermi's constant [1,2]. Therefore it is essential to also improve the accuracy of predictions following from theoretical models. It is the purpose of this article to give an overview of some of the technical tools to perform the corresponding calculations.

At present, the most favoured theoretical model in particle physics is the so-called Standard Model. It incorporates all known quarks and leptons and describes their interactions by the gauge group SU(3)×SU(2)×U(1). Here, the SU(3) generates the strong interactions among the quarks by the exchange of eight massless gluons, while the SU(2)×U(1), usually called the electroweak sector, comprises electromagnetic and weak interactions, carried by the massless photon and the massive charged $W^{\pm}$ and neutral Z boson, respectively.

Up to now all theoretical predictions deduced from the Standard Model have found perfect confirmation in experiment. For example, the existence of the top quark was predicted after the discovery of the Υ-resonances in 1977 [3] merely out of the need for an SU(2) doublet partner for the bottom quark. At least equally impressive, however, was the accurate prediction of its mass, M_t, by comparing precision measurements of certain observables at LEP and SLC to their theoretical values. The top quark mass dependence in these observables only appears as a higher order effect through the so-called radiative corrections, typical for quantum field theories. So when the first evidence of the discovery of the top quark was announced by the CDF and D0 collaborations [4], the agreement of directly and indirectly measured values for the top mass served as a very convincing argument. Today statistics have improved considerably and the original signal has been impressively confirmed [5].

A similar situation exists with the Higgs boson, at the present time being the only particle of the Standard Model not yet discovered. Its existence, however, is by no means guaranteed as was the top quark after b discovery. Furthermore, predictions for the Higgs mass M_H suffer from much larger uncertainties because the dependence of the relevant radiative corrections on M_H is only logarithmic [6] while it is quadratic on M_t. Nevertheless, it is possible to set the limits to $M_H = 76^{+85}_{-47}$ GeV with an upper bound of $M_H < 262$ GeV at 95% confidence level [7]. The current lower limit for M_H from the direct search at LEP reads $M_H \gtrsim 89.8$ GeV [8].

From the theoretical point of view the Standard Model does not look like a "final" theory. First of all, it contains a lot of free parameters and any attempt to predict their values needs a generalization of the model. Furthermore, there is no way to find unification of the strong, weak and electromagnetic coupling constant within the Standard Model. A very promising and theoretically attractive extension is the Minimal Supersymmetric Standard Model (MSSM) which not only allows for gauge unification but even solves the problem of quadratic divergences and many others. Phenomenological calculations, however, are even more difficult here since the particle content is approximately doubled. Moreover, some important technical tools are no longer applicable.

Let us now turn to the question how to extract physical information like cross sections and decay rates from the Standard Model, leading to predictions like the values for M_t and M_H mentioned above. At the moment the vast majority of the calculations are based on perturbation theory, represented by the extremely depictive Feynman diagrams. Together with the Feynman rules, which constitute the translation rules from the graphical to a mathematical notation, they are in complete one-to-one correspondence to the individual terms of the perturbative series. As long as one is interested only in perturbative results, one may formulate the theory in terms of the Feynman rules alone, i.e., by providing the list of possible vertices, describing particle interactions, and propagators. Any physical process may then be evaluated by connecting the vertices and propagators in all possible ways to give the desired initial and final states. The specific order in perturbation theory is reflected in the number of vertices of the corresponding Feynman diagram. Given a fixed initial state and increasing the number of vertices leads to the appearance either of closed loops or of additional particles in the final state. In the former case, this implies an integration over the corresponding loop momentum, while the latter case results in a more complicated phase space integral when calculating the rate. Usually, diagrams with more than one closed loop are called multi-loop, those with more than two particles ("legs") in the final state are called multi-leg diagrams. The technology, in particular the automation of the computations in the multi-loop sector is much further developed than for multi-leg diagrams. Furthermore, to a certain extent it is possible to relate multi-leg diagrams to multi-loop ones via the optical theorem. Thus our main focus in this article lies in the calculation of loop diagrams.

1.2 Motivation for automatic computation of Feynman diagrams

The first realization of the idea to pass purely algebraic or, more precisely, symbolic operations to a computer indeed was driven by particle physics, when in 1967 M. Veltman developed the program `SCHOONSCHIP` [9], mainly to control the evaluation of fermion traces, i.e. traces of gamma matrices. Today, `SCHOONSCHIP` to a large extent is superceded by other systems which have, however, taken over many of its basic ideas. Some important examples from the early days that also have their roots in particle physics are `MACSYMA` [10], `REDUCE` [11] and others. The by now highly developed algebraic system `Mathematica` [12] actually is a derivative of the program `SMP`, created in 1980 by the particle physicist S. Wolfram. While some of these programs claim to be fairly general, the most direct descendant of `SCHOONSCHIP` is `FORM` [13], a system mainly tailored to high energy physics. In Section 2.4 a few representatives of the most important algebraic systems will be introduced. The main concern of this review are not these algebraic programs, but rather their applications, i.e., packages based on these systems. It should be clear, however, that they are very important prerequisites for the automation of Feynman diagram calculations.

The higher the order of perturbation theory under consideration, the larger is the number of contributing Feynman diagrams and the more complicated is their individual evaluation. This makes it desirable and even unavoidable to develop efficient algorithms which can be implemented on a computer in a simple way. Therefore, large efforts in the field of Feynman diagram evaluation are devoted to find algorithms for the specific types of operations arising in a typical calculation. One may characterize these operations by analyzing the steps necessary for a perturbative field theoretic calculation: given a set of Feynman rules, first the contributing Feynman diagrams for the process under consideration have to be generated. This, being mainly a question of combinatorics, is clearly a well suited problem for computerization, and meanwhile very effective algorithms have been implemented with emphasis on slightly diverse purposes. The next step is the evaluation of the corresponding mathematical expressions for the diagrams which can be divided into an algebraic and an analytic part. The algebraic part consists of operations like contracting Lorentz indices and calculating fermion traces. Nowadays many algebraic systems provide functions optimized for this kind of manipulations.

The analytic part of a Feynman diagram calculation is concerned with the Feynman integrals, and usually this is the point where computers fail to be applicable without human intervention. However, Feynman integrals establish a very special class of integrals, and for huge subclasses algorithms exist that *algebraically* reduce each of them to basic sets of meanwhile tabulated expressions. Two of the most important techniques of this kind are the so-called tensor reduction- [14] and the integration-by-parts algorithm [15]. The first one reduces any Lorentz structure in the integrand to invariants and has been worked out explicitly at the one-loop level for the general case and up to two loops for propagator diagrams. Using the integration-by-parts algorithm, on the other hand, recurrence relations can be derived which express diagrams with non-trivial topologies through simpler ones at the cost of increasing their number and the degree of the denominators. The practical realization of these algorithms would be unfeasible in realistic calculations without the use of powerful computer systems.

From the considerations of the previous paragraphs one can see that as long as one is interested in the *exact* evaluation of Feynman diagrams, the number of problems that can completely be passed to computers is rather limited which is mainly due to the relatively small number of exactly solvable Feynman integrals. There are, however, many processes where some kind of approximate result may be equally helpful. This becomes clear by recalling that, on the one hand, working at fixed order in perturbation theory is an approximation anyway, and, on the other hand, experimental results are not free of uncertainties either. Therefore, it is enormously important to have approximation procedures for Feynman diagrams. For instance, the numerical evaluation of complicated Feynman integrals, which is a typical task for a computer, is a possibility to obtain results with finite accuracy. This approach is currently favoured for the automation of tree- and one-loop calculations involving lots of different mass parameters like in the Standard Model or even the MSSM. For recent developments in the numerical evaluation of Feynman diagrams beyond the one-loop level we refer to [16].

A different kind of approximation may be obtained by expanding the integrals with respect to ratios of different scale parameters. Here the so-called asymptotic expansions of Feynman diagrams are becoming more and more popular, since they allow any multi-scale diagram to be reduced to single-scale ones, at least in principle. While the evaluation of the former may be a hard job even at one-loop level, the latter ones can be computed up to three loops using the integration-by-parts algorithm mentioned before. The required efforts for a manual application of asymptotic expansions increase steeply with the number of loops. Meanwhile, however, the most important variants have been implemented and have also been applied to the calculation of three-loop contributions to several important physical observables.

Performing huge calculations with computers inevitably leads to seemingly unrelated difficulties of purely organisational nature: Computers nowadays are not yet as stable as one would like them to be. In this way they force the user to cover on possible breakdowns, keeping the loss minimal. In the case where the long run-time is mainly due to a huge number of diagrams, each of them taking only a few minutes to be evaluated, the solution to this problem is rather straightforward. Saving each diagram on disk after computation naturally guarantees that the loss will be of only a few minutes (apart from more severe problems like disk crashes or the like). For more complicated diagrams, however, it is helpful if the algebra program itself is structured in a way that intermediate results will not be lost after system crashes.

All the algorithms mentioned above allow the calculation of Feynman diagrams to be automated to more or less high degree, where one will certainly choose different combinations of them for different purposes. It is quite difficult to give an exhaustive definition of automation. For instance, often the computer system either applies only for one special problem (or a very restricted class of problems), nevertheless saving the user a great deal of effort and preventing otherwise unavoidable errors. In other cases the system is rather flexible but requires a certain amount of human intervention for each specific problem. The final goal of automation where a typical screen snapshot would look like

```
model: SM
loops: 3
initial-state: ep,em
final-state: t,tb
parameters: default
observable: sigma_tot
energy: 500 GeV

  sigma = 0.8 pb
```

is at the moment still out of range. Nevertheless, many of the recently performed calculations in high energy physics would never have been possible without a high degree of automation. Let us, at the end of this introduction, list a few of these applications that will be addressed in more detail in Section 4.

1.3 Examples for physical applications

Most of the multi-loop and multi-leg computations have been performed for the QCD sector of the Standard Model. On the one hand they are numerically most important simply because the coupling constant is relatively large. On the other hand they are simpler to evaluate due to the fact that very often only one dimensionful scale appears like the external momentum or a large internal mass.

The "classical" example which is currently known to order α_s^3 are QCD corrections to the hadronic R ratio, $R(s) = \sigma(e^+e^- \to \text{hadrons})/\sigma(e^+e^- \to \mu^+\mu^-)$. Already in 1979 $\mathcal{O}(\alpha_s^2)$ corrections have been computed [17] in the massless limit. A large part of the calculations was carried out without using computers and lots of efforts had to be spent on nowadays trivial things like fermion traces. It took more than ten years until the next order was completed [18, 19]. Even then the manipulation of the diagrams was largely done "by hand". As a consequence, important checks like gauge parameter independence could not be performed. Only recently the $\mathcal{O}(\alpha_s^3)$ result was checked by a completely independent calculation which used the powerful methods of automatic generation and computation of diagrams [19].

The full mass dependence of the $\mathcal{O}(\alpha_s^2)$ corrections to $R(s)$ has become available [20] only very recently, when high moments of the polarization function were evaluated. In this calculation the problem was not in the number of diagrams which is less than 20, but in the huge expressions occurring in intermediate steps. (For some diagrams up to four giga-bytes of disk space were required.) For such computations it is indispensible that enormously powerful algebra systems are available and that their use is completely automated to avoid any possible human error in intermediate steps.

The number of diagrams to be considered for the $\mathcal{O}(\alpha_s^3)$ corrections to $R(s)$ at $m = 0$ is of the order of one hundred and thus it is still possible to organize the calculation by hand. The situation is different for the decay of the Higgs boson into gluons, for example. Among other reasons this is a very important process because the inverse reaction of gluon fusion is the dominant production mechanism of Higgs bosons at the future Large Hadron Collider (LHC). Since the leading order QCD corrections to the loop-induced process $H \to gg$ amount to roughly 68% [21, 22], it was necessary to evaluate the next-to-leading order contribution. However, the number of three-loop diagrams to be computed in this case is of the order one thousand. This in combination with the large expressions in the intermediate calculational steps is the reason why one has to rely on a high degree of automation as will be described in Section 4.4.

An even larger number of diagrams had to be dealt with when the renormalization group functions β and γ_m, governing the running of the strong coupling constant and the quark masses, respectively, were computed at four-loop level. Roughly 50 000 diagrams for β [23] and 2 000 for γ_m [24] contributed, and it is quite clear that such calculations would never have been possible without the intensive use of extremely efficient computer algebra systems in combination with administrative software concerned with book-keeping.

So far, none of the examples above really had to rely on one of the approximation procedures like numerical integration or asymptotic expansion mentioned above. This becomes relevant for the decay of the Z boson into quarks at $\mathcal{O}(\alpha\alpha_s)$, for example. Here, the bottom quark channel is of special interest since the top quark appears as a virtual particle. With the present technology, a calculation of the full M_t-dependence is out of reach because up to four different mass scales are involved. However, asymptotic expansions provide a very promising method to get a result which is almost equivalent to the full answer. In the approach which was used to tackle the $\mathcal{O}(\alpha\alpha_s)$ terms to this process [25], the 69 contributing diagrams are split into 234 subdiagrams. The manual application of the method of asymptotic expansions was therefore impossible and the existence of program packages that apply this procedure automatically was important.

Meanwhile, experimental accuracy allows the electroweak sector of the Standard Model to be tested even at the two-loop level, and several groups have tackled the corresponding calculations. As was already outlined, the strategy here is to reduce all integrals to scalar ones. The latter depend on a lot of different mass scales, such that their evaluation must be done numerically. In Section 4.6 we will describe the computation [26] of two-loop corrections induced by the Higgs boson to Δr which enters the relation between G_F, M_Z, M_W and α.

It is not only since the increase of the center-of-mass energy at LEP above the W boson production threshold that processes involving four (or more) particles in the final state became very topical. In such reactions even the contribution from lowest order perturbation theory may pose a serious problem as quite a lot of diagrams are involved and a highly non-trivial phase space integration has to be performed. In Section 4.5 the scattering of vector bosons is discussed in the background of a potentially large coupling among the bosons at high energies [27].

Other applications that will not be discussed in more detail but shall further substantiate the success of computer algebra in Feynman diagram calculations are, for example, QCD corrections to $\Delta\rho$ [28,29] and Δr [30], moments of QCD structure functions [31–33], top mass effects in the decay of intermediate [34] and heavy [35] Higgs bosons as well as in e^+e^- collisions [36]. Involved one-loop calculations concerning radiative corrections to the gauge boson scattering were considered in [37]. The muon anomalous magnetic moment [38] and the neutron anomalous electric moment [39] are important examples of two-loop calculations in the electroweak sector. In [40] the main difficulty was the evaluation of two-loop vertex diagrams in order to obtain corrections to the decay of a heavy Higgs boson. Investigations of b decay [41, 42] and threshold production of heavy quarks [43–46] are examples for the developments of new algorithms and their implementation. A large list of results obtained with the help of programs that automatically compute tree-level processes involving many particles can be found in [47].

Many more important calculations for physical observables in the multi-loop and multi-leg sector have been performed — mainly within the last few years, but let us close the list at this point.

The outline of this review is as follows: Section 2 introduces the most important technical tools that allow the automation of Feynman diagram calculations. This includes on the one hand some of the required algorithms, and on the other hand the most frequently used algebraic programming languages to implement these algorithms. Concrete examples for such implementations will be described in Section 3. This splits into a surveying part whose main purpose is to collect a list of the most important programs, and a more specific part concerned with the actual realization in the light of a representative set of selected packages. Finally, Section 4 gives a list of physical applications that have been considered in the literature and that would not have been feasible without the help of a certain degree of automation. Some of these examples actually were the driving force for some of the afore-mentioned packages to be developed. Others are real applications in the sense that existing programs could successfully be used to tackle as yet unsolved problems even by people that were not involved in the development of the programs.

2 Theoretical tools

2.1 Multi-loop diagrams

The main field for multi-loop calculations certainly is QCD in the perturbative regime. The reason is, on the one hand, that in general the complexity of a given Feynman diagram increases rapidly with the number of dimensional parameters involved. In the Standard Model, for example, with its relatively large amount of different particles and scales, with the currently available technology it is hardly possible to compute processes for electroweak phenomena beyond two-loop level. In QCD, by contrast, the gauge bosons are massless, and the quark masses are such that for many physical reactions one may either neglect them completely or consider only one quark flavour as massive and all others as massless particles. On the other hand, in QED, the fine structure constant is very small, rendering the higher order corrections negligible in general. The coupling constant in QCD is about ten times larger, but still small enough to play the role of an expansion parameter.

2.1.1 Dimensional Regularization and Minimal Subtraction

The momentum integrations of loop diagrams are in general not convergent in four-dimensional space-time. This requires the introduction of the concept of renormalization, rendering physical quantities finite and attributing a natural interpretation to the parameters and fields of the underlying theory. In order to isolate the divergent pieces one introduces a so-called regularization scheme, the most popular one being dimensional regularization [48] at that time. The strategy is to replace the four-dimensional loop integrals appearing in Feynman diagrams by "D-dimensional" ones, obeying the basic relations of standard convergent integrals like linearity *etc.* D is considered to be a complex parameter. The limit of integer D, if it exists, is required to reproduce the value for the standard integral. All manipulations are those of convergent integrals, and only *after* integration one takes the limit $D \to 4$. The divergences are then reflected as poles in $1/(D-4)$.

The concept of renormalization consists in compensating these divergences by adding suitable divergent terms, so-called counter-terms, to the Lagrange density. Since the only requirement on the counter-terms is to cancel the divergences, there obviously is a freedom in choosing their finite parts. A particular choice is called a renormalization scheme.

A convenient renormalization scheme in combination with dimensional regularization is the so-called minimal subtraction or MS scheme [49]. It prescribes to precisely subtract the poles in

$$\varepsilon \;\equiv\; (4-D)/2\,. \tag{1}$$

Even more popular is the $\overline{\text{MS}}$ scheme (modified MS scheme) [50] which is based on the observation that the poles in $1/\varepsilon$ appear only in the combination

$$\Delta = \frac{1}{\varepsilon} - \gamma_{\text{E}} + \ln 4\pi \tag{2}$$

and that therefore it is convenient to subtract Δ instead of $1/\varepsilon$. In the following we will use the $\overline{\text{MS}}$ scheme throughout unless stated otherwise. If divergent parts will be quoted explicitly, the γ_{E} and $\ln 4\pi$ terms will be omitted.

Dimensional regularization, like any other regularization scheme, introduces an arbitrary mass parameter, usually denoted by μ. This becomes clear by considering, for example, the one-loop integral

$$\frac{1}{i}\int \frac{\mathrm{d}^D p}{(2\pi)^D}\frac{1}{\left(-(q-p)^2\right)^a\left(-p^2\right)^b} \;=\; \frac{(-1/q^2)^{a+b-D/2}}{(4\pi)^{D/2}}\frac{\Gamma(a+b-D/2)\Gamma(D/2-a)\Gamma(D/2-b)}{\Gamma(D-a-b)\Gamma(a)\Gamma(b)}\,, \tag{3}$$

where $\Gamma(x)$ is Euler's gamma function. Note that a and b are not necessarily integer. The Laurent-series of the r.h.s. with respect to ε leads to logarithms with dimensional arguments. Therefore, one multiplies any quantity by an appropriate D-dependent power of μ such that the mass dimension of the whole object is independent of D. It is clear that only logarithmic μ-dependence may arise by this procedure. In terms of the Lagrangian, the artificial mass scale manifests itself in a coupling constant with D-dependent mass dimension.

The combination of dimensional regularization and minimal subtraction has many computational consequences. One can show, for example, that within the framework of dimensional regularization massless tadpoles, i.e. integrals that do not carry any dimensional parameter except the integration momenta, may be set to zero consistently. On the other hand, minimal subtraction guarantees that any renormalization constant is a series in the coupling constant alone, without explicit dependence on any dimensional quantity like masses or momenta. Many calculations can be considerably simplified by exploiting one of these properties. Indeed, one of the most powerful tools for the computation of multi-loop diagrams, the so-called algorithm of integration-by-parts, strongly resides on the properties of dimensional regularization. It will be described in more detail below.

There are also certain drawbacks of dimensional regularization as well as the $\overline{\rm MS}$ scheme. One of them, of course, is the lack of any reference to physical intuition, as one has it for regularization schemes like introducing cut-offs or a discrete space-time lattice, and for renormalization schemes like the on-shell scheme. Besides this, dimensional regularization generally causes problems whenever explicit reference to four dimensions is made. For example, the anti-commuting definition of γ_5 leads to inconsistencies when working in D dimensions. A consistent definition of γ_5 was given in [48] and formalized in [51]. It defines γ_5 to anti-commute with $\gamma_0,\ldots,\gamma_3$ and to commute with all the remaining γ-matrices. This definition obviously breaks Lorentz covariance and requires the introduction of so-called evanescent operators which makes practical calculations quite tedious. Also in the Supersymmetric Standard Model, defined in four-dimensional space time, one runs into problems, but we will not dwell on them here since this will not concern what follows.

2.1.2 Integration-by-parts

Let us now describe one of the benefits of dimensional regularization in more detail, namely the integration-by-parts algorithm. It uses the fact that the D-dimensional integral over a total derivative is equal to zero:

$$\int {\rm d}^D p \frac{\partial}{\partial p^\mu} f(p,\ldots) \quad = \quad 0\,. \tag{4}$$

By explicitly performing the differentiations one obtains recurrence relations connecting a complicated Feynman integral to several simpler ones. The proper combination of different recurrence relations allows any Feynman integral (at least single-scale ones) to be reduced to a small set of so-called master integrals. The latter ones have to be evaluated only once and for all, either analytically or numerically.

The integration-by-parts algorithm was initially introduced for massless two-point functions up to three loops [15], where two non-trivial master integrals were to be evaluated. Further on, the technique was applied to those massive tadpole integrals contributing to the three-loop QCD corrections of the photon polarization function in the limit $q^2 \ll m^2$ [52], where q is the external momentum of the correlator and m is the mass of the heavy quark. One non-trivial master integral results in this case. The procedure was extended to apply to the three-loop QCD corrections for the ρ parameter [28, 29] where two more master integrals had to be evaluated.

The recurrence relations for all possible three-loop tadpole integrals with a single mass were derived in [53] and the (most complicated) master integrals were calculated in [54]. At four-loop level the integration-by-parts technique was applied to completely massive tadpole diagrams, only aiming at their divergent parts, however. This restricted problem leads to two four-loop master integrals [23].

The integration-by-parts technique equally well applies to on-shell integrals, the complexity at n-loop level being comparable to the tadpole case at $n+1$ loops, however. The recurrence relations at two loops were derived some time ago [55] and were applied to the fermion propagator in order to determine the relation between the on-shell and $\overline{\rm MS}$ mass in QCD [55], and the wave function renormalization constant [56]. The three-loop on-shell integrals contributing to the anomalous magnetic moment of the electron could be reduced to 18 master integrals with the help of the integration-by-parts algorithm, and thus an analytic evaluation of this quantity could be performed [57]. Very recently $\mathcal{O}(\alpha^2)$ corrections of the μ decay were calculated [58], and the integration-by-parts method was used to determine the pole part of the corresponding four-loop on-shell integrals.

To demonstrate the power of the integration-by-parts method let us consider the scalar two-loop diagram of Fig. 1. The corresponding Feynman integral shall be denoted by

$$I(n_1,\ldots,n_5) = \int \frac{{\rm d}^D p}{(2\pi)^D}\frac{{\rm d}^D k}{(2\pi)^D}\frac{1}{(p_1^2+m_1^2)^{n_1}\cdots(p_5^2+m_5^2)^{n_5}}\,, \tag{5}$$

where $p_1,\ldots,p_5$ are combinations of the loop momenta p,k and the external momentum q (we work in Euclidean space here). $n_1,\ldots,n_5$ are called the indices of the integral. Consider the subloop defined by lines 2, 3 and 5,

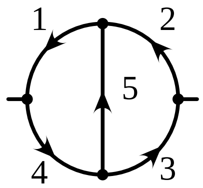


Figure 1: Two-loop master diagram. The arrows denote the direction of momentum flow.

and take its loop momentum to be $p = p_5$. If we then apply the operator $(\partial/\partial p_5)\cdot p_5$ to the *integrand* of I, we obtain a relation of the form (4), where

$$f(p_5,\ldots) = \frac{p_5^\mu}{(p_5^2+m_5^2)^{n_5}(p_2^2+m_2^2)^{n_2}(p_3^2+m_3^2)^{n_3}}\,. \tag{6}$$

Performing the differentiation and using momentum conservation at each vertex one derives the following equation:

$$\begin{aligned}\Big[-n_3\mathbf{3}^+\left(\mathbf{5}^- - \mathbf{4}^- + m_4^2 - m_5^2 - m_3^2\right) - n_2\mathbf{2}^+\left(\mathbf{5}^- - \mathbf{1}^- + m_1^2 - m_5^2 - m_2^2\right)& \\ +D - 2n_5 - n_3 - n_2 + 2n_5 m_5^2\mathbf{5}^+\Big]\, I(n_1,\ldots,n_5) \;&=\; 0\,, \end{aligned}\tag{7}$$

where the operators $\mathbf{1}^\pm, \mathbf{2}^\pm, \ldots$ are used in order to raise and lower the indices: $\mathbf{I}^\pm I(\ldots,n_i,\ldots) = I(\ldots,n_i\pm 1,\ldots)$. In Eq. (7), generally referred to as the triangle rule, it is understood that the operators to the left of $I(n_1,\ldots,n_5)$ are applied *before* integration. If the condition $m_5 = 0, m_3 = m_4$ and $m_1 = m_2$ holds, increasing one index always means to reduce another one. Therefore this recurrence relation may be used to shift the indices n_1, n_4 or n_5 to zero which leads to much simpler integrals.

The triangle rule constitutes an important building block for the general recurrence relations. The strategy is to combine several independent equations of the kind (7) in order to arrive at relations connecting one complicated integral to a set of simpler ones. For example, while the direct evaluation of even the completely massless case for the diagram in Fig. 1 is non-trivial, application of the triangle rule (7) leads to

$$I(n_1,\ldots,n_5) \;=\; \frac{1}{D-2n_5-n_2-n_3}\Big[n_2\mathbf{2}^+\left(\mathbf{5}^- - \mathbf{1}^-\right) + n_3\mathbf{3}^+\left(\mathbf{5}^- - \mathbf{4}^-\right)\Big]\, I(n_1,\ldots,n_5)\,. \tag{8}$$

Repeated application of this equation reduces one of the indices n_1, n_4 or n_5 to zero. For example, for the simplest case ($n_1 = n_2 = \ldots = n_5 = 1$) one obtains the equation pictured in Fig. 2: The non-trivial diagram on the l.h.s. is expressed as a sum of two quite simple integrals which can be solved by applying the one-loop formula of Eq. (3) twice. This example also shows a possible trap of the integration-by-parts technique. In general its application introduces artificial $1/\varepsilon$ poles which cancel only after combining all terms. They require the expansion of the individual terms up to sufficiently high powers in ε in order to obtain, for example, the finite part of the original diagram. This point must carefully be respected in computer realizations of the integration-by-parts algorithm: One must not cut the series at too low powers because then the result goes wrong; keeping too many terms, on the other hand, may intolerably slow down the performance.

In our example, the l.h.s. in Fig. 2 is finite, each term on the r.h.s., however, develops $1/\varepsilon^2$ poles. The first three orders in the expansion for $\varepsilon \to 0$ cancel, and the $\mathcal{O}(\varepsilon)$ term of the square bracket, together with the $1/\varepsilon$ in front of it, leads to the well-known result: $I(1,1,1,1,1) = 6\zeta(3)/q^2$, where q is the external momentum.

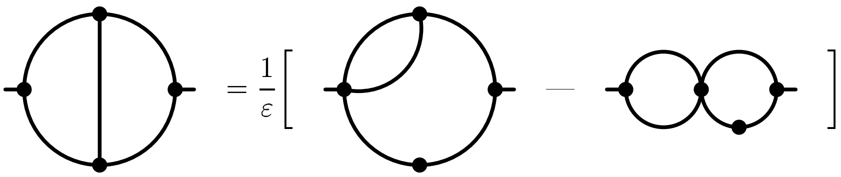


Figure 2: Symbolic equation resulting from Eq. (8) applied to the diagram $I(1,1,1,1,1)$. The dot indicates that the respective denominator appears twice.

In general, the successive application of recurrence relations generates a huge number of terms out of a single diagram. Therefore, a calculation carried out by hand becomes very tedious and the use of computer algebra is essential.

At the end of this section let us describe an alternative approach which tries to avoid the explicit use of recurrence relations and thus the large number of terms in intermediate steps of the calculation. The crucial observation is that an arbitrary integral is expressible as linear combination of the master integrals where the coefficients simply depend on the dimension D and the indices of the original integral. Therefore, an attempt to explicitly solve the system of recurrence relations in terms of integral representations was made in [59,60]. It was even possible to derive additional recurrence relations over the space-time dimension D in this approach. It was successfully applied to the class of three-loop tadpoles pictured in Fig. 3, where a significant reduction of CPU time could be achieved. Further developments in this direction look quite promising.

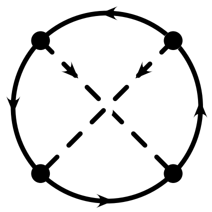


Figure 3: Three-loop topology for which the system of recurrence relations was explicitely solved. Solid lines carry a common mass M, dashed lines are massless.

2.1.3 Tensor decomposition and tensor reduction

The calculation of Feynman diagrams for realistic field theories inevitably leads to tensor integrals, i.e. Feynman integrals carrying loop momenta with free Lorentz indices in the numerator. Since allowing for tensor structure largely increases the number of possible integrals, it is important to have algorithms that reduce them to a set of basis integrals. The algorithm described in the present section is referred to as the Passarino-Veltman method [14]. At one loop-level, it reduces any tensor integral to integrals with unity in the numerator. At two-loop level this is no longer true, except for two-loop propagator-type integrals [61]. We will briefly introduce the corresponding technique at the end of this section.

Consider an arbitrary 1-loop integral carrying tensor structure in the integrand,

$$I^N_{\mu\nu\rho\cdots} = \mu^{4-D}\int\frac{\mathrm{d}^Dk}{(2\pi)^D}\frac{k_\mu k_\nu k_\rho\cdots}{D_0D_1\cdots D_{N-1}} = \left\langle\frac{k_\mu k_\nu k_\rho\cdots}{D_0D_1\cdots D_{N-1}}\right\rangle_D, \tag{9}$$

with

$$D_0 = k^2 - m_0^2, \qquad D_i = (k+p_i)^2 - m_i^2 \tag{10}$$

(the $i\epsilon$ in the denominator is suppressed here). The statement is [14] that it can be expressed in terms of the set of scalar integrals defined as

$$I^N_0 = \left\langle\frac{1}{D_0D_1\cdots D_{N-1}}\right\rangle_D. \tag{11}$$

The strategy is more or less straightforward and relies on the decomposition of the integral under consideration into covariants, built out of $g_{\mu\nu}$ and the external momenta. The general solution to this problem in terms of a recursive algorithm can be found in [62]. It reduces the rank of the tensor in the numerator by one in each step. Here we only want to give an example to get an idea of how the algorithm works. Consider the three-point second-rank tensor integral

$$I^3_{\mu\nu} = \left\langle\frac{k_\mu k_\nu}{D_0D_1D_2}\right\rangle_D, \tag{12}$$

with its tensor decomposition

$$I^3_{\mu\nu} = C_{00}g_{\mu\nu} + C_{11}p_{1\mu}p_{1\nu} + C_{22}p_{2\mu}p_{2\nu} + C_{12}p_{1\mu}p_{2\nu} + C_{21}p_{2\mu}p_{1\nu}\,. \tag{13}$$

The C_{ij} are often called *Passarino-Veltman* coefficients and are named by $A, B, C, \ldots$, corresponding to the value of $N = 1, 2, 3, \ldots$ in Eq. (9). Defining

$$R_i^\mu = \left\langle \frac{k^\mu (p_i \cdot k)}{D_0 D_1 D_2} \right\rangle_D , \quad i = 1, 2, \quad \text{and} \qquad R_{00} = \left\langle \frac{k^2}{D_0 D_1 D_2} \right\rangle_D , \tag{14}$$

we may write the tensor decomposition of those quantities as

$$R_i^\mu = r_{i1}\, p_1^\mu + r_{i2}\, p_2^\mu , \qquad R_{00} = r_{00} . \tag{15}$$

Contracting (13) by $p_{i,\nu}$ $(i = 1, 2)$ one obtains a set of four equations by separately comparing coefficients of p_1^μ and p_2^μ:

$$\begin{aligned} r_{11} &= C_{00} + C_{11}\, p_1^2 + C_{12}\, p_1 \cdot p_2 , \\ r_{12} &= C_{22}\, p_1 \cdot p_2 + C_{21}\, p_1^2 , \\ r_{22} &= C_{00} + C_{22}\, p_2^2 + C_{21}\, p_1 \cdot p_2 , \\ r_{21} &= C_{11}\, p_1 \cdot p_2 + C_{12}\, p_2^2 . \end{aligned} \tag{16}$$

Contraction of (13) with $g_{\mu\nu}$, on the other hand, yields

$$r_{00} = D\, C_{00} + C_{11}\, p_1^2 + C_{22}\, p_2^2 + C_{12}\, p_1 \cdot p_2 + C_{21}\, p_1 \cdot p_2 . \tag{17}$$

Together, these are five equations for the five unknowns C_{ij} (actually there are only *four* unknowns, since $C_{12} = C_{21}$ because of the symmetry $\mu \leftrightarrow \nu$ in $I^3_{\mu\nu}$; this redundancy may serve as a useful check in the end). Combining (17) with the sum of the first and third equation in (16), one immediately gets the solution for C_{00}:

$$C_{00} = \frac{1}{D-2} \left(r_{00} - r_{11} - r_{22} \right) . \tag{18}$$

The remaining coefficients may be determined by rewriting (16) as two sets of systems of linear equations:

$$\begin{pmatrix} r_{11} - C_{00} \\ r_{21} \end{pmatrix} = \mathbf{X} \begin{pmatrix} C_{11} \\ C_{12} \end{pmatrix} , \quad \begin{pmatrix} r_{12} \\ r_{22} - C_{00} \end{pmatrix} = \mathbf{X} \begin{pmatrix} C_{21} \\ C_{22} \end{pmatrix} , \tag{19}$$

where

$$\mathbf{X} = \begin{pmatrix} p_1^2 & p_1 \cdot p_2 \\ p_1 \cdot p_2 & p_2^2 \end{pmatrix} . \tag{20}$$

So, if $\mathbf{X}$ is invertible, Eq. (18) and the inverse of Eqs. (19) determine the C_{ij} in terms of the r_{ij}.

In turn, by rewriting

$$k \cdot p_i = \frac{1}{2} \left[D_i - D_0 - f_i \right] , \quad \text{with } f_i = p_i^2 - m_i^2 + m_0^2 , \tag{21}$$

and inserting this into the first equation of (14), one arrives at

$$\begin{aligned} r_{11}\, p_1^\mu + r_{12}\, p_2^\mu &= \frac{1}{2} \left[\left\langle \frac{k_\mu}{D_0 D_2} \right\rangle_D - \left\langle \frac{k_\mu}{D_1 D_2} \right\rangle_D - f_1 \left\langle \frac{k_\mu}{D_0 D_1 D_2} \right\rangle_D \right] , \\ r_{21}\, p_1^\mu + r_{22}\, p_2^\mu &= \frac{1}{2} \left[\left\langle \frac{k_\mu}{D_0 D_1} \right\rangle_D - \left\langle \frac{k_\mu}{D_1 D_2} \right\rangle_D - f_2 \left\langle \frac{k_\mu}{D_0 D_1 D_2} \right\rangle_D \right] , \end{aligned} \tag{22}$$

which allows to compute the r_{ij} through first-rank tensor integrals. Thus, the first run-through of recurrence is done. The second one, in turn, is only concerned with at most first-rank tensors. In that way, any tensor integral can be reduced to the basic set of scalar integrals defined in (11), provided that $\mathbf{X}$ is invertible. Several improvements to the original algorithm concerned with the problem of vanishing $\det \mathbf{X}$ and also with the numerical evaluation of the scalar integrals have been worked out [63, 64]. An overview and a complete list of references can be found in [65].

At two-loop level this strategy no longer reduces numerators of the integrals to unity. The reason is that in general one encounters “irreducible numerators”, i.e. scalar products that are not expressible in terms of the denominator via relations like (21). But in the case of propagator-type integrals, i.e. those with only one external momentum p, the full reduction may still be achieved by applying the mechanism above first to a subloop [61]. For example, consider the diagram in Fig. 4, and a corresponding first-rank tensor-integral:

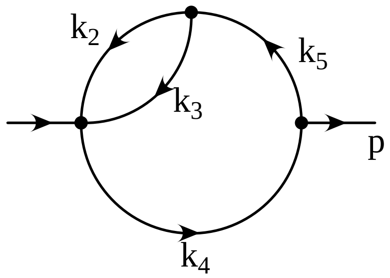


Figure 4: Two-loop propagator-type diagram. The momenta k_i are linear combinations of the loop momenta l, k and the external momentum p.

$$S_\mu = \int \frac{\mathrm{d}^D k}{(2\pi)^D} \frac{\mathrm{d}^D l}{(2\pi)^D} \frac{k_\mu}{D_k D_l D_{k+l} D_{l+p}} \,. \tag{23}$$

The notation here is a bit different from the one in (10):

$$D_{k_i} = k_i^2 - m_i^2 \,, \tag{24}$$

with

$$k_2 = k \,, \qquad k_3 = k + l \,, \qquad k_4 = l + p \,, \qquad k_5 = l \,. \tag{25}$$

Direct application of the Passarino-Veltman algorithm through the ansatz $S_\mu = p_\mu \, S(p^2)$ and contracting this equation by p^μ leads to the irreducible numerator $p \cdot k$, so that no simplification results in this way.

The idea is instead to perform the tensor reduction of the subintegral over k first, considering l as its external momentum:

$$\begin{aligned} S_\mu &= \int \mathrm{d}^D l \, \frac{1}{D_l D_{p+l}} \int \mathrm{d}^D k \, \frac{k_\mu}{D_k D_{k+l}} = \int \mathrm{d}^D l \, \frac{1}{D_l D_{p+l}} \cdot \frac{l_\mu}{l^2} \int \mathrm{d}^D k \, \frac{k \cdot l}{D_k D_{k+l}} \\ &= \int \mathrm{d}^D l \, \frac{l \cdot p}{l^2 D_l D_{p+l}} \int \mathrm{d}^D k \, \frac{k \cdot l}{D_k D_{k+l}} \cdot \frac{p_\mu}{p^2} \\ &= \frac{p_\mu}{4p^2} \int \mathrm{d}^D k \int \mathrm{d}^D l \, \frac{[D_{k+l} - D_k - l^2 - m_3^2 + m_2^2]\,[D_{p+l} - D_l - p^2 - m_5^2 + m_4^2]}{l^2 D_k D_{k+l} D_l D_{p+l}} \,. \end{aligned} \tag{26}$$

After canceling common factors, the numerator has no dependence on loop momenta any more which is what we were aiming for. The application to arbitrary propagator-type tensor integrals can be found in [61].

2.1.4 Tensor reduction by shifting the space-time dimension

As already noted in Section 2.1.3, the Passarino-Veltman method reduces the tensor structure in the numerator to unity only at one-loop level, the reason being "irreducible numerators" appearing at higher loop order. An alternative approach that circumvents this problem has been worked out at one-loop level in [66] and generalized to an arbitrary number of loops in [67,68]. However, at two-loop level it has only been applied to propagator type integrals up to now (see, e.g., [69]). The basic idea is to express tensor integrals in D dimensions through scalar ones with a shifted value of D. Again we do not want to present the algorithm in its full generality here, but try to shed some light on the main ideas by giving a concrete example.

Consider again the integral (12). By introducing an auxiliary vector a_μ, it may be written as

$$\left\langle \frac{k_\mu k_\nu}{D_0 D_1 D_2} \right\rangle_D = \left(\frac{1}{i} \frac{\partial}{\partial a^\mu} \right) \left(\frac{1}{i} \frac{\partial}{\partial a^\nu} \right) \left\langle \frac{1}{D_0 D_1 D_2} e^{iak} \right\rangle_D \bigg|_{a=0} \,. \tag{27}$$

Using the Schwinger-parameterization for propagators,

$$\frac{1}{k^2 - m^2 + i\epsilon} = \frac{1}{i} \int_0^\infty \mathrm{d}\alpha e^{i\alpha(k^2 - m^2 + i\epsilon)} \,, \tag{28}$$

one finds

$$\left\langle \frac{1}{D_0 D_1 D_2} e^{iak} \right\rangle_D = \int_0^\infty \mathrm{d}\vec{\alpha} \int \frac{\mathrm{d}^D k}{(2\pi)^D}\, e^{-i\left[\alpha_0 m_0^2 + \alpha_1 m_1^2 + \alpha_2 m_2^2\right]} \times$$
$$\exp\left\{ i \left[(\alpha_0+\alpha_1+\alpha_2) k^2 + 2\left(\alpha_1 p_1 + \alpha_2 p_2 + \frac{a}{2}\right)\cdot k + \alpha_1 p_1^2 + \alpha_2 p_2^2 \right] \right\}\,, \tag{29}$$

where $\mathrm{d}\vec{\alpha} = \mathrm{d}\alpha_0 \mathrm{d}\alpha_1 \mathrm{d}\alpha_2$, and with the help of

$$\int \frac{\mathrm{d}^D k}{(2\pi)^D} \exp\left[i(A\,k^2 + 2p\cdot k)\right] = i \frac{e^{-i\frac{p^2}{A}}}{(4\pi i A)^{D/2}}\,, \tag{30}$$

one obtains

$$\left\langle \frac{1}{D_0 D_1 D_2} e^{iak} \right\rangle_D = \frac{i}{(4\pi i)^{D/2}} \int_0^\infty \mathrm{d}\vec{\alpha}\, \frac{e^{-i\left(\alpha_0 m_0^2 + \alpha_1 m_1^2 + \alpha_2 m_2^2\right)}}{(\alpha_0+\alpha_1+\alpha_2)^{D/2}} \exp\Big\{ -i\Big[-\alpha_1(\alpha_0+\alpha_2)p_1^2$$
$$-\alpha_2(\alpha_0+\alpha_1)p_2^2 + 2\,\alpha_1\alpha_2 p_1\cdot p_2 + \alpha_1\, p_1\cdot a + \alpha_2\, p_2 \cdot a + a^2/4\Big]/(\alpha_0+\alpha_1+\alpha_2)\Big\}\,. \tag{31}$$

Inserting (31) into (27) and explicitly performing the differentiations yields

$$\left\langle \frac{k_\mu k_\nu}{D_0 D_1 D_2} \right\rangle_D = \frac{i}{(4\pi i)^{D/2}} \int \mathrm{d}\vec{\alpha}\, \frac{e^{-i\left(\alpha_0 m_0^2 + \alpha_1 m_1^2 + \alpha_2 m_2^2\right)}}{(\alpha_0+\alpha_1+\alpha_2)^{D/2+2}} (\alpha_1 p_1 + \alpha_2 p_2)_\mu (\alpha_1 p_1 + \alpha_2 p_2)_\nu \times$$
$$\exp\Big\{ -i\Big[-\alpha_1(\alpha_0+\alpha_2)p_1^2 - \alpha_2(\alpha_0+\alpha_1)p_2^2 + 2\,\alpha_1\alpha_2\, p_1\cdot p_2\Big]/(\alpha_0+\alpha_1+\alpha_2)\Big\}\,. \tag{32}$$

The technique to derive Eq. (32) and its generalization to arbitrary multi-loop diagrams is known since long (see, e.g., [51]). However, instead of differentiating with respect to a we may equally well apply the following operator on the scalar integral $\left\langle \frac{1}{D_0 D_1 D_2}\right\rangle_D$, in that way getting rid of the auxiliary vector a:

$$T_{\mu\nu}\left(\{p_1, p_2\}, \left\{ \frac{\partial}{\partial m_1^2}, \frac{\partial}{\partial m_2^2} \right\}, \mathbf{d}^+ \right) \equiv$$
$$\equiv \left(\frac{1}{i}\frac{\partial}{\partial a^\mu}\right)\left(\frac{1}{i}\frac{\partial}{\partial a^\nu}\right) \exp\left[-i(\alpha_1 p_1\cdot a + \alpha_2 p_2 \cdot a)\rho\right] \Big|_{a=0,\,\alpha_j = i\frac{\partial}{\partial m_j^2},\,\rho = 4\pi i \mathbf{d}^+}$$
$$= -\left(p_{1\mu}\frac{\partial}{\partial m_1^2} + p_{2\mu}\frac{\partial}{\partial m_2^2}\right)\left(p_{1\nu}\frac{\partial}{\partial m_1^2} + p_{2\nu}\frac{\partial}{\partial m_2^2}\right)(4\pi i\, \mathbf{d}^+)^2\,, \tag{33}$$

where the operator $\mathbf{d}^+$ increases the space-time dimension by 2, i.e. $\mathbf{d}^+\langle\cdots\rangle_D = \langle\cdots\rangle_{D+2}$. Finally, we have

$$\left\langle \frac{k_\mu k_\nu}{D_0 D_1 D_2} \right\rangle_D = T_{\mu\nu} \left\langle \frac{1}{D_0 D_1 D_2} \right\rangle_D = (4\pi)^2 \Bigg[2\, p_{1\mu} p_{1\nu} \left\langle \frac{1}{D_0 D_1^3 D_2} \right\rangle_{D+4} +$$
$$+ 2\, p_{2\mu} p_{2\nu} \left\langle \frac{1}{D_0 D_1 D_2^3} \right\rangle_{D+4} + (p_{1\mu} p_{2\nu} + p_{1\nu} p_{2\mu}) \left\langle \frac{1}{D_0 D_1^2 D_2^2} \right\rangle_{D+4} \Bigg]\,. \tag{34}$$

The algorithm formally applies to an arbitrary number of loops and external legs. But this means only that any tensor integral can be reduced to scalar integrals with a shifted number of space-time dimension, the latter ones remaining still to be evaluated. A strategy to cope with these diagrams is to use generalized recurrence relations [68]. At the one-loop level they have been worked out for arbitrary n-point functions in [67]. At the two-loop level, however, so far they are only published for propagator-type diagrams [68]. Since we feel that these generalized recurrence relations are beyond the scope of this review, let us refer the interested reader to the literature [68, 70]. The algorithm described above was used in [69] for the computation of the two-loop QCD corrections to the fermion propagator.

2.2 Asymptotic expansion of Feynman diagrams

2.2.1 Generalities

As was outlined in Section 2.1, the integration-by-parts algorithm was successfully applied to single-scale integrals, i.e. massive tadpole, massless propagator-type, or on-shell integrals. For an arbitrary multi-scale diagram it is in general rather difficult to solve recurrence relations.

However, if the scales involved follow a certain hierarchy, a factorization is possible. For example, consider the operator product expansion for the correlator of currents $j(x) = \bar{\psi}(x)\Gamma\psi(x)$ in the limit $Q^2 \to \infty$, where Γ is some Dirac matrix and ψ a quark field with mass m:

$$i \int \mathrm{d}x e^{iqx} \mathrm{T} j(x) j(0) \stackrel{Q^2 \to \infty}{\simeq} \sum_n C_n \mathcal{O}_n \,, \tag{35}$$

where T denotes the time ordered product. It was realized in [71–74] that if one adopts the minimal subtraction regularization scheme and abandons normal ordering of the operators, then Q appears only in the coefficient functions C_n while the mass m is completely absorbed into the operators $\mathcal{O}_n$ and appears only in the matrix elements.

The attempts to find an algorithm that produces this factorization for arbitrary Feynman integrals finally resulted in the prescriptions for the asymptotic expansion of Feynman diagrams [75–79] (see also [80]). These prescriptions provide well defined recipes that are completely decoupled from any field theoretic derivation and even lack a rigorous proof, as in the cases (iii) and (iv) below. However, their success in practical applications justifies them a posteriori.

At the moment the following procedures are used in the calculations:

(i) Large-Momentum Procedure: $Q \gg q, m$

(ii) Hard-Mass Procedure: $M \gg q, m$

(iii) Threshold Expansion

(iv) Expansion with the external momenta on the mass shell.

The first two of them will be considered more closely in the next subsection. The presentation will be rather informal, explaining the procedures in a ready-to-use form. Currently, the technical apparatus for the latter two cases is much less developed. For this reason they will be touched upon only briefly here.

One may treat large-momentum and hard-mass procedure on the same footing. Thus, in what follows we only present the general formulae in the case of large external momenta — the transition to the hard-mass procedure is straightforward. The prescription for the large-momentum procedure is summarized by the following formula:

$$\Gamma(Q, m, q) \quad \stackrel{Q \to \infty}{\simeq} \quad \sum_\gamma \Gamma/\gamma(q, m) \;\star\; T_{\{q_\gamma, m_\gamma\}} \gamma(Q, m_\gamma, q_\gamma) \,. \tag{36}$$

Here, Γ is the Feynman diagram under consideration, $\{Q\}$ ($\{m, q\}$) is the collection of the large (small) parameters, and the sum goes over all subgraphs γ of Γ with masses m_γ and external momenta q_γ, subject to certain conditions to be described below. $T_{\{q,m\}}$ is an operator performing a Taylor expansion in $\{q, m\}$ *before* any integration is carried out. The notation $\Gamma/\gamma \star T_{\{q,m\}}\gamma$ indicates that the subgraph γ of Γ is replaced by its Taylor expansion which should be performed in all masses and external momenta of γ that do not belong to the set $\{Q\}$. In particular, also those external momenta of γ that appear to be integration momenta in Γ have to be considered as small. Only after the Taylor expansions have been carried out, the loop integrations are performed. In the following we will refer to the set $\{\gamma\}$ as *hard subgraphs* or simply *subgraphs*, to $\{\Gamma/\gamma\}$ as *co-subgraphs*.

The conditions for the subgraphs γ are different for the hard-mass and the large-momentum procedure[1]. For the large-momentum procedure, γ must

- contain all vertices where a large momentum enters or leaves the graph
- be one-particle irreducible after identifying these vertices.

From these requirements it is clear that the hard subgraphs become massless integrals where the scales are given by the large momenta. In the simplest case of one large momentum one ends up with propagator-type integrals. The co-subgraph, on the other hand, may still contain small external momenta and masses. However, the resulting integrals are typically much simpler than the original one.

In the case of hard-mass procedure, γ has to

[1] Actually they are very similar and it is certainly possible to merge them into one condition using a more abstract language. For our purpose, however, it is more convenient to distinguish the two procedures.

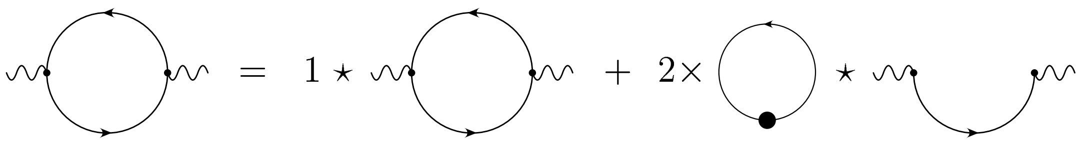


Figure 5: Large-momentum procedure for the one-loop photon polarization function.

- contain all propagators carrying a large mass
- be one-particle irreducible in its connected parts after contracting the heavy lines.

Here, the hard subgraphs reduce to tadpole integrals with the large masses setting the scales. The co-subgraphs are again simpler to evaluate than the initial diagram.

The large-momentum and hard-mass procedure provide expansions for *off-shell* external momenta which are either large or small as compared to internal masses. Recently a procedure allowing the asymptotic expansion of Feynman integrals near threshold was suggested. Graphical prescriptions similar to those for the large-momentum and hard-mass procedure have been worked out and applied to the cross section of quark production at e^+e^- colliders near threshold [46,43]. The expansion parameter in this case is given by the velocity of the produced quarks.

A method to expand on-shell Feynman diagrams was developed in [81, 82]. Two typical limits for a large external momentum on a mass shell were considered: one where the mass shell is itself large, and the other one where the mass shell is zero. The latter case, called the Sudakov limit, is a purely Minkowskian phenomenon. This distinguishes it from the cases described so far which can be formulated completely in Euclidean space, simplifying rigorous proofs. The "philosophy" of these expansions is very similar to the hard-mass and large-momentum procedure. However, apart from the criteria on the subgraphs, the way of performing the expansion of propagators is also different. For example, the expansion of lines carrying large masses and not belonging to a one-particle-irreducible component of the subgraph (so-called cut lines) looks as follows:

$$T_\kappa \frac{1}{\kappa k^2 + 2Qk}\bigg|_{\kappa=1} . \tag{37}$$

k is an integration momentum and Q a large external momentum. The graphical representation of the procedure becomes much less transparent because of that.

In [82] the two-loop master integral with two different masses and on-shell external momentum was considered: $m \ll M$, $Q^2 = M^2$. The first 19 terms of the expansion in m/M were evaluated. Similar computations have also been performed for the fermion propagator [83]. Two-loop vertex diagrams with external momenta p_1 and p_2 obeying the Sudakov limit $p_i^2 = 0$ and $(p_1 + p_2)^2 \to -\infty$ were examined in [84], and an expansion in $m^2/(p_1 + p_2)^2$ was obtained.

Let us finally emphasize that due to analyticity the obtained expansions often provide valuable information also in other regions of the parameter space. This knowledge was used to reconstruct the photon polarization function by combining the results of asymptotic expansions in different limits (see also Section 4.1).

2.2.2 Examples

Let us first consider the one-loop contribution to the photon propagator shown on the l.h.s. of the diagrammatic equation in Fig. 5. Both fermion lines are supposed to carry the same mass, m, and q is the external momentum. The application of the large-momentum procedure leads to the subdiagrams shown on the r.h.s. of this equation. The first one represents a simple Taylor expansion w.r.t. m, thus leading to massless one-loop integrals which can be solved with the help of Eq. (3). Starting from a certain order in m^2, this subdiagram develops infra-red poles which are absent in the original diagram. They are due to the massless denominators arising in the Taylor expansion. However, they cancel against twice the ultra-violet poles of the second subgraph. The factor of two arises because a symmetric subgraph should be considered as well. To make this cancellation more transparent we present the first four terms of an expansion in m^2 for the transverse part of the one-loop polarization function (cf. Section 4.1):

$$
\begin{aligned}
\Pi^{(0)}_{\rm bare}(q^2) \stackrel{q^2 \gg m^2}{=} & \frac{3}{16\pi^2}\Bigg\{\frac{4}{3\varepsilon}+\frac{20}{9}-\frac{4}{3}l_{q\mu}+8\frac{m^2}{q^2}+\left(\frac{m^2}{q^2}\right)^2\left(-\frac{8}{\varepsilon}-8+8\,l_{q\mu}\right) \\
& +\left(\frac{m^2}{q^2}\right)^3\left(-\frac{32}{3\varepsilon}-\frac{80}{3}+\frac{32}{3}l_{q\mu}\right) \\
& +2\left[\left(\frac{m^2}{q^2}\right)^2\left(\frac{4}{\varepsilon}+6+4\,l_{\mu m}\right)+\left(\frac{m^2}{q^2}\right)^3\left(\frac{16}{3\varepsilon}+\frac{88}{9}+\frac{16}{3}l_{\mu m}\right)\right]+\ldots\Bigg\} = \\
= & \frac{3}{16\pi^2}\Bigg\{\frac{4}{3\varepsilon}+\frac{20}{9}-\frac{4}{3}l_{q\mu}+8\frac{m^2}{q^2}+\left(\frac{m^2}{q^2}\right)^2\left(4+8\,l_{qm}\right) \\
& +\left(\frac{m^2}{q^2}\right)^3\left(-\frac{64}{9}+\frac{32}{3}l_{qm}\right)+\ldots\Bigg\}, \qquad (38)
\end{aligned}
$$

with $l_{q\mu}=\ln(-q^2/\mu^2)$, $l_{qm}=\ln(-q^2/m^2)$ and $l_{\mu m}=\ln(\mu^2/m^2)$. The first two lines correspond to the first subgraph of Fig. 5, and the terms in square brackets are due to the second subgraph. The fourth and fifth lines show the sum of all subgraphs which corresponds to the consistent expansion of the full one-loop diagrams. The remaining pole in $1/\varepsilon$ is the ultra-violet divergency of the full diagram and is usually removed by renormalization (see below). On the other hand, all spurious poles cancel in the sum.

Let us now analyze the diagram on the l.h.s. of Fig. 5 in the limit $q^2 \ll m^2$. In this case the hard-mass procedure applies. It leads to a trivial Taylor expansion, and one ends up with bubble integrals. The first few terms read:

$$
\Pi^{(0)}_{\rm bare}(q^2) \stackrel{q^2 \ll m^2}{=} \frac{3}{16\pi^2}\left\{\frac{4}{3\varepsilon}+\frac{4}{3}l_{\mu m}+\frac{4}{15}\frac{q^2}{m^2}+\frac{1}{35}\left(\frac{q^2}{m^2}\right)^2+\frac{4}{945}\left(\frac{q^2}{m^2}\right)^3+\ldots\right\}. \qquad (39)
$$

Note that the $1/\varepsilon$ pole is the same as in Eq. (38). In the case of the photon propagator the pole is usually removed by requiring that the polarization function vanishes for $q^2=0$. Finally the one-loop polarization function in the two limiting cases reads:

$$
\begin{aligned}
\Pi^{(0)}(q^2) \stackrel{q^2 \gg m^2}{=} & \frac{3}{16\pi^2}\Bigg\{\frac{20}{9}-\frac{4}{3}l_{qm}+8\frac{m^2}{q^2}+\left(\frac{m^2}{q^2}\right)^2\left(4+8\,l_{qm}\right) \\
& +\left(\frac{m^2}{q^2}\right)^3\left(-\frac{64}{9}+\frac{32}{3}l_{qm}\right)+\ldots\Bigg\}, \\
\Pi^{(0)}(q^2) \stackrel{q^2 \ll m^2}{=} & \frac{3}{16\pi^2}\left\{\frac{4}{15}\frac{q^2}{m^2}+\frac{1}{35}\left(\frac{q^2}{m^2}\right)^2+\frac{4}{945}\left(\frac{q^2}{m^2}\right)^3+\ldots\right\}. \qquad (40)
\end{aligned}
$$

Consider now the the case of the double-bubble diagrams pictured in Fig. 6. They provide a gauge invariant subclass of all three-loop graphs contributing to the photon polarization function. Note, however, that the particle type is irrelevant for the procedures described in the previous section; only the mass and momentum distribution is important. The outer and the inner mass are denoted by m_1 and m_2, respectively, and q is the external momentum.

Of special interest is the high energy expansion of the current correlators, i.e., $q^2 \gg m_1^2, m_2^2$, for example the case where $m_1=0$ and $m_2=m$. The imaginary part leads to $\mathcal{O}(\alpha_s^2)$ corrections to $R(s)$ in the energy range sufficiently large as compared to the mass of the produced quarks. One can think of the production of light quarks (u,d,s) accompanied by a pair of charm quarks ($m=M_c$) at energies $\sqrt{s}\gtrsim 5$ GeV, taking into account charm mass effects.

Application of the large-momentum procedure to the diagram of Fig. 6(a) results in the subdiagrams displayed in Fig. 7. We have combined topologically identical diagrams there. Similar terms arise from the graph of Fig. 6(b). Since the hard subgraphs have to be Taylor expanded in m and any "external" momentum except q, they belong to the class of massless two-point integrals, and can therefore be computed with the help of the integration-by-parts technique. The co-subgraphs, on the other hand, are massive tadpole integrals, so that also here integration-by-parts can be applied. Note that the first of the four subdiagrams in Fig. 7 represents

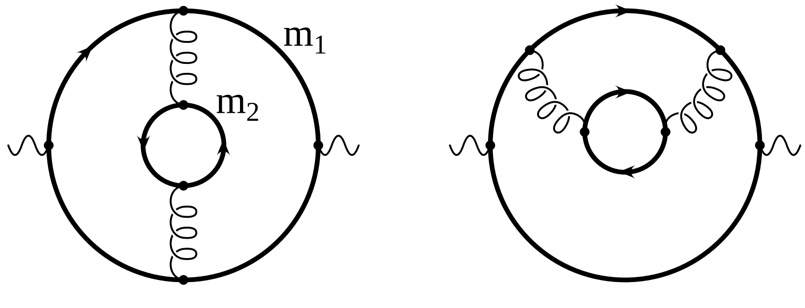


Figure 6: Fermionic double-bubble diagrams with generic masses m_1 and m_2.

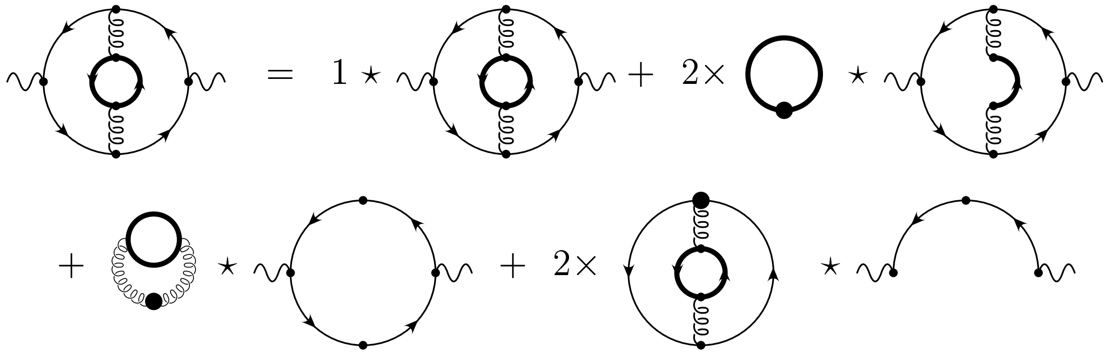


Figure 7: Large-momentum procedure for the double-bubble diagram. The inner loop (thick lines) carries mass m, the outer one is massless, and so are the gluon lines. The square of the momentum q flowing through the diagram is supposed to be much larger than m^2. It is understood that the hard subgraphs (right of "$\star$") are to be expanded in m and all external momenta except for q, and reinserted into the fat vertex dots of the co-subgraphs (left of "$\star$"). Contributions involving massless tadpoles are not displayed since they are zero in dimensional regularization.

the naive Taylor expansion of the integrand with respect to m. It is clear that starting from a certain order this term contains infra-red poles which are artificial as the original diagram is infra-red finite. The cancellation of these poles in the sum of all terms on the r.h.s. of the equation in Fig. 7 provides a non-trivial check of the calculation. The result of this expansion up to $\mathcal{O}(m^8/q^8)$ looks as follows:

$$\begin{aligned}\bar{\Pi}_{gs}(q^2) \stackrel{q^2 \gg m^2}{=} \; & \frac{3}{16\pi^2}\left(\frac{\alpha_s}{\pi}\right)^2 C_{\mathrm{F}}\,T\Bigg[-\frac{3701}{648}+\frac{38}{9}\zeta_3+l_{q\mu}\left(\frac{11}{6}-\frac{4}{3}\zeta_3\right)-\frac{1}{6}l_{q\mu}^2 \\ & +\frac{m^2}{q^2}\left(-\frac{64}{3}+16\,\zeta_3\right)+\left(\frac{m^2}{q^2}\right)^2\left(-\frac{67}{3}+16\,\zeta_3+l_{qm}\,(-\frac{26}{3}+8\,\zeta_3)-l_{qm}^2\right) \\ & +\left(\frac{m^2}{q^2}\right)^3\left(-\frac{1552}{243}+\frac{160}{27}\zeta_3-\frac{272}{243}l_{qm}+\frac{56}{81}l_{qm}^2+\frac{16}{81}l_{qm}^3\right) \\ & +\left(\frac{m^2}{q^2}\right)^4\left(\frac{1435}{324}-\frac{10}{3}\zeta_3+\frac{113}{54}l_{qm}-\frac{1}{18}l_{qm}^2-\frac{1}{9}l_{qm}^3\right)\Bigg]+\cdots,\end{aligned} \tag{41}$$

with $l_{q\mu} = \ln(-q^2/\mu^2), l_{qm} = \ln(-q^2/m^2)$ and $\zeta_3 = 1.202056903\ldots$. In Section 3.2.2 it is shown how the computation of all subdiagrams can be done using program packages designed to automate the large-momentum procedure. The logarithmic terms up to the fourth order can be found in [85], the others are new. We have adopted the $\overline{\mathrm{MS}}$ scheme, i.e., after taking into account the counter-terms for α_s induced by the (massless) one and two-loop diagrams, the local poles are subtracted. Note that because the mass m is absent in the lower order terms, it does not need to be renormalized.

As an application of the hard-mass procedure let us consider the hierarchy $m_1^2 \ll q^2 \ll m_2^2$. The imaginary part again leads to contributions for $R(s)$. This time one may think of charm quark production ($m_1 = M_c$) in

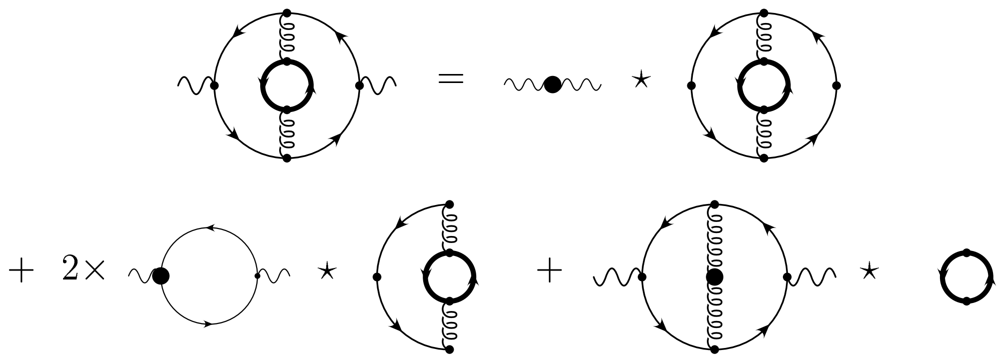


Figure 8: Hard-mass procedure for the double-bubble diagram. Now the hierarchy $q^2 \ll m^2$ is considered where m is the mass of the inner line. The hard subdiagrams (right of "$\star$") are to be expanded in all external momenta including q and reinserted into the fat vertex dots of the co-subgraphs (left of "$\star$").

the presence of a virtual bottom quark ($m_2 = M_b$). It turns out that already the first term provides a very good approximation almost up to the threshold $\sqrt{s} = 2M_b$ [86–88]. For simplicity we set $m_1 = 0$ and $m_2 = m$ in the following.

The corresponding diagrammatic representation is shown in Fig. 8. There are three subdiagrams, one of which again corresponds to the naive Taylor expansion of the integrand in the external momentum q. After Taylor expansion, the subdiagrams are reduced to tadpole integrals with mass scale m. The scale of the co-subgraphs is given by q thus leading to massless propagator-type integrals. The result for the first three terms reads [89]:

$$\begin{aligned}\bar{\Pi}_{gs}(q^2) \stackrel{q^2 \ll m^2}{=} \; & \frac{3}{16\pi^2}\left(\frac{\alpha_s}{\pi}\right)^2 C_{\mathrm{F}}\, T \Bigg[\frac{295}{648} + \frac{11}{6} l_{q\mu} - \frac{1}{6} l_{q\mu}^2 - \frac{11}{6} l_{qm} + \frac{1}{6} l_{qm}^2 \\ & - \frac{4}{3}\zeta_3 l_{q\mu} + \frac{4}{3}\zeta_3 l_{qm} + \frac{q^2}{m^2}\left(\frac{3503}{10125} - \frac{88}{675} l_{qm} + \frac{2}{135} l_{qm}^2\right) \\ & + \left(\frac{q^2}{m^2}\right)^2 \left(-\frac{2047}{514500} + \frac{1303}{529200} l_{qm} - \frac{1}{2520} l_{qm}^2\right)\Bigg] + \cdots . \end{aligned} \tag{42}$$

If m_1 was different from zero, successive application of the large-momentum procedure to the co-subgraphs would lead to a subsequent expansion in m_1^2/q^2. An example for such a repeated use of asymptotic expansions will be described in Section 4.2. Alternatively, one may evaluate the integration analytically as it has been done for the q^2/m^2 term in [90], thus leading to the full m_1 dependence of the power-suppressed result.

2.3 Helicity-amplitude technique

The standard way to obtain a cross section or a decay rate in perturbative quantum field theories is to compute the squared amplitude and integrate over the phase space for the final state particles. A typical amplitude is a sum over terms of the form

$$c \prod_i \epsilon_{\mu_i}(k_i, \chi_i) \prod_j \bar{u}(p_j, \lambda_j)\, \Gamma_j\, u'(p'_j, \lambda'_j)\,, \tag{43}$$

where ϵ are polarization vectors of vector particles with momentum k_i and polarization χ_i, u and u' are spinors of fermions (or anti-fermions) with momentum p_j, p'_j and helicity λ_j, λ'_j, respectively. Γ_j are matrix-valued objects in Dirac space, and c is a scalar function of momenta, masses, coupling constants etc. Each term of the form (43) corresponds to a certain Feynman diagram. The standard way to deal with the Dirac structure is to square the amplitude *before* evaluating the expressions in (43) any further. After summing over polarizations of final and initial states, one employs the relations

$$\sum_{\lambda=\pm 1/2} u(p,\lambda)\bar{u}(p,\lambda) \sim \not{p} \pm m \tag{44}$$

for fermions/anti-fermions and takes the trace in Dirac space which can be evaluated with the help of the anti-commutator $\{\gamma_\mu, \gamma_\nu\} = 2g_{\mu\nu}$. Specific polarization configurations can be investigated by introducing suitable projectors in Dirac space.

However, the higher the order of perturbation theory, the more diagrams contribute, and because the amplitude must be squared, the number of terms to be evaluated in the above way increases even quadratically with the number of diagrams. As long as one stays with integrated quantities (total rates), one way out is to apply the optical theorem, i.e., to take the imaginary part of the forward scattering amplitude (see, e.g., Sections 4.1 and 4.2). But as soon as one is interested in differential distributions, this method is no longer applicable. A solution here is the so-called helicity-amplitude technique. The basic idea is to rewrite expressions of the form

$$u(p,\lambda)\bar{u}'(p',\lambda') \tag{45}$$

for fixed helicities λ, λ' in terms of Dirac matrices. This allows the trace to be taken *before* squaring the amplitude. For given four-momenta one arrives at a single complex number for the full amplitude, one for each helicity configuration. If desired, the summation over polarizations is done only *after* squaring the amplitude.

This strategy goes back to [91] and was further developed in [92]. Improved algorithms have been worked out in [93, 94]. For detailed discussions of the various methods let us refer to these original works.

2.4 Algebraic Programs

Automatic computation of Feynman diagrams would not be possible without the development of powerful algebraic programming systems. While these systems themselves are based on conventional languages like `C` or `LISP`, in turn they constitute the breeding ground for all the application software which is our main concern in this review. Therefore, before discussing concrete packages that often are exclusively designed for Feynman diagram calculations, it may be helpful to briefly introduce four of the most frequently used algebraic programming languages, with emphasis on their different strategies and philosophy.

2.4.1 `Mathematica` and `MAPLE`

`Mathematica` certainly is the program with the largest bandwidth of possible applications. Apart from the algebraic operations, `Mathematica` performs analytical and numerical integrations for quite huge classes of integrals, it can do finite and infinite sums, differentiation, matrix operations, it provides graphic routines etc.

`MAPLE` is very similar to `Mathematica` and often it is only a question of taste which one to use. There certainly are fields where one of them is superior to the other, and in extreme cases one may use them in combination to find the result of a calculation.

The advantage of writing applications in `Mathematica` or `MAPLE` is that a lot of algorithms are built in and, of course, the whole environment — the interactive platform constituting an important piece in this respect — is available for the user to operate on the input and output. For example, one may perform numerical studies by inserting numbers for the parameters in the symbolic result. Meanwhile, a lot of users provide additional packages for very different purposes.

One of the disadvantages of this generality is, of course, the rather big requirements of disk and memory space. Furthermore, the improper use of some very general commands (`Integrate[]`, `Simplify[]`, etc.) may lead to inadequately long operational times or even hang-ups, a problem mainly occurring with unexperienced users.

2.4.2 `FORM`

The power of `FORM` lies in its capability of manipulating huge algebraic expressions, the only limit in size being essentially the amount of free disk space. While for `Mathematica` or `MAPLE` operating on expressions of a few mega-bytes becomes rather inconvenient, `FORM` may easily deal with several hundred mega-bytes and even giga-bytes. However, it is very restricted in its available operations. But after some experience one will be surprised how powerful the so-called `id`-statement is which replaces one quantity by an in general more complicated expression.

The operational philosophy of `FORM` is completely different to the ones of `Mathematica` and `MAPLE`, and so far `FORM` is clearly optimized for algebraic operations in high energy physics. The concept is mainly to bring any expression to a unique form by fully expanding it into monomials ("terms"). Consider, for example, the following screen-shot of a `Mathematica`-session:

```
In[1]:= f1 = (a+b)^2

                 2
Out[1]= (a + b)

In[2]:= f2 = a^2 + 2*a*b + b^2

         2             2
Out[2]= a  + 2 a b + b

In[3]:= diff = f1 - f2

          2             2          2
Out[3]= -a  - 2 a b - b  + (a + b)

In[4]:= Expand[diff]

Out[4]= 0
```

This clearly demonstrates that the internal representations of `f1` and `f2` inside `Mathematica` are not the same. `FORM`, in contrast, will instantly expand the bracket in `f1`, i.e., the input file

```
Symbol a,b;
NWrite statistics;

L f1 = (a + b)^2;
L f2 = a^2 + 2*a*b + b^2;

L diff = f1 - f2;

print;
.end
```

leads to the following output:

```
f1 =
   2*a*b + a^2 + b^2;

f2 =
   2*a*b + a^2 + b^2;

diff = 0;
```

All further operations on `f1` or `f2` will be *term-wise*, i.e., an `id`-statement

```
id a = b + c;
.sort
```

will first replace `a` by `b + c` in each of the three terms of `f1` and expand them again into monomials. The `.sort` means to collect all terms that only differ in their numerical coefficient and to bring the new expression for `f1` to a unique form again. This term-wise operation is the reason for `FORM`'s capability of dealing with such huge expressions: Only a single term is treated in each operational step, all the others can be put on disk or wherever `FORM` considers it convenient.

In addition, identification of certain terms ("pattern matching") is more transparent than in `Mathematica`. For example, the user must be aware of the fact that in order to nullify the mixed term in `f1`, the function `Expand[]` has to be applied before. To be concrete, in `Mathematica`:

```
In[2]:= f1 /. a*b -> 0

                  2
Out[2]= (a + b)

In[3]:= Expand[ f1 ] /. a*b -> 0

           2    2
Out[3]= a   + b
```

It should be clear that the discussion above is not meant to judge if FORM's or Mathematica's concept is the better one. Each one of them has its pros and cons, although the number of operations in FORM is much more restricted than in Mathematica.

2.4.3 REDUCE

REDUCE [11] is an algebra program that also was created for particle physics. It is not so general like Mathematica, but for specific problems especially in high energy physics it may be more efficient. In contrast to FORM it has an interactive mode and many more built-in functions, but again its capability of dealing with huge expressions is more limited.

3 Implementation

In Section 3.1 a brief survey of the program packages is given which are very frequently used in high energy physics. Section 3.2 describes those packages in more detail which were used to obtain the physical results presented in Section 4.

3.1 Survey of the existing program packages

This section gives a short overview of the existing program packages written to automate the treatment of Feynman diagrams. Meanwhile quite a number of such packages exist, some of them published in journals and available via anonymous ftp, others still under development and therefore only accessible for a restricted group. This review is not supposed to serve as a catalogue to select the one package most suitable for ones purposes. It shall simply provide an indication of what has been done, what is doable, and what are the most urgent topics to improve. Therefore we will not respect the question of availability. Furthermore, the description of packages and applications unavoidably will be biased, with emphasis on the topics the authors are and were personally involved in. However, since it allows us to go into details as far as ever necessary, we hope the reader will benefit from this strategy. Nevertheless, we will try to be as objective as possible and to fairly cover the class of most successful packages.

The first part will be devoted to programs concerned with the generation of diagrams. For higher order corrections this becomes more and more important, because the number of diagrams increases rapidly with the order of perturbation theory. The second part describes programs that apply to the evaluation of the corresponding amplitudes on a diagram-by-diagram basis. Some of them are optimized for the use in combination with one of the generators mentioned before. Part three deals with two packages that automatically apply the rules of the hard-mass and the large-momentum procedure. Finally, Section 3.1.4 contains a collection of programs that mainly combine some of the previously mentioned ones in order to treat full processes from the very generation up to the summation of the results of all diagrams.

The aim of this section is not to compare or judge the listed programs. This is inadmissible anyway since each of them has its specific main focuses. We will just describe their needs, abilities and applications. Each package is based on one (sometimes also several) algebraic programs, the most important of which were introduced in Section 2.4. A summary in table form will also include some additional packages that could not be described in more detail.

3.1.1 Generation of Feynman diagrams

Two programs of quite different design will be discribed: FeynArts and QGRAF.

FeynArts

FeynArts [95] is written in Mathematica and may be used interactively. Given the number of loops and external particles, FeynArts first creates all possible topologies, allowing for additional criteria to select some subset of diagrams. In the second step, fields must be attributed to the lines which is most conveniently done by choosing a specific field theoretical model. The most popular models are predefined, but the user may equally well provide his own ones. FeynArts has the nice feature of drawing the generated diagrams which works to two loops with the default setup but may be extended to higher loop order by the user. This is very helpful for debugging, for example, by figuring out if the desired subset of diagrams was selected correctly. Finally, a mathematical expression is generated for each diagram which may then be further evaluated. This is preferably done using the program packages FeynCalc, FormCalc or TwoCalc (see below) because the user's intervention remains minimal in that case. It is very convenient that FeynArts works within the Mathematica environment since a lot of powerful commands are available here, allowing the manipulation of intermediate and final results.

QGRAF

The program QGRAF [96] is written in FORTRAN 77 and is a rather efficient generator for Feynman diagrams. It takes, for example, only a few seconds to generate 10000 diagrams. The user has to provide two files: the first one, called *model file*, contains the vertices and propagators in a purely topological notation. In the second one, the *process file*, the initial and final states as well as the number of loops must be defined. Furthermore, as for FeynArts, one may provide several options allowing the selection of certain subclasses of diagrams.

Similar to the input, also the output of QGRAF is very abstract. It encodes the diagrams in a symbolic notation, thereby reproducing all necessary combinatoric factors as well as minus signs induced by fermion loops. While QGRAF suggests a distribution for the loop- and external momenta among the propagators, it is the users task to insert the Feynman rules, i.e., the proper mathematical expressions for the vertices and propagators. Nevertheless, QGRAF allows one to choose among different output formats which increases the flexibility for further operations.

3.1.2 Computation of diagrams

This section lists programs to be used for the computation of Feynman diagrams, all working on a diagram-by-diagram basis. Packages dealing with whole processes will be described in Section 3.1.4.

We will start with three packages based on FORM [13] and dealing with single-scale integrals: massive integrals with zero external momentum, massless integrals with one external momentum and two-point functions on their mass shell. Concerning their input requirements and their principle structure they are quite comparable.

Furthermore we will describe certain Mathematica packages computing one- and two-loop diagrams and a MAPLE program which provides a graphical interface for such calculations.

MINCER

The package MINCER computes one-, two- and three-loop integrals where all lines are massless and only one external momentum is different from zero. The first version of MINCER [97] was written in SCHOONSCHIP [9], but here we will only describe the much more elaborate FORM-version [98]. MINCER was the first implementation of the integration-by-parts algorithm and certainly is one of the most important programs for multi-loop calculations. It is supposed to be highly efficient, in particular because the author of FORM was involved in the translation to this system.

The user has to provide the diagrams that may all be listed in a single file, separated with the help of the fold-option of FORM. The input notation is based on the following scheme: The momenta carried by the propagators are numerated by integer numbers fixed by the topology of the diagram. The topology itself, uniquely classifying the diagram, must be specified through a keyword. The final result is given as an expansion in ε where one-loop results are expanded up to $\mathcal{O}(\varepsilon^2)$, two-loop ones up to $\mathcal{O}(\varepsilon)$ and three-loop ones up to the finite part.

MATAD

One-, two- and three-loop vacuum integrals can be evaluated with the help of `MATAD` [89], also written in `FORM`. Each propagator may either be massless or carry the common mass, M, and all external momenta have to be zero.

The input notation as well as the whole concept is very similar to the one of `MINCER`. In particular, the core of the routines is again formed by the integration-by-parts algorithm. `MATAD` provides an interface for using `MINCER` and, furthermore, it supports Taylor expansion in small masses or momenta. For example, given a massive diagram with one small external momentum, after Taylor expansion of the integrand in the small momentum one is left with a vacuum integral again. At different stages of the computation there is the opportunity to interact from outside in order to control the expansion, apply projectors, or perform other operations.

The applicability of both `MINCER` and `MATAD` seems to be quite restricted. However, as we have seen in Section 2.2, if a diagram has a certain hierarchy of mass scales which happens to be the case for quite a lot of physical applications, it can be reduced to products of single-scale integrals. Then `MINCER` and `MATAD` can be used in combination to arrive at an analytical result.

SHELL2

The program `SHELL2` [99] is also written in `FORM` and deals with one- and two-loop propagator-type on-shell integrals. The implemented topologies allow computations mainly in QED and QCD. The prototype examples are the two-loop contribution to $g-2$ of the electron and the relation between the $\overline{\mathrm{MS}}$ and on-shell mass up to order α_s^2 in QCD.

The user has to provide the diagrams in terms of a polynomial representing the numerator and abbreviated denominators. This must be supplemented by a label indicating both the number of loops and the topology. Except for the on-shell momentum no other external momentum may be present because then the numerator can be completely decomposed in terms of the denominators. A small `FORM` program ensures that an input very similar to the one of `MATAD` and `MINCER` can be applied so that one may run all three packages in parallel.

`FeynCalc`, `FormCalc` and `TwoCalc`

The `Mathematica` package `FeynCalc` [100] is based on a quite different philosophy than the ones described above. Its main applications are one-loop radiative corrections in the Standard Model and its extensions. The diagrams may either be provided by hand or one uses the output of the generator `FeynArts` (see above). `FeynCalc` performs the Dirac algebra and applies the tensor reduction algorithm of Section 2.1.3 in order to express the result in terms of scalar integrals. Special functions and the power of `Mathematica` allow to conveniently handle the intermediate and final expressions.

Since `FeynCalc` is fully based on `Mathematica`, its performance is rather slow if the underlying expressions get large. Combining the advantages of `Mathematica` and `FORM`, the package `FormCalc` [101] is a sped-up version of `FeynCalc` well suited for huge problems at the one-loop level.

The `Mathematica` package `TwoCalc` [61] is to some degree the extension of `FeynCalc` to two loops. It applies, however, only to two-point functions, since only for them may the tensor reduction algorithm be generalized to two loops, as was already noted in Section 2.1.3. Operations that are not specific for two-loop calculations, like the evaluation of Dirac traces, are passed to `FeynCalc`, so that mainly the reduction of the tensor integrals to a basic set of one- and two-loop integrals is performed.

The numerical evaluation of the scalar one-loop integrals may conveniently be performed with the help of `LoopTools` to be described below. The two-loop integrals resulting from the `TwoCalc`-routines may be evaluated using the `C`-programs `s2lse` and `master` [102].

LoopTools

LoopTools [101] is an integration of the FORTRAN program FF [103] into Mathematica. It allows a convenient numerical evaluation of the results as obtained by FeynCalc or FormCalc. In addition, it extends the ability of FF from doing only scalar integrals to the coefficients of the tensor decomposition $B_{ij}, C_{ijk}, D_{ijkl}$ as described in Section 2.1.3, so that it is not even necessary to fully reduce the tensor integrals with the help of FeynCalc in order to arrive at numerical results. Indeed, FormCalc reduces the tensor integrals only up to the point where LoopTools is applicable.

ProcessDiagram

The package ProcessDiagram [104] is written in Mathematica and deals with one- and two-loop vacuum diagrams without any restrictions concerning masses. One may apply Taylor expansions with respect to external momenta before integration. In addition, some auxiliary functions to simplify the results are available. It is, for example, possible to expand the result with respect to the masses.

XLOOPS

One- and two-loop diagrams can be processed in a convenient way with the help of the program package XLOOPS [105]. An Xwindows interface based on Tcl/Tk allows even unexperienced users to compute loop diagrams. After choosing the topology, the particle type of each line must be specified. XLOOPS then performs the Dirac algebra and expresses the result in terms of certain one- and two-loop integrals. Whereas the one-loop integrals are evaluated analytically, for the two-loop ones the result is reduced to an at most two-fold integral representation which is then evaluated numerically. The main underlying strategy in this reduction is based on the parallel space technique [106]. For the analytical part of the calculations the algebra program MAPLE V [107] is used; the numerical integrations are done with the help of VEGAS [108,109].

3.1.3 Generation of Subdiagrams

This section describes two packages, LMP and EXP, that automatically apply the hard-mass and the large-momentum procedure to a given diagram. The basic concept and the realization of these programs is quite similar. However, whereas LMP is written in PERL and was specialized for the large-momentum procedure, EXP is written in FORTRAN 90 and applies to the case of large external momenta, large internal masses, and a combination of both.

LMP and EXP

LMP [110] is written in PERL and was developed to fully exploit the power of the large-momentum procedure. It therefore extends the range of analytically calculable three-loop diagrams from single-scale ones to two-point functions carrying small masses. It found its main application in the evaluation of quark mass corrections to certain QCD processes.

Its basic concept was then carried over to the FORTRAN 90 program EXP [90] which not only evaluates the large-momentum as well as the hard-mass procedure but also any successive application of both. Although the first version of EXP is only capable of dealing with two-point functions, the next version will also incorporate the case of an arbitrary number of external momenta and masses, so that any three-loop integral will be calculable in terms of a (possibly nested) series in ratios of the involved mass scales. So far, EXP was applied to the decay rate of the Z boson into quarks [25], but one can certainly think of a huge number of further possible applications.

The usage of both LMP and EXP is very similar. Their input and output is adapted to MINCER and MATAD, so that the experienced user of these two packages will only have to provide some additional input information.

3.1.4 Complete packages

In this section five program packages are described which automate the evaluation of a given process from the generation up to the computation of the corresponding amplitudes. Both their methods and purposes are quite different.

`GEFICOM` computes Feynman diagrams up to three loops in analytical form by expanding them in terms of single-scale integrals; processes containing different mass scales are evaluated in terms of expansions. A different setup was used at NIKHEF (Amsterdam) to compute, for example, four-loop tadpole diagrams up to their simple poles in ε. Further on, `CompHEP` is a multi-leg system calculating cross sections at tree level, involving up to five particles in the final state. The automation of `GRACE` relies heavily on numerical methods and has a similar field of application like `CompHEP`. Finally, an approach to provide a rather general environment for the computation of one- and two-loop diagrams is described with the programs `TLAMM` and `DIANA`.

GEFICOM

The program package `GEFICOM` [111] combines the generator `QGRAF`, the integration packages `MATAD` and `MINCER` and the programs `EXP` and `LMP` concerned with asymptotic expansion to compute Feynman diagrams up to three loops. The translation of the `QGRAF` output to `MINCER/MATAD` notation is done by a collection of `Mathematica` routines. The links between the single packages is done by script languages like `AWK` and `PERL`.

With the short descriptions of its components in the previous section it is rather clear what the purpose of `GEFICOM` is: Given the initial and final states, `GEFICOM` generates and computes all contributing diagrams up to three loops in analytic form, provided that the involved mass scales are subject to a certain hierarchy. The result is obtained in terms of an in general multiple expansion in the ratios of the mass scales. The input therefore is essentially the same as for `QGRAF`, except that one additionally assigns masses to the different particles and defines a hierarchy among them.

“NIKHEF setup”

When the number of diagrams contributing to a single quantity increases and gets of the order 10^4 one should think carefully about the organization of the calculation. For example, storing the result of each individual diagram to a separate file may push the operational system over its limits. In the “NIKHEF setup” a database-like tool named `Minos` [112] is used to circumvent such book-keeping problems. It contains `make`-like and lots of additional features. For example, it helps to find bottlenecks of the setup by reporting on the subproblem on which most of the CPU time was spent.

For the generation of the diagrams the program `QGRAF` is used. The output of `QGRAF` is translated to the notation of the `FORM` routines `MINCER` and `BUBBLES` [113] which are concerned with the integration of the massless two-point functions and the massive tadpole integrals, respectively. Another `FORM` program, named `Color` [114], determines the colour factor. The resulting expressions are inserted into a database. After integration, the results of the single diagrams are written to another database. Finally, the diagrams are multiplied by their colour factor and summed up.

CompHEP

`CompHEP` [47] is a program package which allows the evaluation of scattering processes and decay rates at tree level. A menu-driven interface makes its use quite handy. There are several built-in models among which one finds the Standard Model both in unitary and in 't Hooft-Feynman gauge, or the Minimal Supersymmetric Standard Model. Modification of these models and definition of new ones can be done in a very convenient way. Using `LanHEP` (see Section 3.1.5) which works out the Feynman rules in `CompHEP` format, it even suffices to provide a Lagrangian density.

After a model is selected and the process is specified, the Feynman diagrams are generated and graphically displayed on the screen. The user may now select the diagrams to be treated further. Then the squared Feynman amplitudes are generated and displayed, and the corresponding analytical expressions are computed. They may be stored either in `REDUCE` or `Mathematica` format which simplifies further symbolical manipulations. For complicated processes `FORTRAN` or `C` code is generated, allowing for numerical studies. On the other hand, if the number of diagrams is not too large, numerical integrations may be performed immediately and plots showing angular distributions and cross sections can be produced.

The numerical part of `CompHEP` is based on the Monte-Carlo integration routine `VEGAS`. It is possible to introduce cuts on various kinematical variables. Further on, distributions, cross sections and particle widths can be evaluated. For the incoming particles one may define structure functions and then repeat the same integrations. Finally, `CompHEP` generates events and displays the corresponding histograms.

GRACE

A different approach for the automatic computation of Feynman diagrams is realized by the program package `GRACE` [115]. It was developed to compute cross sections and radiative corrections to them. Also here several models are available, including the Minimal Supersymmetric Standard Model [116].

`GRACE` has its own generator for Feynman diagrams, producing `FORTRAN` source code for each of them and passing it to the package `CHANEL` [117] which performs the calculation with purely numerical methods. The user is free to directly evaluate the amplitudes rather than the squared ones. This significantly reduces the size of the expressions (see Section 2.3).

In the final step the integration over the phase space for the particular final state is performed with the help of the multi-dimensional integration package `BASES` and the event generator `SPRING` which allows to generate unweighted event flow [118].

TLAMM and DIANA

There are two projects, partly under development, the purpose of which is to automate the evaluation of processes involving one- and two-loop diagrams. The package `TLAMM` [119] is written in `C` and was mainly developed to compute the two-loop corrections to the anomalous magnetic moment of the muon. It uses `QGRAF` to generate the diagrams, translates the output to `FORM` code and executes the integration routines.

The second package, called `DIANA` [120], is supposed to work on a more universal basis. Again, in a first step the output of `QGRAF` is read, the topologies are determined and internal representations for the diagrams are created. Subsequently, an interpreter executes a “special text manipulating language” (`TM`) which allows one to select the algebra language to be adopted for the calculation, to pass the expressions to `FORTRAN` in order to perform a numerical calculation or to generate, for example, PostScript files of the diagrams.

3.1.5 Miscellaneous

There are many more programs developed by several different groups. Many of them are neither published nor documented in the literature, especially if they were designed for a very special task only. This section is supposed to touch on some of them in a more or less encyclopedic form[2]. For completeness also the programs already discussed are included in the list below. Note that when we did not include information on the availability of a program this means that we could not find an apropriate `ftp` or `http` site. Interested readers should contact the authors of the corresponding programs in this case. Some programs are also available from the CPC program library (`http://www.cpc.cs.qub.ac.uk`) upon request, even if it is not indicated below.

- `BASES` [118]:
  - *Availability:* CPC program library
  - *Purpose:* Monte Carlo integration
  - *Source:* `FORTRAN`
- `BUBBLES` [113]:
  - *Purpose:* analytical computation of purely massive four-loop tadpole integrals up to $\mathcal{O}(1/\varepsilon)$
  - *Algorithms:* integration-by-parts
  - *Source:* `FORM`

[2] The links to the corresponding `www` and `ftp` sites can also be found at
`http://www-ttp.physik.uni-karlsruhe.de/Links/algprog.html`

- `CHANEL` [117]:
  - *Availability:* CPC program library
  - *Purpose:* library for the calculation of helicity amplitudes
  - *Source:* `FORTRAN`
- `Color` [114]:
  - *Availability:* `http://norma.nikhef.nl/~t68/FORMapplications/Color`
  - *Purpose:* computation of colour factors
  - *Source:* `FORM`
- `CompHEP` [47]:
  - *Availability:* `http://theory.npi.msu.su/~comphep` or `http://www.ifh.de/~pukhov`
  - *Purpose:* symbolic and numerical computation of tree level processes with up to six external legs
  - *Algorithms:* symbolic evaluation of squared diagrams, recursive representation of kinematics, Monte Carlo integration
  - *Source:* `FORTRAN`, `C`
  - *Uses:* `VEGAS`
  - *Preferably combined with:* `LanHEP`
- `DIANA` [120]:
  - *Availability:* upon request from the author
  - *Purpose:* general environment for generating and evaluating Feynman diagrams
  - *Source:* `C`
  - *Uses:* `QGRAF`
- `EXP` [90]:
  - *Purpose:* reduce two-point functions to single-scale integrals
  - *Algorithms:* large-momentum procedure, hard-mass procedure (see Section 2.2)
  - *Preferably combined with:* `MATAD`, `MINCER`
  - *Source:* `FORTRAN 90`
- `FeynArts` [95]:
  - *Availability:* `http://www-itp.physik.uni-karlsruhe.de/feynarts`
  - *Purpose:* diagram generator with main focus on one- and two-loop cases
  - *Source:* `Mathematica`
  - *Preferably combined with:* `FeynCalc`, `FormCalc`, `TwoCalc`
- `FeynCalc` [100]:
  - *Availability:* `http://www.mertig.com` (commercial);
    the latest free version is available at `http://www.mertig.com/oldfc`
  - *Purpose:* reduction of arbitrary one-loop to a set of basis integrals
  - *Algorithms:* tensor decomposition and tensor reduction by means of [14] (see Section 2.1.3)
  - *Source:* `Mathematica`
  - *Preferably combined with:* `FeynArts`, `LoopTools`
- `FF` [64, 103]:
  - *Availability:* `http://www.xs4all.nl/~gjvo/FF.html`
  - *Purpose:* numerical computation of scalar and vector one-loop integrals up to six-point functions
  - *Source:* `FORTRAN`
- `FormCalc` [101]:
  - *Availability:* `http://www-itp.physik.uni-karlsruhe.de/formcalc`
  - *Purpose:* slimmed, high speed version of `FeynCalc`
  - *Algorithms:* tensor decomposition
  - *Source:* `Mathematica`, `FORM`
  - *Preferably combined with:* `FeynArts`, `LoopTools`

- GEFICOM [111]:
  - *Purpose:* automatic generation and computation of three-loop Feynman diagrams in terms of expansions
  - *Uses:* QGRAF, MATAD, MINCER, EXP or LMP
  - *Source:* Mathematica, FORM, AWK
- GRACE [115]:
  - *Availability:* ftp://ftp.kek.jp/kek/minami/grace
  - *Purpose:* numerical computation of $2 \to 2$ scattering processes to one-loop and multi-particle scattering processes at tree level
  - *Algorithms:* helicity-amplitude method (see Section 2.3), Monte Carlo integration
  - *Uses:* CHANEL, BASES, SPRING
  - *Source:* C, FORTRAN
- HELAS [121]:
  - *Purpose:* helicity-amplitude subroutines for Feynman diagram evaluations
  - *Source:* FORTRAN
- HEPLoops [122]:
  - *Availability:* upon request from the author
  - *Purpose:* analytical computation of massless propagator-type diagrams up to three loops
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* FORM
- LanHEP [123]:
  - *Availability:* http://theory.npi.msu.su/~semenov/lanhep.html
  - *Purpose:* generate Feynman rules from Lagrangian
  - *Preferably combined with:* CompHEP
  - *Source:* C
- LMP [110]:
  - *Purpose:* factorize large external momentum in two-point functions
  - *Algorithms:* large-momentum procedure (see Section 2.2)
  - *Source:* PERL
  - *Preferably combined with:* MATAD, MINCER
- LOOPS [124]:
  - *Availability:* CPC Program Library
  - *Purpose:* computation of one- and two-loop propagator type integrals
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* REDUCE
- LoopTools [101]:
  - *Availability:* http://www-itp.physik.uni-karlsruhe.de/looptools
  - *Purpose:* implementation and extension of FF in Mathematica
  - *Uses:* FF
  - *Source:* Mathematica, FORTRAN
  - *Preferably combined with:* FeynCalc, FormCalc
- MadGraph [125]:
  - *Availability:* http://pheno.physics.wisc.edu/Software/MadGraph/
  - *Purpose:* automatic generation of Feynman diagrams; calculation of helicity amplitudes
  - *Uses:* HELAS
  - *Source:* FORTRAN
- master [102]: supplement to s2lse

- `MATAD` [89]:
  - *Purpose:* analytical computation of massive three-loop tadpole integrals
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* `FORM`
- `MINCER` (`FORM` version) [98]:
  - *Availability:* `ftp://nikhefh.nikhef.nl/pub/theory/form/libraries/form2/mincer`
  - *Purpose:* analytical computation of massless propagator-type diagrams up to three loops
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* `FORM`
- `MINCER` (`SCHOONSCHIP` version) [97]:
  - *Availability:* CPC program library
  - *Purpose:* analytical computation of massless propagator-type diagrams up to three loops
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* `SCHOONSCHIP`
- `MINOS` [112]:
  - *Purpose:* controlling facility for the calculation of processes with a huge number of diagrams
  - *Source:* `C`
- `oneloop` [126]:
  - *Availability:* `http://wwwthep.physik.uni-mainz.de/~xloops`
  - *Purpose:* algebraic and numerical calculation of one-loop diagrams
  - *Source:* `MAPLE`
- `PHACT` [127]:
  - *Purpose:* numerical computation of tree processes up to four particles in the final state
  - *Algorithms:* helicity amplitudes by means of [94]
  - *Source:* `FORTRAN`
- `ProcessDiagram` [104]:
  - *Purpose:* computation of one- and two-loop bubble diagrams allowing for several different masses
  - *Source:* `Mathematica`
- `PVEGAS` [109]: parallel version of `VEGAS`
  - *Availability:* `ftp://ftpthep.physik.uni-mainz.de/pub/pvegas/`
  - *Source:* `C`
- `RECURSOR` [128]:
  - *Purpose:* analytical computation of massive three-loop tadpole integrals
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* `REDUCE`
- `s2lse` [102]:
  - *Availability:* `ftp://ftp.physik.uni-wuerzburg.de/pub/hep/index.html`
  - *Purpose:* numerically evaluate integral representations for scalar two-loop self-energy integrals
  - *Source:* `C`
  - *Preferably combined with:* `TwoCalc`
- `SHELL2` [99]:
  - *Availability:* CPC Program Library
  - *Purpose:* computation of propagator-type on-shell integrals up to two loops
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* `FORM`
- `SIXPHACT` [129]: extension of `PHACT` to six particles in the final state

- `SLICER` [130]:
  - *Purpose:* analytical computation of massless propagator-type diagrams up to three loops
  - *Algorithms:* integration-by-parts (see Section 2.1.2)
  - *Source:* `REDUCE`
- `SPRING` [118]:
  - *Purpose:* event generation
  - *Source:* `FORTRAN`
- `TARCER` [131]:
  - *Availability:* `http://www.mertig.com/tarcer/`
  - *Purpose:* tensor reduction of two-loop propagator-type integrals and reduction to a set of basic integrals
  - *Algorithms:* tensor reduction by shifting the space-time dimension (see Section 2.1.4)
  - *Source:* `Mathematica`
- `TLAMM` [119]:
  - *Purpose:* automatic evaluation of two-loop vertex diagrams
  - *Source:* `C`
- `TRACER` [132]:
  - *Purpose:* Dirac trace calculations in `Mathematica`, optionally with 't Hooft-Veltman or naive anti-commuting γ_5
  - *Source:* `Mathematica`
- `TwoCalc` [61]:
  - *Availability:* upon request from the author
  - *Purpose:* reduction of two-loop propagator diagrams to a set of basis integrals
  - *Algorithms:* two-loop tensor reduction by means of [61]
  - *Source:* `Mathematica`
  - *Uses:* `FeynCalc`
  - *Preferably combined with:* `FeynArts`, `s2lse`, `master`
- `QGRAF` [96]:
  - *Availability:* `ftp://gtae2.ist.utl.pt/pub/qgraf/`
  - *Purpose:* efficiently generate multi-loop Feynman diagrams in symbolic notation
  - *Source:* `FORTRAN`
- `VEGAS` [108]:
  - Availability: see, e.g. [133]
  - *Purpose:* adaptive multi-dimensional integration
  - *Source:* `FORTRAN`
- `WPHACT` [134]: extension of `PHACT` to W and Higgs physics
  - *Availability:* `http://www.to.infn.it/~ballestr/wphact/`
- `XLOOPS` [105]:
  - *Availability:* `http://wino.physik.uni-mainz.de/~xloops/`
  - *Purpose:* compute one- and two-loop diagrams using X-interface
  - *Algorithms:* parallel space technique [106]
  - *Uses:* `PVEGAS`, `oneloop`
  - *Source:* `MAPLE`, `Tcl/Tk`

3.2 Selected packages

We will start with the description of the FORM programs MATAD and MINCER. Details on the automatic application of asymptotic expansions using LMP and EXP are given in Section 3.2.2. In Section 3.2.3 the package GEFICOM will be explained. The way the generator QGRAF works should also become clear then, which is why it will not be described separately. The subsequent section is devoted to the Mathematica packages FeynArts, FeynCalc, FormCalc, TwoCalc and LoopTools. In the concluding section we discuss the package CompHEP as an application of automated multi-leg computations.

3.2.1 MATAD and MINCER

Let us examine the three-loop ladder diagram (see Fig. 9) in order to see in detail how the programs MATAD and MINCER work. We will focus on the case where the external momentum, q_1, is much smaller than the mass of the fermions, m. Application of the rules for the hard-mass procedure (Section 2.2) shows that it suffices to perform a Taylor expansion of the integrand in order to arrive at an expansion in q_1^2/m^2. The resulting integrals to be evaluated are of the tadpole type, and the calculation can completely be passed to MATAD. In the opposite limit, $q_1^2 \gg m^2$, in addition to the naive Taylor expansion of the integrand in m^2/q_1^2, the large-momentum procedure generates non-trivial subgraphs. In general their evaluation requires to use MATAD and MINCER in combination.

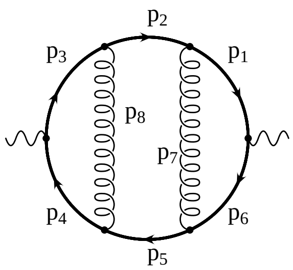


Figure 9: Ladder-type diagram contributing to the photon propagator. The solid, wavy and curly lines represent quarks, photons and gluons, respectively.

The input for the Feynman graph is written to a file using the fold construct of FORM [13]:

```
*--#[ ladder:
        ((-1)
        *Dg(nu1,nu2,p7)
        *Dg(nu3,nu4,p8)
        *S(mu1,q1,p3m,nu3,q1,p2m,nu1,q1,p1m,mu2,p6m,nu2,p5m,nu4,p4m)
        );

        #define TOPOLOGY "LA"
*--#] ladder:
```

Some words concerning the notation are in order: The function S(...) represents the fermion string and contains momenta (p1 $\widehat{=} p_1$, ...) and Lorentz indices (mu1 $\widehat{=} \mu_1$, ...) as its arguments. The momenta are labeled in accordance with the MINCER conventions for the specific topology which is LA here, meaning "ladder". Note that it is more convenient to express the input in terms of eight momenta and to use momentum conservation at a later stage than to impose it already in the input. The ending m indicates that massive fermions should be considered. A small FORM program transforms each Lorentz index to the corresponding γ matrix and the momenta to fermion propagators. The expansion in the external momentum, q1, is indicated by the consecutive appearance of q1 together with an integration momentum. The depth of the expansion, which is encoded as a pre-processor variable in FORM, must be defined at program call. Dg(...) represents the gluon propagator. After reading the diagram these functions are replaced according to the Feynman rules. Then MATAD evaluates the fermion trace and expands the denominators with respect to q1. At this point the user may influence the procedure by providing an additional file, the so-called "special treat-file". For the matter in hand, this file contains a projector on the transverse part of the diagram:

```
multiply, (-d_(mu1,mu2)+q1(mu1)*q1(mu2)/q1.q1)*deno(3,-2);
```

where `deno(3,-2)` means $1/(3-2\varepsilon)$. In this way one obtains a scalar expression where `q1` is absent in the denominators of the propagators. However, it may still appear in scalar products with integration momenta. Application of d'Alembertian operators performs an averaging in the numerator, i.e. terms like $(q_1 \cdot p)^2$ are replaced by $p^2 q_1^2/D$. This projects out the coefficients of $(q_1^2)^n$ which have no dependence on the external momenta any more. After that it is possible to decompose the numerators in terms of the denominators. At this stage the expression may already have split into millions of different integrals. Recurrence relations derived with the integration-by-parts technique reduce them to a small set of tabulated master integrals which have been evaluated once and for all. In the case of the photon propagator, for example, all three-loop integrals reduce to only three massive three-loop master integrals. The same is true for the massless case with non-vanishing external momentum, which is where MINCER applies.

In intermediate steps, when the recurrence relations are applied, it may happen that the expression blows up and disk space up to several Gigabytes is needed. But the final result is rather compact and fills only of the order of one output screen (depending, of course, on the depth of the expansion).

3.2.2 EXP and LMP

The programs EXP and LMP are computer-technical realizations of the prescriptions for the asymptotic expansion of Feynman diagrams described in Section 2.2. They both work at the graphical level, i.e. they are not dealing with the analytic expressions corresponding to the Feynman diagrams but rather with the diagrams themselves.

Given an arbitrary diagram with a single external momentum (not counting momenta that can be gotten rid of by a simple Taylor expansion), EXP and LMP generate all possible subdiagrams by successively removing any combination of lines (propagators). For those subdiagrams matching the conditions of the large-momentum, respectively, the hard-mass procedure, the programs determine the momentum distribution. In particular, they determine which lines should be Taylor expanded in which momenta, and how the momenta of the hard subgraph are related to the ones of the co-subgraph. Thus, every diagram is split into several subdiagrams, all of them being products of single-scale integrals as discussed in Section 2.2.

As it was already mentioned in Section 3.1.3, LMP is written in PERL and is only concerned with the large-momentum procedure. EXP, being in some sense its successor, is a FORTRAN 90 program[3] and deals also with the hard-mass procedure. In addition, if several scales are involved, it repeatedly applies both methods in combination, thus reducing an arbitrary three-loop self-energy diagram to single-scale integrals (see Section 4.2).

To be specific, let us consider the example for the double-bubble diagram of Section 2.2.2 where the outer line is massless and the inner one carries mass m. The input file for LMP looks as follows:

```
*--#[ GLOBAL :
#define POWER "12"
*--#] GLOBAL :

*--#[ TREAT :
multiply, (-d_(mu,nu) + q(mu)*q(nu)/q.q)*deno(3,-2);
*--#] TREAT :

*--#[ db :
    *S(mu,p1,ro,p2,nu,p3,si,p4)
    *SS(tau,p6m,al,-p7m)
    *Dg(tau,ro,p8)
    *Dg(si,al,p5)
    ;
    #define TOPOLOGY "01"
*--#] db :
```

where the fold construct of FORM is used not only to encode the diagram itself but also the projectors (TREAT-fold) and possible options (GLOBAL-fold). The projector is again on the transverse part of the diagram (see

[3]A forthcoming version will be written in C.

Section 3.2.1). The external momentum is denoted by q this time. The only option here is POWER, denoting the required depth for the expansion in the mass m.

The diagram is contained in the fold named db (for "double bubble"). The encoding is based on MINCER/MATAD notation (see Section 3.2.1). In particular, the momentum distribution is in accordance with the corrsponding topology defined in these programs.

LMP reads the content of the db-fold, translates it to an internal graph-based notation and applies the large-momentum procedure. There are two main output files; the first one is called db.dia and contains the relevant subdiagrams:

```
*--#[ db_1 :
    *S(mu,+p1,ro,+p2,nu,+p3,si,+p4)
    *SS(tau,+p6mexp,al,-p7mexp)
    *Dg(tau,ro,+p5)
    *Dg(si,al,+p5)
    ;
#define DIANUM "1"
#define TOPOLOGY "arb"
#define INT1 "inpo1"
#define MASS1 "0"
*--#] db_1 :

[...]

*--#[ db_5 :
    *S(mu,+p11,ro,+p12,nu,+p13,si,+p14)
    *SS(tau,p1m,al,+aexp,-p15mexp)
    *Dg(tau,ro,+p15)
    *Dg(si,al,+p15)
    ;
#define DIANUM "5"
#define TOPOLOGY "arb"
#define INT1 "topL1"
#define MASS1 "M"
#define INT2 "inpt1"
#define MASS2 "0"
*--#] db_5 :

[...]
```

Only two out of six contributing subdiagrams are displayed here, the remaining ones being represented by [...]. The information on the topology of the diagrams is now split according to the factorization of loops. It is contained in the pre-processor variables INT1, INT2, etc. The variable TOPOLOGY is set to the dummy value arb.

The first subdiagram, db_1, corresponds to the naive Taylor expansion of the integrand with respect to m. This can be seen from the fact that the momentum distribution is the same as in the original diagram, and that the massive lines are denoted by p$i$mexp instead of pim. In this way MATAD is advised to perform a Taylor expansion in m. The second subdiagram displayed above, db_5, corresponds to the second one on the right-hand side of Fig. 7 (note that there is a topologically identical one, indicated by the factor 2 in Fig. 7; LMP, however, generates both of them as separate diagrams). The co-subgraph is a one-loop tadpole-diagram carrying the momentum p_1. It must not be expanded in the mass, therefore its propagator is simply denoted by p1m above. The hard subgraph is a two-loop diagram of the topology shown in Fig. 1. Its momenta p_i $(i = 1,\ldots,5)$ are denoted by p1i, where the additional "1" is introduced to distinguish them from the momenta (here it is only one momentum, actually) of the co-subgraph. One propagator carries the momentum $p_1 - p_{15}$ which is indicated by the notation +aexp,-p15mexp. Note that this propagator also has to be expanded w.r.t. m. Here, p_1 is denoted by aexp to let MATAD know that it has to expand with respect to p_1. After expansion, aexp is identified with `p1`. This is one of the main contents of the second output file of `LMP`, called `treat.db`. It again splits into folds, one for each subdiagram. The fold corresponding to the subdiagram `db_5` above, for example, contains the line

```
id aexp = + p1;
```

which does the identification of momenta mentioned above. Most of the remaining statements of the `treat.db`-file are concerned with the proper "power-counting", i.e., to take care that the Taylor expansions are performed such that not too few and not too many terms are generated. (Expanding up to unnecessarily high power may drastically slow down the computation.)

To compute the second example of Section 2.2.2, the one concerned with the hard-mass procedure, one has to use `EXP`. The input file, however, is almost identical to the one shown above. The only difference is the `GLOBAL`-fold, since in the case of `EXP` also the hierarchy of scales has to be fixed (for `LMP`, only $q^2 \gg m_1^2, m_2^2, \ldots$ is allowed):

```
*--#[ GLOBAL :
#define POWERM "6"
#define SCALES "M,q"
*--#] GLOBAL :
```

This means to assume $m^2 \gg q^2$ and to expand up to $(q^2/m^2)^3$. For multi-scale problems, this fold may look like

```
*--#[ GLOBAL :
#define POWERMa "4"
#define POWERMc "2"
#define POWERQ "8"
#define SCALES "Ma,q,Mc,Mb"
*--#] GLOBAL :
```

indicating the hierarchy $m_a^2 \gg q^2 \gg m_c^2 \gg m_b^2$ and an expansion up to the corresponding degrees. The output files of `EXP` are very similar to the ones above, although the power-counting statements are obviously much more involved than for `LMP`.

In addition, `LMP` and `EXP` produce the whole set of administrative files, namely `FORM` files which control the running of `MINCER/MATAD`, and `make` files concerned with program calls and the reduction of loss caused by system crashes.

The only task to be done by the user is to feed in the input file shown above, call `LMP` or `EXP`, and finally to type "`make`" in order to start the very computation of the diagrams. The output is exactly a result like the one quoted in Eq. (41) and (42), respectively.

3.2.3 GEFICOM

This section investigates in more detail how to use `GEFICOM` to evaluate a whole set of Feynman diagrams. Fig. 10 gives a schematic overview of `GEFICOM` and the possible interactions from outside. It unifies several packages described above and thus allows to automatically compute a specified class of diagrams in analytic form as long as a certain hierarchy among the involved mass scales is justified. The strategy then is to apply asymptotic expansions in order to split all integrals into massless propagator-type diagrams and massive tadpoles.

As an example let us consider corrections to the gluon propagator from which follows, for example, the gluon wave function renormalization constant. `GEFICOM` needs a name for each problem, and we will choose "`gg`" for it. A more sophisticated example involving up to four different dimensional parameters is considered in Section 4.2.

In a first step the user sets up a file containing the Lagrangian, or rather its topological content, i.e., the plain vertices and propagators without the corresponding analytical expressions. In our example it is called `gg.lag` and looks as follows:

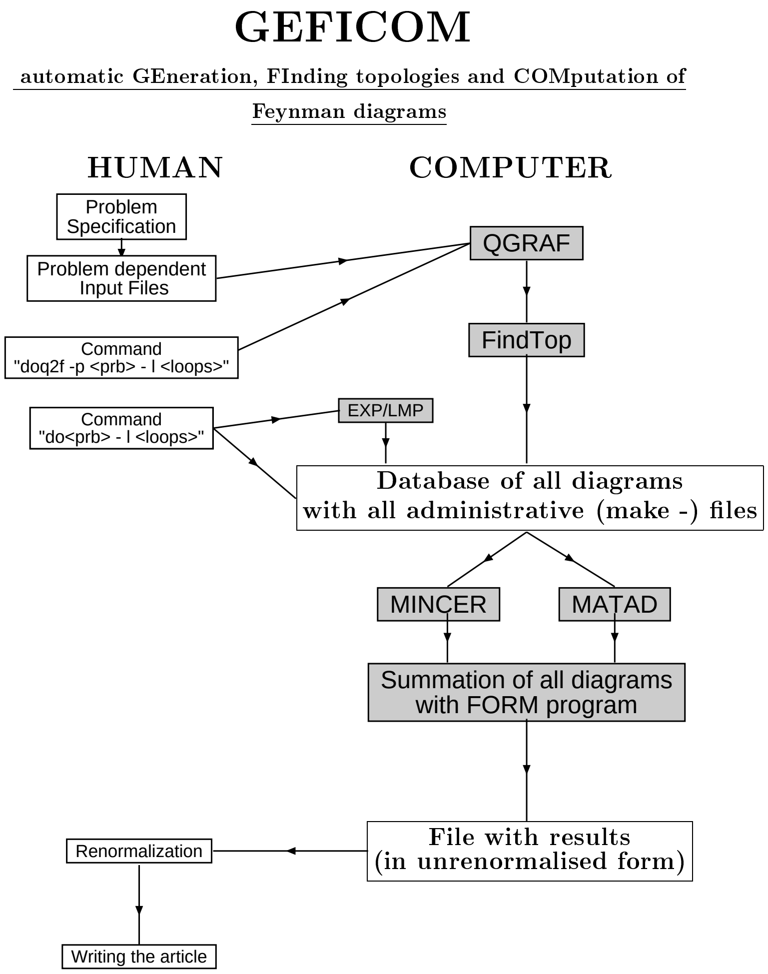


Figure 10: The structure of `GEFICOM`.

```
*  propagators
  [g,g;3+t]
  [c,C;1-]
  [q,Q;2-]
  [sigma,sigma;1+t]
*  vertices
  [Q,q,g]
  [C,c,g]
  [g,g,g]
  [g,g,sigma]
```

This file serves as input for `QGRAF`[4] and the notation is adopted from this program. In the first part the propagators are defined for gluons (`g`), ghosts (`c`) and quarks (`q`). The auxiliary particle σ (`sigma`) is introduced to split the four-gluon vertex into three-linear vertices which are much easier to deal with. In addition to the spin multiplicity one also has to specify if the particle is a boson ("+") or fermion ("−"). The optional parameter `t` guarantees that no diagrams with tadpoles of the corresponding particle are generated. The notation for the vertices in the second part is rather obvious.

A second file, called `gg.def`, contains the underlying process, accompanied by additional options:

```
*** MINCER

* gauge 1
```

[4]The notation of the `QGRAF` version 1.3 has been adopted.

```
* scheme 2
* power 0

      list = symbolic ;
      lagfile = 'q.lag' ;
      in = g[q];
      out = g[q];
      nloop = ;
      options = ;
      true = bridge[ g,c,q, 0,0 ];
```

The first line determines that the output of the generation procedure should be transformed into `MINCER` notation. The next two statements fix the gauge and the regularization scheme, and the variable `power` defines the depth of the expansion in small masses and momenta. Since in our case all particles are taken to be massless, only one scale is left, namely the external momentum. Thus, an expansion is not needed and the variable `power` may be assigned the value zero. The remaining part again uses `QGRAF` notation and specifies the diagrams to be generated. For example, "`in`" and "`out`" denote in- and outgoing states, and the last line excludes diagrams that are one-particle reducible w.r.t. gluons, ghosts or quarks. `options` is left empty, and `nloop`, the number of loops, will be defined via a command line option later.

Like in Section 3.2.1, the user again may provide "special `treat`-files" — in the case of the gluon propagator two of them. The first one, called `treat.gg.1`, again contains the projector on the transverse structure of the correlator:

$$\frac{1}{D-1}\left(-g^{\mu_1\mu_2}+\frac{q^{\mu_1}q^{\mu_2}}{q^2}\right) . \tag{46}$$

Besides the consideration in momentum space `GEFICOM` also takes care about the colour factor which is why in general one should supply a projector also in colour space. For the gluon propagator with the colour indices $a(1)$ and $a(2)$ it is given by

$$\frac{\delta^{a(1)a(2)}}{n_g} , \tag{47}$$

where $n_g = N_c^2 - 1$ is the number of gluons. The corresponding `FORM` file, called `treatcol.gg.1`, looks as follows:

```
multiply, prop(a(1),a(2))/ng;
```

After providing these simple input files, `GEFICOM` may be called by executing the shell-script command:

```
> doq2f -p <prb> -l <loops>
```

where `<prb>` gives the name of the problem (`gg` here) and `<loops>` is the number of loops. `GEFICOM` calls `QGRAF` and transforms the output to `Mathematica` format. Unfortunately `QGRAF` does not provide the topology of the generated diagrams. While for a human being it is very simple to figure out this information by just looking at the diagram, for a computer this is a non-trivial task, especially if the number of loops exceeds two. One of the core parts of `GEFICOM` indeed is a `Mathematica` program precisely concerned with this problem.

Once the topology is available, the notation for the diagrams is translated to a format which is suitable for the `FORM` packages `MINCER` and `MATAD`. In addition, a set of administrative files controlling the calculation and ensuring minimal loss in the case of computer breakdowns is generated. For the case of the gluon propagator considered in this section, at three-loop level 494 diagrams are generated, and it takes roughly 12 hours of CPU time on a DEC-Alpha machine with 600 MHz for the complete evaluation. A sample diagram for the three-loop gluon propagator is shown in Fig. 11.

In analogy to the momentum-space diagram also a `FORM` version of the "colour diagram" is generated which is evaluated with the help of a small `FORM` program.

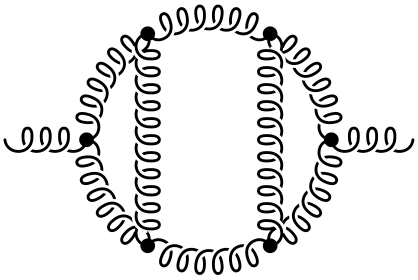


Figure 11: Sample diagram contributing to the gluon propagator at three loops.

The computation of the diagrams is not initiated automatically. Instead, GEFICOM generates a shell-script, do<prb> (dogg in our case) that allows, for example, only a subset of diagrams, or even only a single diagram, to be computed via command line options. The advantage of this strategy is that different computers may be used in parallel for different subsets, or that a part of the diagrams may even be evaluated before the generation is finished. After the results of all diagrams are available they are summed up, taking into account the correct colour (and other) factors and the (unrenormalized) result is at hand.

3.2.4 FeynArts, FeynCalc, FormCalc, TwoCalc and LoopTools

FeynArts: FeynArts is a Mathematica-package generating Feynman diagrams and the corresponding amplitudes. Its great flexibility has allowed it to become an enormously useful tool not only for high energy particle physicists. It has found its applications also in effective field theories, and by knowing its power one could imagine that it may be useful even for at first sight completely unrelated subjects.

There exists a manual describing its features in a rather clear and pedagogical way. The four following functions constitute the heart of FeynArts: CreateTopologies, InsertFields, Paint and CreateFeynAmp.

As an example let us consider the one-loop corrections to the triple Higgs vertex in the Standard Model. The use of CreateTopologies in this case is the following[5]:

```
tops = CreateTopologies[1,3, Tadpoles -> False, SelfEnergies -> False];
```

The first two arguments of CreateTopologies determine the number of loops and external legs, respectively. The remaining arguments are options, where the first one prevents creation of tadpole insertions and the second one states that no diagrams with self-energy insertions on external legs are generated[6]. To view the topologies, one types

```
Paint[tops,1,2];
```

where the first argument specifies the list of topologies and the second and third one refer to the number of incoming and outgoing fields, respectively. The outcome is the graphic shown in Fig. 12.

Next the fields are attributed to the lines. This is done by typing

```
ins = InsertFields[tops, {S[1]} -> {S[1],S[1]}, Model -> {SM}];
```

where the first argument again refers to the list of topologies created above, the third one fixes the model (SM $\hat{=}$ Standard Model), and the second one defines the in- and outgoing particles (S[1] for the physical Higgs boson). This produces a list of 64 diagrams that can be viewed using Paint again, this time, however, with only one argument, since in- and outgoing fields are already specified in "ins":

```
Paint[ins];
```

An extract of the list of diagrams is shown in Fig. 13.

[5]In what follows we will use the notation of the FeynArts version 1.2

[6]There is a single option comprising both of the latter, named WFCorrections (for "wave function corrections").

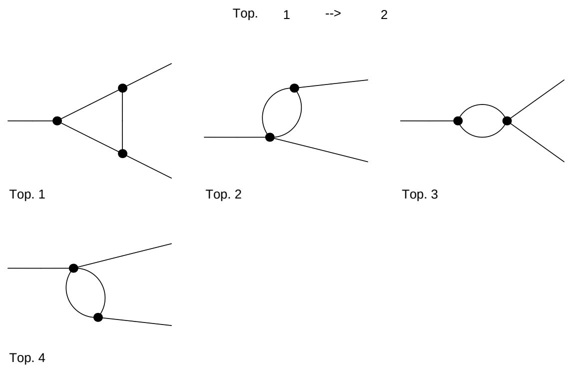


Figure 12: Topologies created by FeynArts.

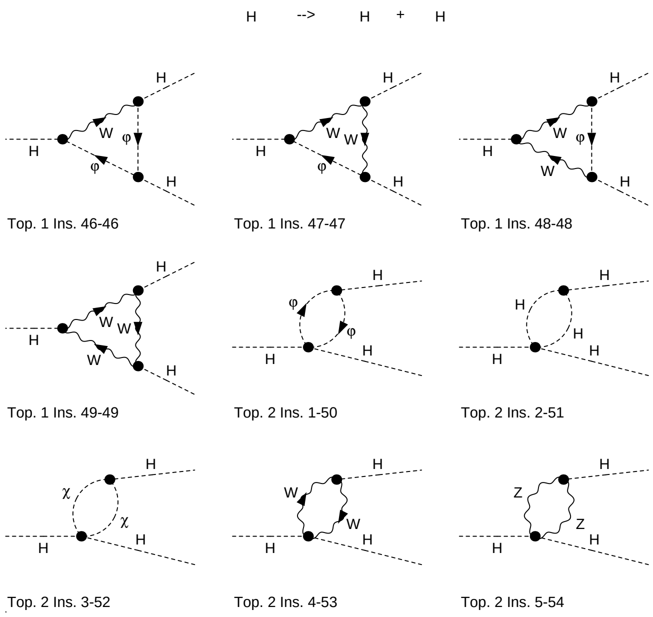


Figure 13: Sample of diagrams generated by FeynArts for the triple Higgs vertex to one-loop order.

The last step to be done by FeynArts is to generate analytic expressions which are understood by FeynCalc, FormCalc and TwoCalc:

```
amps = CreateFeynAmp[ins];
```

This produces a list containing the amplitudes of the individual diagrams. The 17^{th} element corresponds to the diagram with a top-quark triangle (Fig. 14) and reads as follows:

```
In[12]:= amps[[17]]

                                        3 I   3    3
Out[12]= FeynAmp[S1S1S1, T1, I17, N17][(--- EL   MT   Integral[q1]
                                        128

>       tr[(MT + gs[k1 - p1 + q1]) . (MT + gs[q1]) . (MT + gs[k1 + q1])]) /

         3   4      2      2       2             2       2                   2     3
>     (MW  Pi  (-MT  + q1 ) (-MT  + (k1 + q1) ) (-MT  + (k1 - p1 + q1) ) SW )]
```

pi are the ingoing, ki the outgoing, and qi the loop momenta. EL is the electric charge, gs[p] means $\not{p}$ and tr

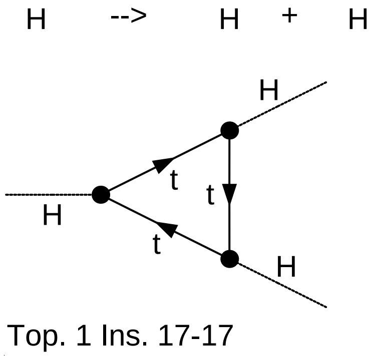


Figure 14: 17th diagram produced by `FeynArts` for the triple Higgs vertex.

the Dirac trace.

`FeynCalc`, `FormCalc` and `TwoCalc`: The list of expressions named **amps** above may now be directly fed into `FeynCalc`[7] or `FormCalc`. Its entries will automatically be transformed to the internal notation (see below). The key function of these programs is `OneLoop`, respectively derivatives of it, in particular `OneLoopSum`. As a simple example consider the integral

$$\int \frac{\mathrm{d}^D q}{(2\pi)^D} \frac{1}{(m_1^2 - q^2)(m_2^2 - (k+q)^2)} = \frac{i}{16\pi^2} B_0(k^2, m_1^2, m_2^2)\,. \tag{48}$$

In the notation of `FeynCalc` this reads

```
In[3]:= OneLoop[q,1/(2*Pi)^4 FeynAmpDenominator[
              PropagatorDenominator[q,m1],
              PropagatorDenominator[q+k,m2]]]

        I            2    2
        -- B0[k.k, m1 , m2 ]
        16
Out[3]= --------------------
                  2
                Pi
```

Note that the pre-factor in `In[3]` above is D-independent. In principle, this could be assigned to a definition of `B0` in `Out[3]` that differs from the one in (48) by a D-dependent factor; however, another interpretation is to implicitely assume $\overline{\mathrm{MS}}$-regularization of all integrals. In fact, as will become clear in a moment, this is what is done in the numerical routine `LoopTools`.

In this way `FeynCalc` rewrites any one-loop diagram to the standard integrals defined in Eq. (9). For example, the diagram corresponding to `amps[[17]]` defined above produces the output

```
In[4]:= OneLoopSum[amps,SelectGraphs -> {17}]

Out[4]= K[7]

In[5]:= FixedPoint[ReleaseHold[#]&,%]
```

[7] We use version 2.2β here which is the latest version which is freely available (see Section 3.1.5).

```
                 3    4              2    2          3    4              2    2
          -3 EL   MT   B0[k1.k1, MT  , MT ]   3 EL   MT   B0[p1.p1, MT  , MT ]
Out[5]= ----------------------------- - ---------------------------- -
                            3    2    3                          3    2    3
                      32 MW   Pi   SW                      32 MW   Pi   SW

       3    4                              2    2
  3 EL   MT   B0[k1.k1 - 2 k1.p1 + p1.p1, MT  , MT ]
  ----------------------------------------------- -
                             3    2    3
                       32 MW   Pi   SW

        3    4                                              2    2    2
  (3 EL   MT   C0[k1.k1, p1.p1, k1.k1 - 2 k1.p1 + p1.p1, MT  , MT  , MT ]

           2                                      3    2    3
     (4 MT   - k1.k1 + k1.p1 - p1.p1)) / (32 MW   Pi   SW )
```

The action of `FormCalc` is quite similar to the one of `FeynCalc`, except that is does not perform the full reduction to the scalar integrals of Eq. (11), but rather only decomposes the expressions into covariants, keeping the corresponding coefficients as they are. In general this reduces the length of the expressions and still can be evaluated numerically with the help of `LoopTools`, to be described below. The same strategy can also be followed by `FeynCalc`, simply by setting the option `ReduceToScalars` in `OneLoop` to `False`. The main feature of `FormCalc` in comparison to `FeynCalc` is its speed which is mainly due to the fact that it passes large expressions to `FORM`.

`TwoCalc` is the extension of `FeynCalc` to two-loop propagator type diagrams, and its usage is very similar.

`FF` and `LoopTools`: `FF` is a `FORTRAN` program which numerically evaluates the scalar one-loop integrals defined in (11). The extension to the coefficients appearing in the tensor decomposition of tensor integrals and its integration into `Mathematica` is done in `LoopTools`. E.g., the numerical evaluation of the B_0-integral defined above may be simply done in `Mathematica` by saying

```
In[2]:= B0[k.k,m1^2,m2^2] //. {k.k -> m1^2,m1 -> 100,m2 -> 10}

Out[2]= -7.49109
```

This example shows that the $1/\varepsilon$ poles (together with the constants $\gamma_{\rm E}$ and $\ln 4\pi$) are set to zero as it corresponds to an $\overline{\rm MS}$-renormalized quantity; this was already pointed out in the discussion below Eq. (48). Both the ultra-violet and the infra-red divergent parts are controlled with the help of parameters which specify the square of the renormalization scale μ. In a similar way, it is possible to obtain numerical values for the diagram `amps[[17]]` above by providing numbers for the scalar products and masses.

3.2.5 `CompHEP`

Let us now discuss in more detail how the package `CompHEP` works. As already mentioned in Section 3.1.4, `CompHEP` essentially consists of two parts: the symbolic and the numerical one. A flowchart representing the symbolic part is shown in Fig. 15.

After choosing the underlying model in *menu 1*, one gets to *menu 2* where the process must be defined and certain diagrams may be discarded from further considerations with the help of *menu 4*. One may also modify the parameters of the selected model (see *menu 3*). *menu 5* is concerned with the graphical output and the algebraic squaring of the matrix elements for the generated diagrams.

In *menu 6*, analytical expressions for the squared matrix elements are computed. They can be stored to disk either in `C`, `FORTRAN`, `REDUCE` or `Mathematica` format (see *menu 8*). Instead of applying the helicity-amplitude technique, `CompHEP` uses the conventional method of squaring the amplitudes before calculating traces. Whether

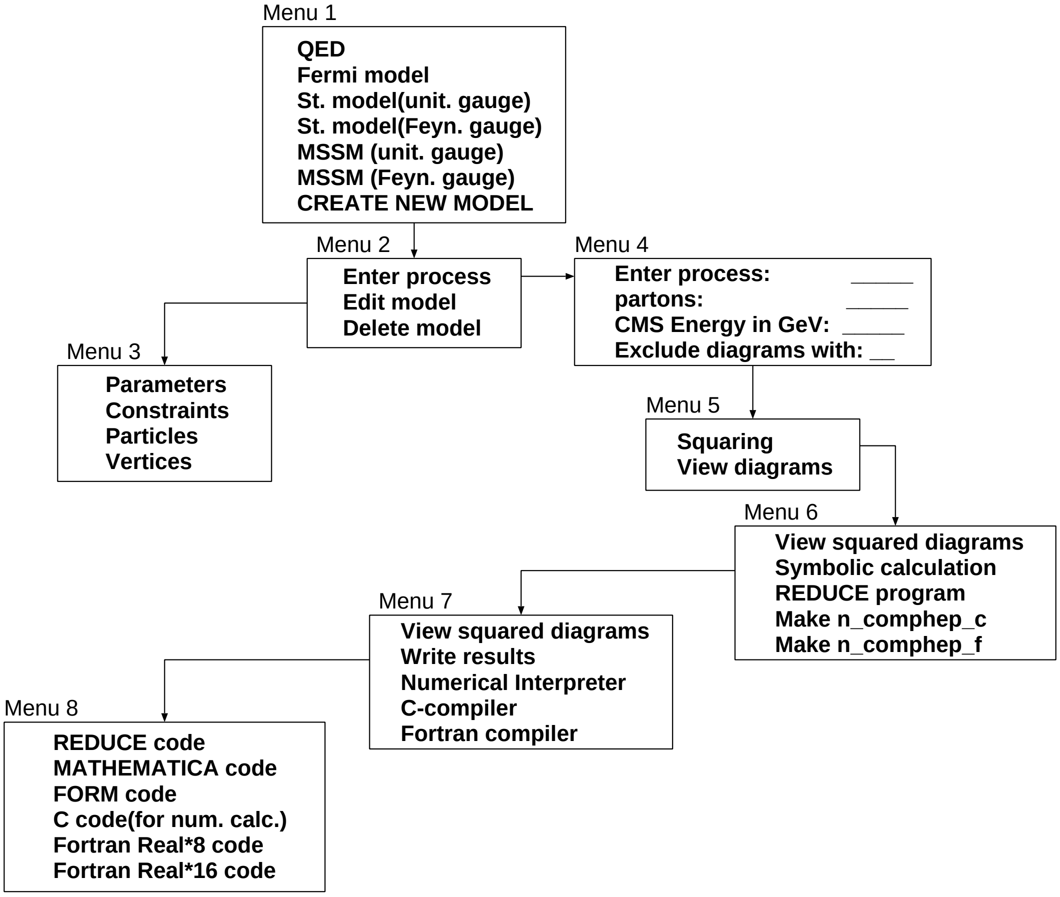


Figure 15: Menus of the symbolic part of CompHEP [135].

menu 1

Subprocess	IN state
Model parameters	QCD scale
Breit-Wigner	Cuts
Kinematics	Regularization
Vegas	Simpson
Batch	

Figure 16: Main menu of the numerical part of CompHEP [135].

this is an advantage or a disadvantage depends on the specific problem under consideration. For simple processes a built-in *numerical interpreter* (see *menu 7*) is available to perform the phase space integration. For more complicated problems one has to initiate the actual numerical part of CompHEP. It deals with processes that exceed the capability of the built-in calculator.

The main options of the numerical part of CompHEP are displayed in Fig. 16. Before starting the computation, the environment for the Monte Carlo integration (using VEGAS) must be set. This includes, for example, the modification of model parameters and the choice of the center-of-mass energy. Furthermore, the user decides whether structure functions for the incoming particles should be used or not. A central point is represented by the item *Regularization*, where the integration variables, previously defined in the item *Kinematics*, are mapped in such a way that the integrand becomes a smooth function. This is one of the main features of CompHEP: The users task is just to provide possible singularities in the propagators that are close to the edges of the phase space region; the mappings themselves are done completely automatically. Other items are concerned with cuts over energy, angles, transverse momenta, squared momentum transfers, and invariant masses or the rapidity for a set of outgoing particles. Note that all the numerical calculations may equally well be performed in batch mode which is useful as soon as the run-time becomes large.

In order to reduce the CPU time, CompHEP saves the matrix elements computed during a VEGAS run into a file. This file is short because it contains only one number for each event. Later-on, a special program called `genEvents` repeats the Monte Carlo evaluations, reading the previously calculated matrix elements from the file instead of performing a re-calculation. Thus the CPU time is drastically reduced for this second run, being typically of the order of only a few minutes as compared to a few hours for the initial run. Therefore, it is possible to study the influence of various additional cuts on the cross-section and to fill histograms rather quickly by performing several subsequent runs of `genEvents`. Each working session produces two output files: one containing the results of the calculation together with a list of model parameters and a copy of the screen report, and a second one containing the computed matrix elements.

4 Applications

This section reports on recent results that would have been impossible to obtain without the use of computer algebra. The selected examples are supposed to underline the fields of applications for some of the programs described above.

The first example is concerned with one of the most classical subjects for multi-loop calculations, the photon polarization function. We will describe its evaluation, using the program `MATAD` in the limit $q^2 \ll m^2$, and additionally `MINCER` and `LMP` in the opposite case, $q^2 \gg m^2$. The technique of repeatedly applying the hard-mass procedure was successfully applied to the decay rate of the Z boson into b quarks using `EXP` in the `GEFICOM` environment. The calculation of renormalization group functions was the driving item for `BUBBLES` to be developed. As part of the "NIKHEF-setup" it was used to compute four-loop tadpole diagrams up to their simple poles in ε. Another example for the calculation of a four-loop quantity is the decay rate of the Higgs boson into gluons. Applying the approach of an effective Hamiltonian, `GEFICOM` could be used to compute the corresponding diagrams. As an application of the multi-leg program `CompHEP` the scenario of a strong interaction among electroweak gauge bosons will be discussed. It is traded as an alternative to the Higgs mechanism to restore unitarity at high energies. Calculations in the full electroweak theory usually involve a large number of different scales. For important contributions to the parameter Δr it was possible to reduce the contributing diagrams to two-point functions which then were accessible with the help of the programs `FeynArts` and `TwoCalc`.

4.1 QCD corrections to the photon polarization function

Notation and methods

This section is concerned with the computation of QCD corrections to the photon propagator. We define the polarization function in the following way:

$$\left(-g_{\mu\nu}q^2 + q_\mu q_\nu\right) \Pi(q^2) \quad = \quad i\int \mathrm{d}^4x\, e^{iqx}\langle 0|Tj_\mu(x)j_\nu(0)|0\rangle\,, \tag{49}$$

where $j^\mu = \bar\psi\gamma^\mu\psi$ is the diagonal vector current of two quarks with mass m. The main motivation is the simple connection of $\Pi(q^2)$ to the physical quantity $R(s)$ which is defined as the normalized total cross section to the production of heavy quarks:

$$R(s) \quad \equiv \quad \frac{\sigma\left(e^+e^- \to \text{hadrons}\right)}{\sigma\left(e^+e^- \to \mu^+\mu^-\right)} \quad = \quad 12\pi \mathrm{Im}\Pi(q^2 = s + i\epsilon)\,. \tag{50}$$

The advantage of making a detour to $\Pi(q^2)$ and not considering $R(s)$ from the very beginning is the implicit summation of real radiation and infra-red singularities. The fact that the number of loops to be evaluated for $\Pi(q^2)$ is larger by one is made up for by having to deal with two-point functions only, for which a huge amount of technology is available.

It is convenient to separately consider the following contributions:

$$\Pi(q^2) \quad = \quad \Pi^{(0)}(q^2) + \frac{\alpha_s(\mu^2)}{\pi}C_F\Pi^{(1)}(q^2) + \left(\frac{\alpha_s(\mu^2)}{\pi}\right)^2 \Pi^{(2)}(q^2) + \ldots\,, \tag{51}$$

$$\Pi^{(2)} \quad = \quad C_F^2\Pi_A^{(2)} + C_AC_F\Pi_{NA}^{(2)} + C_FTn_l\Pi_l^{(2)} + C_FT\Pi_F^{(2)}, \tag{52}$$

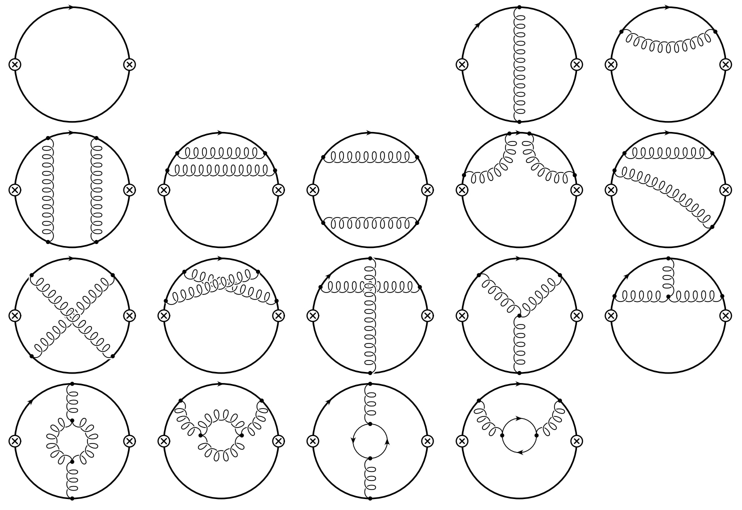


Figure 17: Diagrams contributing to the one-, two- and three-loop polarization function. Solid lines represent quarks, loopy ones are gluons.

and similarly for $R(s)$. The colour factors $C_F = (N_c^2 - 1)/(2N_c)$ and $C_A = N_c$ correspond to the Casimir operators of the fundamental and adjoint representations of $SU(3)$, respectively. The case of QCD corresponds to $N_c = 3$, the trace normalization of the fundamental representation is $T = 1/2$, and the number of light (massless) quark flavours is denoted by n_l. In Eq. (52), $\Pi_A^{(2)}$ is the Abelian contribution (corresponding to quenched QED) and $\Pi_{NA}^{(2)}$ is the non-Abelian part specific for QCD. There are two fermionic contributions arising from diagrams with two closed fermion lines, so-called double-bubble diagrams: For $\Pi_l^{(2)}$ the quark in the inner loop is massless, the one in the outer loop massive, whereas for $\Pi_F^{(2)}$ both fermions carry mass m. The case where the external current couples to massless quarks and these via gluons to massive ones will not be addressed here. Its imaginary part was considered in [87]. The Feynman diagrams contributing to one-, two- and three-loop order are shown in Fig. 17.

The two-loop corrections of $\mathcal{O}(\alpha_s)$ were computed in analytic form in the context of QED quite some time ago [136]. In subsequent works the calculation was redone and more convenient representations were found [137–139]. It is, at least with the currently available technology, out of range to compute the three-loop diagrams in complete analytic form for arbitrary q and m. Only for a subclass of diagrams, namely the ones with two closed fermion lines where one of them is massless, was it possible to compute the corresponding contribution to $R(s)$ analytically [87, 140, 88, 141]. Nevertheless, there are essentially two approaches which provide very good approximations for $\Pi^{(2)}(q^2)$, both of them relying heavily on the use of computer algebra.

One method is to consider the polarization function $\Pi(q^2)$ in the limit $q^2 \gg m^2$. Application of the large-momentum procedure (see Section 2.2) leads to an expansion in m^2/q^2 with the coefficients still depending on logarithms $\ln(-q^2/m^2)$. The idea is to evaluate as many terms as possible (the limitations essentially coming from the CPU time and the size of the intermediate expressions) and thus to approximate the true function $\Pi(q^2)$ even for rather small values of q^2.

Whereas the above method results in analytical expressions, an alternative way is to construct a semi-numerical result for $\Pi(q^2)$ which will, however, be valid for all values of q and m. In addition to the high energy expansion discussed above, this method also requires the expansion of $\Pi(q^2)$ in the limit $q^2 \ll m^2$. Together with some information about the (two-particle) threshold $q^2 = 4m^2$, a suitable conformal mapping and the use of Padé approximation, one obtains an approximation for $\Pi(q^2)$ over the full kinematical region [20, 142].

Referring to the literature for physical applications (e.g., [143, 144]), in the following we will concentrate on the technical aspect of the calculation in the limits $q^2 \ll m^2$ and $q^2 \gg m^2$. While the latter case directly corresponds to the first method above, the former one is needed for the second method.

As can be seen from Fig. 17 only a small number of diagrams contribute at three-loop level. This makes the use of a diagram generating program like `QGRAF` unnecessary. The main challenge instead is to compute as many terms in the expansions as possible. Apart from fast computers this requires an efficient algebraic language and optimized programs to deal with a large number of terms. The choice for this problem was `MATAD` and

MINCER, both written in FORM (see Section 3.2.1).

Large mass limit

In the limit $q^2 \ll m^2$ application of the hard-mass procedure (see Section 2.2) shows that it suffices to keep the naive Taylor expansion of the diagrams in their external momentum. Thus one stays with the calculation of massive tadpole integrals. Let us consider the moments C_n of the polarization function, defined through the Taylor series

$$\Pi^{(2)}(q^2) \;=\; \frac{3}{16\pi^2}\sum_{n>0} C_n^{(2)}\left(\frac{q^2}{4m^2}\right)^n \,, \tag{53}$$

where m is the on-shell mass. Although this series does not develop an imaginary part, one may gain information on the rate from it by exploiting the analyticity of $\Pi(q^2)$. We do not want to discuss this approach here and refer the interested reader to [145, 20]. At three-loop level the evaluation of the coefficients up to C_8 was performed in [20]. The calculation required disk space of the order of several GB for FORM (so-called "formswap") to store the intermediate expressions. The total CPU time on a 256 MHz DEC-Alpha workstation with 128 MB main memory was roughly two weeks. With the input for the ladder-type diagram already shown in Section 3.2.1, let us now have a look at the output of MATAD. The expansion up to terms of order $(q^2/m^2)^4$ reads:

```
ladder =
    + ep^-3 * (  - 8/9*Q.Q )

    + ep^-2 * (  - 1/693*Q.Q^5*M^-8 + 8/945*Q.Q^4*M^-6 - 2/35*Q.Q^3*M^-4 +
      8/15*Q.Q^2*M^-2 - 134/27*Q.Q )

    + ep^-1 * (  - 20*M^2 - 4042699/108056025*Q.Q^5*M^-8 + 273004/1488375*
      Q.Q^4*M^-6 - 63029/66150*Q.Q^3*M^-4 + 10658/2025*Q.Q^2*M^-2 - 473/81*
      Q.Q - 4/3*Q.Q*z2 )

    + 96*z3*M^2 - 482/3*M^2 - 19508/225*Q.Q^5*z3*M^-8 - 1/462*Q.Q^5*M^-8*z2
       + 129728592122581/1248047088750*Q.Q^5*M^-8 + 42496/675*Q.Q^4*z3*M^-6
       + 4/315*Q.Q^4*M^-6*z2 - 559517166977/7501410000*Q.Q^4*M^-6 - 44*
      Q.Q^3*z3*M^-4 - 3/35*Q.Q^3*M^-4*z2 + 6119442979/125023500*Q.Q^3*M^-4
       + 328/27*Q.Q^2*z3*M^-2 + 4/5*Q.Q^2*M^-2*z2 - 880963/121500*Q.Q^2*
      M^-2 + 11909/486*Q.Q - 392/9*Q.Q*z3 - 67/9*Q.Q*z2;
```

Note that this expression still contains an overall factor q^2 (the results above are in Euclidean space, indicated by the capital Q, i.e. $Q^2 = -q^2$). After dividing by q^2 there are terms of $\mathcal{O}(m^2/q^2)$ which cancel in the sum of all three-loop diagrams. The constant terms disappear after requiring the QED-like on-shell condition $\Pi(0) = 0$, and finally the structure of Eq. (53) is obtained.

After all diagrams up to three loops are computed and added up (taking into account the correct colour factors), the parameters α_s and m are renormalized. To give an impression of the structure of the final result, we list the first two and the eighth term of the Abelian part, $\Pi_A^{(2)}$:

$$\begin{aligned}
C_{A,1}^{(2)} &= -\frac{8687}{864} - \frac{32}{5}\,\zeta_2\,\ln 2 + 4\,\zeta_2 + \frac{22781}{1728}\,\zeta_3\,,\\
C_{A,2}^{(2)} &= -\frac{223404289}{1866240} - \frac{192}{35}\,\zeta_2\,\ln 2 + \frac{24}{7}\,\zeta_2 + \frac{4857587}{46080}\,\zeta_3\,,\\
&\dots\\
C_{A,8}^{(2)} &= -\frac{1903021824172553128988861156484526 91}{630638332036365850509312000 00}\\
&\quad - \frac{786432}{230945}\,\zeta_2\,\ln 2 + \frac{98304}{46189}\,\zeta_2 + \frac{31209476560803609727258477}{12432176686773043200}\,\zeta_3\,,
\end{aligned} \tag{54}$$

where $\zeta_n \equiv \zeta(n)$, with Riemann's ζ-function. For the specific values above it is $\zeta_2 = \pi^2/6$ and $\zeta_3 = 1.20206\ldots$. The left-out coefficients and the ones for the other colour structures of (52) can be found in [20].

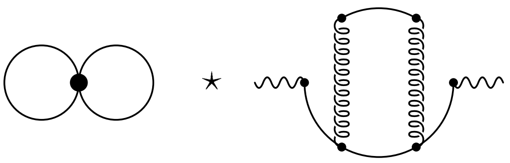


Figure 18: Sixth subdiagram as generated by LMP for the large-momentum procedure of the ladder diagram (cf. Fig. 9).

Large momentum limit

The calculation for $q^2 \gg m^2$ is more involved in the sense that here the asymptotic expansion does not reduce to a naive Taylor expansion. Instead, the large-momentum procedure to be applied in this case generates a large number of subgraphs for each of the diagrams of Fig. 17. Whereas for the small-momentum expansion a computer only had to be used for the computation of the diagrams, in this case even the generation of the subdiagrams is a non-trivial task and program packages like `EXP` or `LMP` (see Section 3.2.2) are indispensable. For the three-loop case the 18 initial diagrams produce 240 subgraphs and their manual selection would be very tedious and error-prone if not impossible. The corresponding numbers in the two- (one-) loop case are 2 (1) initial and 14 (3) subdiagrams which would still allow a manual treatment. These numbers also show that there is a big step in complexity when going from two to three loops.

For the calculation the package `LMP` (see Section 3.2.2) was used. As an example let us again consider the ladder diagram of Fig. 9. The input file, looking similar to the one of the low-energy case above, produces 27 subdiagrams. The one of Fig. 18 (the sixth subgraph generated by `LMP`), for example, corresponds to the following code:

```
*--#[ d3l3_6 :
        ((-1)
        *Dg(nu1,nu2,+aexp,-p2)
        *Dg(nu3,nu4,-bexp,+p2)
        *S(nu2,+p1mexp,nu4,-bexp,+qmexp,mu2,-p21m,nu3,-p2mexp,nu1,
           -p11m,mu1,-aexp,+qmexp)
        *1);
#define DIANUM "6"
#define TOPOLOGY "arb"
#define INT1 "inpl1"
#define MASS1 "0"
#define INT2 "topL1"
#define MASS2 "M"
#define INT3 "topL1"
#define MASS3 "M"
*--#] d3l3_6 :
```

After Taylor expansion, the integral factorizes into a massless and two massive one-loop integrals. The topology of each of these one-loop integrals is encoded in the pre-processor variables `INT1`, `INT2`, and `INT3`, where `inp..` denotes a massless, `top..` a massive topology (see also Section 3.2.2). The corresponding mass is given by `MASS1`, `MASS2`, and `MASS3`.

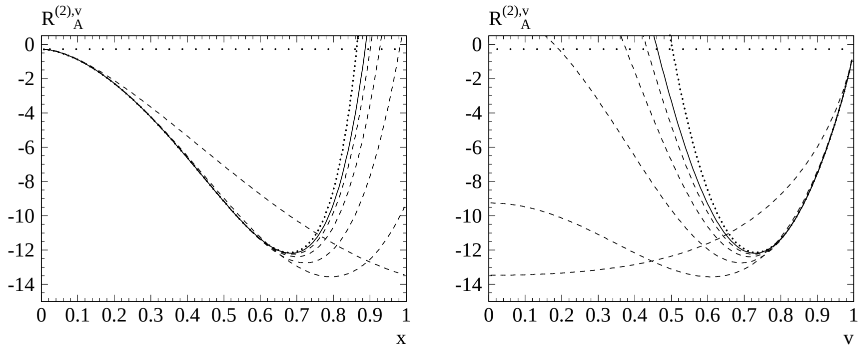
Again the first two and the highest available (in this case the seventh) term in the expansion for $\Pi_A^{(2)}$ will be displayed:

$$
\begin{aligned}
\bar{\Pi}_A^{(2)} &= \frac{3}{16\pi^2}\bigg\{ -\frac{143}{72} - \frac{37}{6}\zeta_3 + 10\,\zeta_5 + \frac{1}{8}\,l_{q\mu} \\
&\quad + \frac{\bar{m}^2}{q^2}\left[\frac{1667}{24} - \frac{5}{3}\zeta_3 - \frac{70}{3}\zeta_5 - \frac{51}{2}\,l_{q\mu} + 9\,l_{q\mu}^2\right] \\
&\quad + \cdots
\end{aligned}
$$

$$
\begin{aligned}
&+ \left(\frac{\bar{m}^2}{q^2}\right)^6 \Bigg[- \frac{420607059143}{19440000} - \frac{13059229}{2700}\,\zeta_3 - 1120\,\zeta_4 + 4000\,\zeta_5 + \frac{560}{3}\,B_4 \\
&\qquad + \left(\frac{133099291}{972000} + \frac{16842}{5}\,\zeta_3\right) l_{qm} + \frac{54076013}{7200}\, l_{qm}^2 + \frac{8575579}{2700}\, l_{qm}^3 \\
&\qquad + \left(\frac{13274779}{450} - \frac{142256}{15}\, l_{qm} - 8992\, l_{qm}^2\right) l_{q\mu} \\
&\qquad + \left(-10854 + 7560\, l_{qm}\right) l_{q\mu}^2 \Bigg]\Bigg\} + \ldots\,,
\end{aligned}
\tag{55}
$$

with $l_{qm} = \ln(-q^2/\bar{m}^2)$ and $l_{q\mu} = \ln(-q^2/\mu^2)$, ζ_n as in (54) and the additional values $\zeta_4 = \pi^4/90$, $\zeta_5 = 1.03693\ldots$. B_4 was analytically calculated in [52] and evaluates numerically to $B_4 = -1.76280\ldots$. $\bar{m}$ is the $\overline{\rm MS}$ mass which can easily be transformed to the on-shell scheme [55]. For the left-out terms in (55) we refer to [146, 147].

In Fig. 19 (a) the imaginary part of the Abelian contribution is plotted as a function of $x = 2m/\sqrt{s}$ including successively higher orders in x. Up to $x \approx 0.7$ very quick convergence is observed. For larger values of x, however, the inclusion of higher terms does not significantly improve the approximation. This becomes manifest in an even more drastic way in Fig. 19 (b), showing the same curves as functions of $v = \sqrt{1-4m^2/s}$. This choice of variable enlarges the threshold region and thus demonstrates the breakdown of the high-energy expansion close to threshold.



(a) (b)

Figure 19: The Abelian contribution $R_A^{(2)}$ as functions of (a) $x = 2m/\sqrt{s}$ and (b) $v = \sqrt{1-4m^2/s}$. Wide dots: no mass terms; dashed lines: including mass terms $(m^2/s)^n$ up to $n = 5$; solid line: including mass terms up to $(m^2/s)^6$; narrow dots: semi-analytical result obtained via Padé approximation [20].

4.2 Corrections of $\mathcal{O}(\alpha\alpha_s)$ to the decay of the Z boson into bottom quarks

In the examples considered so far essentially only the QCD part of the Standard Model is involved, which is, as already noted, the most important field for multi-loop calculations. The electroweak sector, on the contrary, resides on a broken gauge symmetry which is the reason why the diagrams in general carry a lot more scales. Higher order calculations are thus much more involved.

As an in some sense intermediate case, one may consider mixed electroweak/QCD corrections. An important example in this class are the $\mathcal{O}(\alpha\alpha_s)$ corrections to the decay rate of the Z boson into quarks. One may distinguish two cases, namely the decay into the light u, d, s or c quarks and the one into $b\bar{b}$. The latter case is of special interest as the top quark enters the vertex diagrams already at one-loop order and leads to corrections proportional to M_t^2. The complete $\mathcal{O}(\alpha)$ corrections were considered in [148, 149]. Since the mixed $\mathcal{O}(\alpha\alpha_s)$ correction to the Z decay into u, d, s and c turned out to be quite sizable [150], it was tempting to consider also the decay into bottom quarks at this order. The leading terms of $\mathcal{O}(G_F M_t^2)$ and $\mathcal{O}(G_F \ln(M_t^2/M_W^2))$ were computed in [151–154].

However, the full M_t dependence to $\mathcal{O}(\alpha\alpha_s)$ is currently out of reach. On the other hand, the top quark is certainly the heaviest particle entering the $Zb\bar{b}$ vertex at this order, which makes asymptotic expansions a well suited tool to obtain a very good approximation to the full answer.

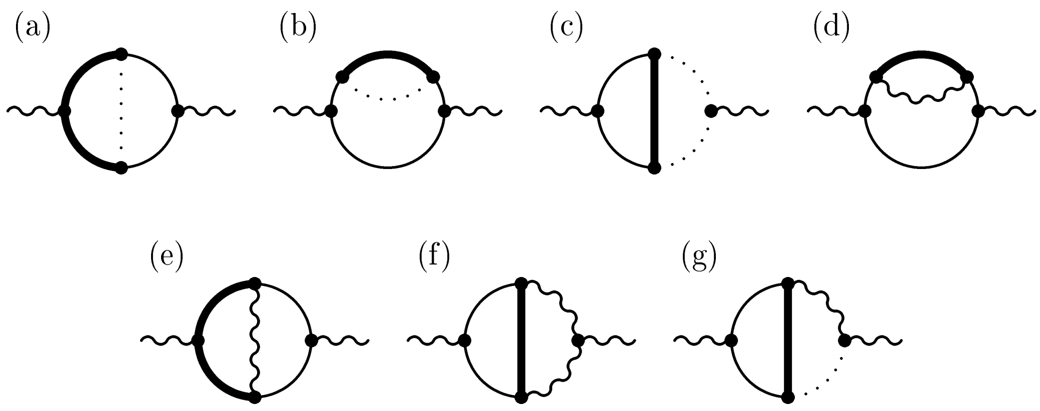
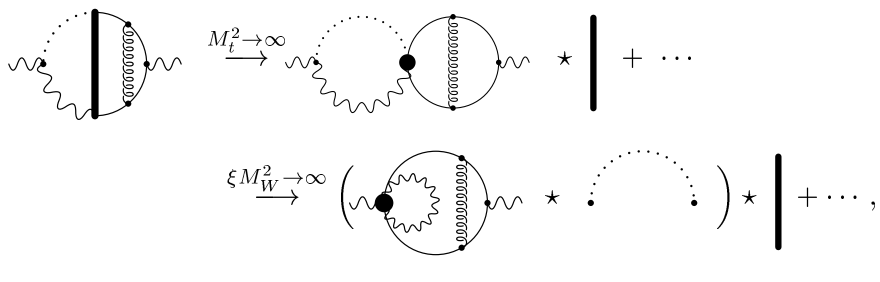


Figure 20: Diagrams containing a top quark that contribute to $Z \to b\bar{b}$. Thin lines correspond to bottom quarks, thick lines to top quarks, dotted lines to Goldstone bosons and inner wavy lines represent W bosons.

Let us describe the technical aspects of the calculation for $Z \to b\bar{b}$ in more detail. The $\mathcal{O}(\alpha\alpha_s)$ corrections are computed by applying the optical theorem, i.e., by evaluating the imaginary part of the Z boson propagator. Our concerns are only the diagrams involving the top quark. The ones contributing to $\mathcal{O}(\alpha)$ are shown in Fig. 20. The diagrams to be evaluated to $\mathcal{O}(\alpha\alpha_s)$ can be obtained by attaching a gluon in all possible ways, thereby increasing the number of loops by one.

The different scales in these diagrams are given by M_t, M_W, M_Z and M_Φ. The latter one is the mass of the charged Goldstone boson. It is related to M_W via $M_\Phi^2 = \xi_W M_W^2$, where ξ_W is the corresponding gauge parameter which is arbitrary and should drop out in the final result. In a first step we will consider it such that $M_t^2 \gg M_\Phi^2$. For the diagrams (a), (b), (d) and (e) the application of the hard-mass procedure with respect to M_t immediately leads to a complete factorization of the different scales, i.e., only single-scale integrals with at most three loops are left. For the diagrams (c), (f) and (g), however, some co-subgraphs still carry more than one scale which makes an evaluation quite painful, especially as their $\mathcal{O}(\varepsilon)$ part is also needed. It is very suggestive to apply the hard-mass procedure again to these co-subgraphs. In order to avoid unwanted imaginary parts arising from W and Φ cuts the hierarchy to be chosen among the physical masses is $M_t^2 \gg M_W^2 \gg M_Z^2$. The last inequality is seemingly inadequate, but a closer look at the corresponding diagrams shows that in this context it is actually equivalent to $4M_W^2 \gg M_Z^2$, respectively, $(M_W + M_t)^2 \gg M_Z^2$. The mass of the Goldstone boson may be incorporated using the possible hierarchies $\xi_W M_W^2 \gg M_W^2 \gg M_Z^2$ or $M_W^2 \gg \xi_W M_W^2 \gg M_Z^2$. This procedure leads to a nested series in M_W^2/M_t^2 and M_Z^2/M_W^2. Diagrammatically, an example for this procedure of successively applying asymptotic expansions looks as follows:



where only those terms are displayed which are relevant in the discussion above and all others contributing to the hard-mass procedure are merged into the three dots. The mass hierarchy is assumed to be $M_t^2 \gg \xi_W M_W^2 \gg M_W^2 \gg M_Z^2$.

The technical realization of the calculation was performed with the help of `GEFICOM` (see Section 3.2.3). The main new ingredient in comparison to the examples quoted so far is the use of the automated version of the hard-mass procedure and its successive application, implemented in the program `EXP`, in combination with the generator `QGRAF` and the `FORM` packages `MINCER` and `MATAD`. This automation not only avoids human errors, but also allows several checks of the results. For example, since the gauge parameter ξ_W is arbitrary, it is possible to choose it such that no longer the top quark, but rather the Goldstone boson is the heaviest particle involved in the process. Then one is left with the hierarchy $\xi_W M_W^2 \gg M_t^2 \gg M_W^2 \gg M_Z^2$. Note that for this choice the hard-mass procedure produces completely different subdiagrams than for the cases above, where M_t was considered to be the largest scale. Furthermore, while a manual treatment of both cases would almost double the effort, the automated version allows one to go from one hierarchy to the other by simply interchanging two input parameters, as we will see below.

To be concrete, let us consider the input file needed for `GEFICOM`. It is certainly more sophisticated than the examples mentioned so far, but this is mainly due to the enriched particle spectrum.

```
*** MINCER

* scheme 2
* gauge 1
* exp y ma mc mb q
* powerma 2
* powermb 4
* powermc 4
* mass t Ma
* mass Wp Mb
* mass pp Mc
* loops 1 true=iprop[t,0,0];
* loops 2 true=iprop[g,0,0];
* loops 2 false=iprop[t,0,0];
* loops 3 true=iprop[g,1,1];
* loops 3 false=iprop[t,0,0];

      list = symbolic ;
      lagfile = 'q.lag' ;
      in = Z[q];
      out = Z[q];
      nloop = ;
      options = onepi;
      true = iprop[Wm,pm,0,2];
      false = iprop[bq,0,1];
```

The meaning of most of the lines can be deduced by comparison with the example of Section 3.2.3. The mass of the top quark (`t` $\widehat{=} t$), the W (`Wp` $\widehat{=} W^+$), and the Goldstone boson (`pp` $\widehat{=} \Phi^+$) are denoted by `Ma`, `Mb` and `Mc`, respectively. The change in mass hierarchy mentioned before is achieved by rewriting the line

```
* exp y ma mc mb q
```

to

```
* exp y mc ma mb q
```

This input file is combined together with the one containing the vertices and propagators, which shall not be displayed here, in order to generate in a first step the relevant diagrams at three-loop level. After inserting the Feynman rules, `EXP` applies the hard-mass procedure and reduces all diagrams to single-scale integrals, if necessary by applying the procedure twice. `EXP` also produces the relevant administrative files which then call `MINCER` and `MATAD`. The runtime for `QGRAF` is of the order of a few seconds and it takes a few minutes for `EXP` to generate the subdiagrams. The time spent in the integration routines strongly depends on the required depth of the expansion, of course. The computation in [25] took about three weeks.

The contribution of the vertex corrections to the decay rate of the Z boson to bottom quarks, induced by the exchange of a W or a Goldstone boson, may be written in the following way:

$$\delta\Gamma_b^W = \delta\Gamma_d^W + (\delta\Gamma_b^W - \delta\Gamma_d^W) = \delta\Gamma_d^W + (\delta\Gamma_b^{0,W} - \delta\Gamma_d^{0,W}) = \delta\Gamma_d^W + \delta\Gamma_{b-d}^W\,, \tag{56}$$

where $\delta\Gamma_q^W$ and $\delta\Gamma_q^{0,W}$ denote renormalized and unrenormalized quantities. In the difference between the b and d contributions the relevant counter-terms drop out. This means that $\delta\Gamma_{b-d}^W$ is independent of the renormalization scheme and is therefore well suited for installation in data analyzing programs like the ones described, e.g., in [155]. In addition, $\delta\Gamma^W_{b-d}$ is also gauge independent. $\delta\Gamma^W_d$ has been computed in [150]. For convenience we list the result for $\delta\Gamma^W_{b-d}$ in numerical form [25]:

$$\begin{aligned}
\delta\Gamma^W_{b-d} &= \Gamma^0 \frac{1}{s^2_\theta}\frac{\alpha}{\pi}\Big\{ -0.50 + (0.71 - 0.48) + (0.08 - 0.29) + (-0.01 - 0.07) + (-0.007 - 0.006) \\
&\quad + \frac{\alpha_s}{\pi}\Big[1.16 + (1.21 - 0.49) + (0.30 - 0.65) + (0.02 - 0.21 + 0.01) + (-0.01 - 0.04 + 0.004)\Big]\Big\} = \\
&= \Gamma^0 \frac{1}{s^2_\theta}\frac{\alpha}{\pi}\Big\{ -0.50 - 0.07 + \frac{\alpha_s}{\pi}\Big[1.16 + 0.13\Big]\Big\}\,, \qquad (57)
\end{aligned}$$

with $\Gamma^0 = N_c M_Z \alpha/(12 s^2_\theta c^2_\theta)$, $s_\theta = \sin\theta_W$, θ_W being the weak mixing angle, $c^2_\theta = 1 - s^2_\theta$, and M_t the on-shell top mass. For Eq. (57) we used the values $M_t = 175$ GeV, $M_Z = 91.19$ GeV and $s^2_\theta = 0.223$. The numbers after the first equality sign correspond to successively increasing orders in $1/M_t^2$, where the brackets collect the corresponding constant, $\ln(M_t^2/M_W^2)$ and, if present, $\ln^2(M_t^2/M_W^2)$ terms. The numbers after the second equality sign represent the leading M_t^2 term and the sum of the subleading ones. The $\mathcal{O}(\alpha)$ and $\mathcal{O}(\alpha\alpha_s)$ results are displayed separately. Comparison of this expansion of the one-loop terms to the exact result of [149] shows agreement up to 0.01% which gives quite some confidence in the $\alpha\alpha_s$ contribution. One can see that although the M_t^2, M_t^0 and $M_t^0\ln(M_t^2/M_W^2)$ terms are of the same order of magnitude, the final result is surprisingly well represented by the M_t^2 term, since the subleading terms largely cancel among each other.

4.3 Four-loop β function

The β function is an object of common interest in any field theory, especially in non-Abelian ones. It describes the dependence of the corresponding coupling constant with respect to the energy scale. For QCD it is convenient to define

$$\mu^2\frac{\mathrm{d}}{\mathrm{d}\mu^2}\frac{\alpha_s(\mu^2)}{\pi} = \beta(\alpha_s) = -\left(\frac{\alpha_s(\mu^2)}{\pi}\right)^2\sum_{i\geq 0}\beta_i\left(\frac{\alpha_s(\mu^2)}{\pi}\right)^i. \qquad (58)$$

The β function is directly related to the renormalization factor Z_g defined through $\alpha_s^0 = Z_g^2\alpha_s$, where α_s^0 and α_s are the bare and renormalized coupling constant of QCD, respectively:

$$\beta(\alpha_s) = \frac{\alpha_s^2}{\pi}\frac{\partial}{\partial\alpha_s}Z_g^{2,(1)}\,, \qquad (59)$$

where $Z_g^{2,(1)}$ is the residue of Z_g^2 with respect to its Laurent expansion in ε. In Eq. (59) it is already indicated that in mass independent renormalization schemes, like the $\overline{\mathrm{MS}}$ scheme, the renormalization constants and thus also the β function only depend on α_s.

There are several ways to compute Z_g, all of them related through Slavnov-Taylor identities. One choice would be to combine the renormalization constants of the four-gluon vertex and the gluon propagator. At four-loop level, however, this requires the computation of about half a million diagrams. On the other hand, the approach of [23] was based on the relation

$$Z_g = \frac{\tilde{Z}_1}{\tilde{Z}_3\sqrt{Z_3}}\,, \qquad (60)$$

where $\tilde{Z}_1$ is the renormalization constant of the ghost-ghost-gluon vertex, and $\tilde{Z}_3$ and Z_3 are the ones of the ghost and gluon propagators, respectively. Thus the pole parts of the corresponding Green functions had to be computed up to four loops.

The roughly 50,000 contributing diagrams were generated by means of `QGRAF`. Introducing an artificial mass M in each propagator as infra-red regulator allowed a Taylor expansion to be performed with respect to all external momenta, leading to four-loop massive tadpole integrals. Recurrence relations based on the integration-by-parts algorithm were implemented in the program `BUBBLES` [113] to reduce the integrals to a minimal set of master integrals. At one-, two- and three-loop level only one, at four-loop level two master integrals are required. The colour factors of the individual diagrams were determined with the help of a `FORM` program [114]. To cope with the huge number of diagrams a special database-like tool called `MINOS` was developed. It controls the calculation and allows to conveniently access the results of single diagrams. The total CPU time for this calculation was of the order of a few months.

The final result which we will only quote for the physically most interesting case of QCD reads

$$\begin{aligned}
\beta_0 &= \frac{1}{4}\left(11-\frac{2}{3}n_f\right)\,,\\
\beta_1 &= \frac{1}{16}\left(102-\frac{38}{3}n_f\right)\,,\\
\beta_2 &= \frac{1}{64}\left(\frac{2857}{2}-\frac{5033}{18}n_f+\frac{325}{54}n_f^2\right)\,,\\
\beta_3 &= \frac{1}{256}\left[\left(\frac{149753}{6}+3564\zeta_3\right)-\left(\frac{1078361}{162}+\frac{6508}{27}\zeta_3\right)n_f\right.\\
&\quad\left.+\left(\frac{50065}{162}+\frac{6472}{81}\zeta_3\right)n_f^2+\frac{1093}{729}n_f^3\right]\,,
\end{aligned}\tag{61}$$

where n_f is the number of active flavours.

4.4 Hadronic Higgs decay

4.4.1 Introduction

As was already pointed out in the introduction, the only particle of the Standard Model not yet discovered is the Higgs boson. Thus it is necessary to learn as much as possible about potential production and decay mechanisms of such a particle in order to correctly interpret the signals in the detector. An important decay channel of the Higgs boson is the one into gluons. Although it is a loop-induced process it is numerically not negligible mainly due to the fact that the $\mathcal{O}(\alpha_s)$ corrections are very large [21, 22]. In this section it is shown how to compute the $\mathcal{O}(\alpha_s^2)$ corrections exploiting the powerful tools of automatic Feynman diagram computation. For completeness also the decay into quarks will be discussed.

The Higgs boson will be assumed to be of intermediate-mass range, i.e., $M_H \lesssim 2M_W$, which suggests the use of the approach of an effective Lagrangian where the top quark is integrated out. For convenience of the reader let us, in a first step, collect the relevant formulae. Detailed derivations can be found in [156,157,21,158].

The starting point is the Yukawa Lagrange density describing the coupling of the H boson to quarks. In the limit $m_t \to \infty$ it can be written as a sum over five operators [156,157] formed by the light degrees of freedom, with the corresponding coefficient functions containing the residual dependence on the top quark:

$$\mathcal{L}_Y = -\frac{H^0}{v^0}\sum_{q\in\{u,d,s,c,b,t\}} m_q^0\bar\psi_q^0\psi_q^0 \quad\longrightarrow\quad \mathcal{L}_{\rm eff} = -\frac{H^0}{v^0}\sum_{i=1}^{5} C_i^0\mathcal{O}_i'\,. \tag{62}$$

It turns out that only two of the operators, in the following called $\mathcal{O}_1'$ and $\mathcal{O}_2'$, contribute to physical processes. Expressed in terms of bare fields they read

$$\mathcal{O}_1' = \left(G_{\mu\nu}^{0\prime,a}\right)^2\,,\qquad \mathcal{O}_2' = \sum_{q\in\{u,d,s,c,b\}} m_q^{0\prime}\bar\psi_q^{0\prime}\psi_q^{0\prime}\,, \tag{63}$$

where $G_{\mu\nu}^{0\prime,a}$ is the gluonic field strength tensor. The renormalization matrix for the coefficient functions can be expressed in terms of the well-known renormalization constants of QCD and may be found in the papers cited above.

To evaluate the ingredients entering the calculation of $\Gamma(H\to\text{hadrons})$ at three-loop level, namely C_i^0 and the imaginary part of the correlators $\langle\mathcal{O}_i'\mathcal{O}_j'\rangle$, a large amount of diagrams must be calculated. Thus it is necessary to use both a generator for the Feynman diagrams and an efficient interface transforming the output to a format readable by the integration packages. In the matter at hand one is either confronted with propagator-type diagrams or with tadpole integrals where the scales are given by M_H and M_t, respectively. Therefore the package **GEFICOM** in association with **QGRAF** (for the generation) and **MATAD/MINCER** (for the computation) is an ideal candidate for the evaluation of the Feynman diagrams contributing to $\Gamma(H\to\text{hadrons})$.

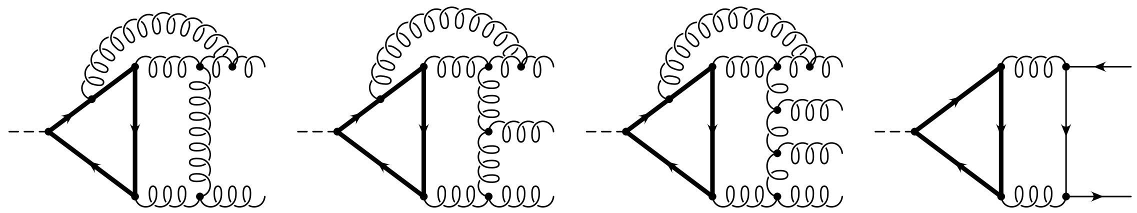


Figure 21: Typical Feynman diagrams contributing to the coefficients C_1^0 and C_2^0. Looped, bold-faced, and dashed lines represent gluons, t quarks, and H bosons, respectively.

In the next section we will give an example of the calculation of three-loop objects in the case of C_1^0 in some detail and concentrate in Section 4.4.3 on the evaluation of the massless correlators. GEFICOM was even used to compute the $\mathcal{O}(\alpha_s^3)$ contribution to $\langle \mathcal{O}_2' \mathcal{O}_2' \rangle$ which actually is a four-loop calculation. It is shown how such a calculation is nevertheless possible using the three-loop tools provided by GEFICOM. Finally, in Section 4.4.4 the physical results are briefly presented.

4.4.2 Coefficient functions

The top quark dependence of the effective Lagrangian is contained in the coefficient functions C_i^0. Thus it is not surprising that their evaluation can be reduced to massive tadpole integrals with the scale given by M_t. As an example let us consider C_1^0. There are three independent ways to obtain C_1^0, namely from the sets of ggH three-point, $gggH$ four-point, or $ggggH$ five-point diagrams. At three-loop level, these sets contain 657, 7362, and 95004 diagrams, respectively. A sample diagram of each class is displayed in Fig. 21 (together with a diagram to be evaluated for C_2^0). The first set requires an expansion up to second order in the external momenta, whereas for the second one it suffices to keep only linear terms. For the diagrams of the third class all external momenta may be set to zero from the very beginning. However, this advantage is counterbalanced by the huge number of contributing diagrams. Note that the depth of the expansion is given by the individual terms of the operator $\mathcal{O}_1$ which dictate the structure in the external momenta.

Ref. [159] deals with the ggH channel, using the package GEFICOM to evaluate the 657 diagrams. The file defining the process is rather short because only the package MATAD has to be used:

```
*** MATAD
* power 2
* gauge 1
* dala12

        list = symbolic ;
        lagfile = 'q.lag' ;
        in = h[q1p2];
        out = g[q1], g[q2];
        nloop = ;
        options =;
        true = bridge[ g,q,c, 0,0 ];
```

The second line indicates that an expansion in the external momenta up to second order should be performed, and dala12 prompts the application of projectors for the $q_1 \cdot q_2$ terms since we are dealing with on-shell gluons ($q_1^2 = q_2^2 = 0$). The file containing the propagators and vertices is essentially identical to the pure QCD case (cf. Section 3.2.3) except for the additional coupling of the Higgs boson to top quarks.

If one chooses a covariant gauge with arbitrary gauge parameter, the CPU time for the calculation is of the order of a few days. This is drastically reduced after going, for example, to Feynman gauge. Recently also the decay of a pseudo-scalar Higgs boson into gluons was considered [160]. The number of diagrams being the same as in the scalar case, there is a slight complication according to the treatment of γ^5 in this case. Therefore the 7362 diagrams of the three-gluon channel were evaluated in Feynman gauge to have a stringent check of the

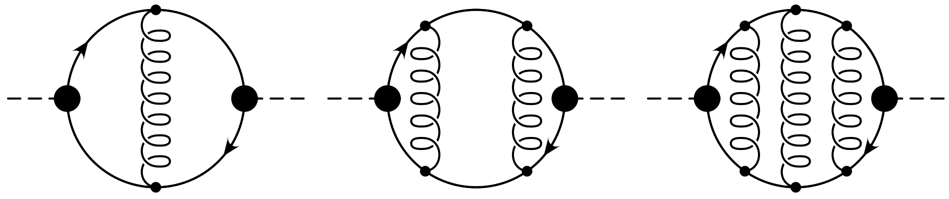


Figure 22: Typical Feynman diagrams contributing to $\langle \mathcal{O}_2' \mathcal{O}_2' \rangle$. The solid circles represent the operator $\mathcal{O}_2'$.

calculation at one's disposal. The CPU time amounts to roughly two weeks. Let us refrain from listing the analytical results and refer to [159,161] instead. Nevertheless, they are incorporated in the numerical formula given in Section 4.4.4.

4.4.3 Correlators

The computation of the one-, two- and three-loop correlators $\langle \mathcal{O}_i' \mathcal{O}_j' \rangle$ proceeds in close analogy to the coefficient functions described in the previous section. Therefore, let us in this subsection focus on the computation of the imaginary part of the four-loop contribution to $\langle \mathcal{O}_2' \mathcal{O}_2' \rangle$ which was originally performed in [162]. Some sample diagrams are shown in Fig. 22. Although at first sight this seems to require the evaluation of four-loop diagrams, the calculation can be reduced to massless propagator-type and massive tadpole integrals at three-loop level. The underlying idea is quite simple: since the imaginary part of massless propagator-type diagrams only arises from $\ln(-q^2)$ terms and the logarithms are in one-to-one correspondence to the poles in ε it suffices to compute the divergent parts of the integrals. It is a feature of dimensional regularization accompanied with MS-like schemes that the ultra-violet poles and thus the renormalization constants are polynomials in the masses and external momenta. Consequently, for logarithmically divergent diagrams the poles are independent of any mass scale. This allows us to assign additional masses to individual lines of the diagrams and to nullify the external momentum. If we take one of the lines at the left vertex to be massive and set the external momentum to zero, the four-loop integrals can be expanded in terms of three-loop propagator-type diagrams and one-loop tadpole integrals for which the technology was described in some detail above. The drawback of the described method is that artificial infra-red singularities are introduced which, of course, influence the pole structure. They have to be removed using the so-called Infra-red Rearrangement [163] which heavily relies on the R^* operation. It was developed on a diagram-by-diagram basis in [164]. In order to be able to perform the calculation automatically a global version of the R^* operation is mandatory [19,162,165].

In [162] the computation of $\mathrm{Im}\langle \mathcal{O}_2' \mathcal{O}_2' \rangle$ is presented and explicit formulae are given, demonstrating that the combination of one-, two- and three-loop massless propagator-type and massive vacuum integrals are adequate to get the $\mathcal{O}(\alpha_s^3)$ corrections. The computation was performed with the help of `GEFICOM` which had to be used at four-loop order. A proper flag in the file defining the process tells `GEFICOM` to introduce the auxiliary mass and to rename the momenta in accordance with the factorization taking place. Again we refrain from listing explicit expressions and refer to Section 4.4.4 for numerical results.

4.4.4 Results

This section summarizes the current knowledge of hadronic Higgs decay by presenting numerical results for the decay rates $\Gamma(H \to gg)$ and $\Gamma(H \to b\bar{b})$.

Note that C_1 starts at order α_s. Hence the combination $C_1^2 \mathrm{Im}\langle \mathcal{O}_1' \mathcal{O}_1' \rangle$ governing the gluonic decay rate of the Higgs boson is available up to $\mathcal{O}(\alpha_s^4)$. Normalized to the Born rate it reads in numerical form ($\mu = M_H$):

$$
\begin{aligned}
\frac{\Gamma(H \to gg)}{\Gamma^{\mathrm{Born}}(H \to gg)} &\approx 1 + 17.917\,\frac{\alpha_s^{(5)}(M_H)}{\pi} + \left(\frac{\alpha_s^{(5)}(M_H)}{\pi}\right)^2 \left(156.808 - 5.708\,\ln\frac{M_t^2}{M_H^2}\right) \\
&\approx 1 + 0.66 + 0.21\,, \qquad (64)
\end{aligned}
$$

with $\Gamma^{\mathrm{Born}}(H \to gg) = G_F M_H^3/(36\pi\sqrt{2}) \times (\alpha_s^{(5)}(M_H)/\pi)^2$. In the last line $M_t = 175$ GeV and $M_H = 100$ GeV is inserted. The analytical expressions can be found in [159]. We observe that the new $\mathcal{O}(\alpha_s^2)$ term further increases the well-known $\mathcal{O}(\alpha_s)$ enhancement [21,22] by about one third. If we assume that this trend continues to $\mathcal{O}(\alpha_s^3)$ and beyond, then Eq. (64) may already be regarded as a useful approximation to the full result. Inclusion of the new $\mathcal{O}(\alpha_s^2)$ correction leads to an increase of the Higgs-boson hadronic width by an amount of order 1%.

Concerning the decay rate into quarks we restrict ourselves to the case of bottom quarks. Inserting numerical values into the coefficient functions C_1 and C_2 and the correlators $\mathrm{Im}\langle\mathcal{O}_1'\mathcal{O}_2'\rangle$ [166] and $\mathrm{Im}\langle\mathcal{O}_2'\mathcal{O}_2'\rangle$ [162, 166] leads to:

$$
\begin{aligned}
\Gamma(H \to b\bar{b}) &= A_{b\bar{b}} \Bigg\{ 1 + 5.667\, a_H^{(5)} + 29.147\, \left(a_H^{(5)}\right)^2 + 41.758 \left(a_H^{(5)}\right)^3 \\
&\quad + \left(a_H^{(5)}\right)^2 [3.111 - 0.667\, L_t] + \left(a_H^{(5)}\right)^3 \left[50.474 - 8.167\, L_t - 1.278\, L_t^2\right] \Bigg\},
\end{aligned}
\tag{65}
$$

with $A_{b\bar{b}} = 3G_F M_H m_b^2/(4\pi\sqrt{2})$, $L_t = \ln M_H^2/M_t^2$ and $a_H^{(5)} = \alpha_s^{(5)}(M_H)/\pi$. In Eq. (65) electromagnetic [167] and electroweak [168, 158] corrections have been neglected. Also mass effects [35] and second order QCD corrections which are suppressed by the top quark mass [34] are not displayed. One observes from Eq. (65) that the top-induced corrections at $\mathcal{O}(\alpha_s^3)$ (second line) are of the same order of magnitude as the "massless" corrections (first line).

4.5 Strongly interacting vector bosons

As an application for the package `CompHEP` let us consider a scenario where the violation of unitarity in elastic scattering of massive vector bosons at high energies is avoided by letting them become strongly interacting. This is an alternative approach to the Standard Model solution of introducing a scalar Higgs boson. In [27] a new fundamental strong interaction among the vector bosons is assumed that occurs at a scale of order 1 TeV. The model is described by a Lagrangian with global chiral symmetry which is spontaneously broken in order to generate the masses of the vector bosons. The corresponding interaction terms have been worked out and implemented in `CompHEP`. Although this would have been much simpler in unitary gauge due to the absence of ghost particles, the authors of [27] decided to use 't Hooft-Feynman gauge as then the size of the final expressions is much smaller. One possibility to test the new interaction terms are elastic and quasi-elastic $2 \to 2$ scattering experiments with $W^\pm$ and Z bosons which requires colliders with a center-of-mass energy in the TeV range. Indeed, a future e^+e^- linear collider is expected to operate at energies of 1.5 to 2 TeV in a second stage [169].

The following processes are considered in [27]:

$$
\begin{aligned}
e^+e^- &\to \nu_e\bar{\nu}_e W^+W^- &&: \quad W^+W^- \to W^+W^- \\
e^+e^- &\to \nu_e\bar{\nu}_e ZZ &&: \quad ZZ \to ZZ \\
e^-e^- &\to \nu_e\bar{\nu}_e W^-W^- &&: \quad W^-W^- \to W^-W^- \\
e^+e^- &\to e^-\bar{\nu}_e W^+Z &&: \quad W^+Z \to W^+Z \\
e^+e^- &\to e^+e^- ZZ &&: \quad ZZ \to ZZ
\end{aligned}
\tag{66}
$$

where after the colon the subprocess characterizing the vector boson scattering is given. However, not only those diagrams that contain the desired subprocess contribute. Instead, also background diagrams must be computed where, for example, the final state vector bosons are radiated off the fermion lines or one of the massive vector bosons is replaced by a photon. Furthermore, diagrams are involved where the neutrinos result from the decay of a Z boson which is in turn generated from a $W^\pm$ pair. In Fig. 23 three sample diagrams are listed.

It is straightforward to go through the menus of `CompHEP`, to enter the different processes, and to produce `FORTRAN` output for the squared matrix elements. This is then used in the numerical part of `CompHEP` to compute the cross sections. The first run contains no cuts, but computes all matrix elements corresponding to the generated set of four-momenta. In subsequent runs, cuts are applied that reduce the background and isolate the signal. The data computed in the first run are taken over so that the CPU time is significantly smaller in this second step, as already mentioned in Section 3.2.5. For example, the cuts include a lower limit on the

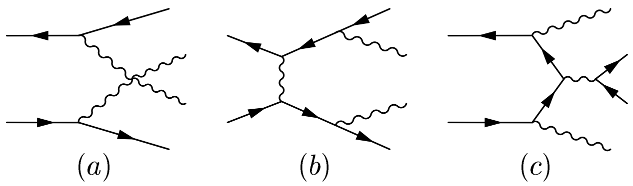


Figure 23: Sample diagrams contributing to processes specified in Eq. (66). Diagram (a) is part of the signal of the vector boson scattering process whereas (b) and (c) belong to the background.

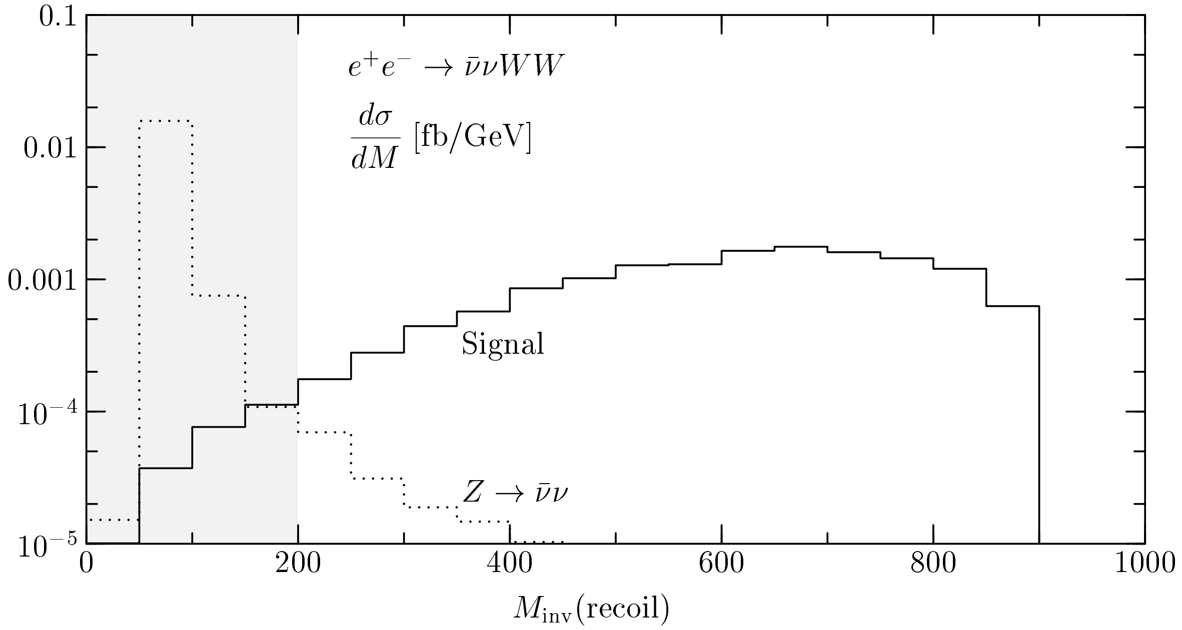


Figure 24: Distribution of the invariant mass of the produced vector bosons (here generically called W) in the process $e^+e^- \to W^+W^-\bar{\nu}_e\nu_e$ (signal). The cut (shaded area) removes events in which the neutrinos are generated through Z decays.

invariant mass of the $\nu_e\bar{\nu}_e$ system, or the selection of central events in combination with cuts on the transverse momentum of the $W^\pm$, respectively, the Z boson. Fig. 24 shows how the number of events containing neutrinos from Z boson decay (c.f. Fig. 23(c)) is reduced. Figures of this kind can easily be produced by using the data files generated by `CompHEP`.

This is not the place to go into details concerning the physical consequences for which we shall refer to [27]. The key point, however, is that the couplings of the new interaction terms can be related to the $2 \to 2$ vector boson scattering amplitudes. Thus, the investigation of the corresponding total cross sections as well as several other distributions like the one for invariant masses or transverse momenta provide important tests on the mechanism responsible for restoring unitarity at high energies.

4.6 Higgs mass dependence of two-loop corrections to Δr

As was already pointed out in the introduction, the dependence of radiative corrections on yet undiscovered particles may be exploited in order to gain information on the masses of these particles. This has been performed very successfully for the top quark and nowadays, as M_t is measured with reasonable accuracy, the same strategy is used to pin down the mass of the Higgs boson. An important quantity in this respect is Δr, which comprises the radiative corrections to muon decay. In this way it relates four fundamental parameters of the Standard Model to each other — the electromagnetic coupling α, the Fermi constant G_F and the masses of the W and Z boson:

$$M_W^2\left(1-\frac{M_W^2}{M_Z^2}\right) = \frac{\pi\alpha}{\sqrt{2}G_F}(1+\Delta r)\,. \tag{67}$$

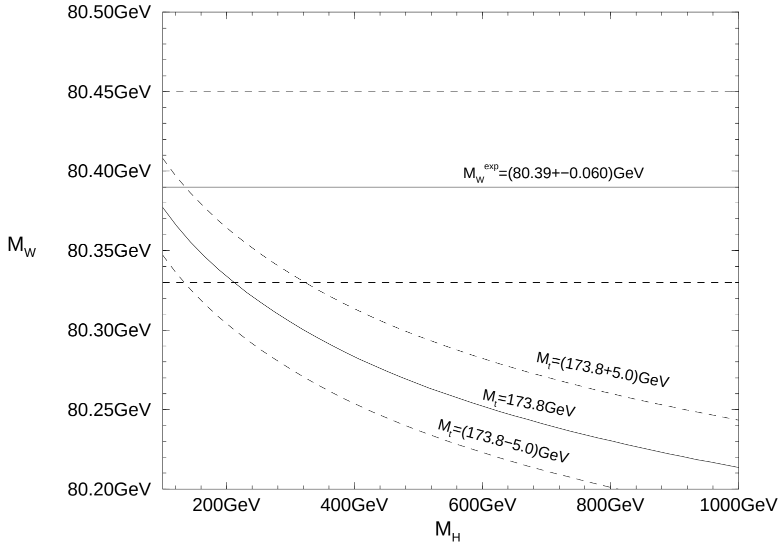


Figure 25: Limits on the Higgs mass depending on the values of the top and W mass [174]. The value of $\Delta r(M_H = 65\text{ GeV})$ was taken from [173].

G_F, α and M_Z are very well measured quantities, and the experimental accuracy of M_W is expected to be considerably improved within the near future by the current and upcoming measurements at LEP2 and TEVATRON.

Therefore, precise knowledge of the M_H-dependence of Δr is highly desirable. The one-loop corrections to Δr are known analytically since long [170], and two-loop corrections up to next-to-leading order in $1/M_t$ were considered in [171–173]. It was the concern of [26] to precisely account for the M_H dependence of the two-loop diagrams containing the top quark and the Higgs boson. The idea was to get a result valid for arbitrary values of M_t and M_H without performing an expansion in these masses. This obviously leads to two-loop multi-scale diagrams, their number being of the order of one hundred. Not only the large number of diagrams but also the complicated tensor structure and the evaluation of the scalar two-loop integrals makes the use of highly automated software indispensable. A further technical problem arises from the renormalization which has to be known up to two loops in the electroweak sector of the Standard Model. In order to investigate the Higgs-mass dependece of Δr, in [26] the subtracted quantity

$$\Delta r(M_H) - \Delta r(M_H = 65\text{ GeV}) \tag{68}$$

was considered which describes the change in the prediction for Δr when M_H is varied. The value for $\Delta r(M_H = 65\text{ GeV})$ was taken from [173].

It turned out that the computation of the diagrams contributing to the considered corrections to Δr could be reduced to two-loop tadpole integrals and two-point functions. This apparently is a task for the `Mathematica` programs `FeynArts` and `TwoCalc`. The various contributions to Δr together with the corresponding counter-terms were generated by means of `FeynArts`. Contraction of Lorentz indices, evaluation of the Dirac algebra and the two-loop tensor reduction was performed by `TwoCalc`. The resulting expressions were split into finite and divergent pieces which explicitly demonstrates the cancellation of poles. The scalar one- and two-loop integrals were reduced to one-dimensional integrals allowing a fast numerical evaluation to high accuracy with the help of special `C` routines which themselves were fully integrated into the `Mathematica` environment [102]. CPU time for this calculation added up to the order of a few days.

Instead of displaying the full result, we only note that the resulting prediction for the W mass agrees within a few MeV with the one of the heavy-top expansion [173], which is far below the experimental precision at the moment. (For a more detailed discussion see [174] and [175].) It is remarkable that the top-induced corrections turn out to be maximal at the physical value of the top quark mass within a range of ± 50 GeV around this physical value. As shown in Fig. 25, the interrelation between M_W and M_t favours relatively small values for M_H, consistent with other indirect determinations using the radiative shift in the weak mixing angle or the leptonic decay rate of the Z boson. Both of these quantities recently have been calculated with similar techniques as described in this section [176].

Acknowledgements

First of all we would like to thank J.H. Kühn for his kind advice and numerous valuable suggestions and discussions. We are grateful to the following colleagues for carefully reading the manuscript (or parts of it) and their expert advice on various subjects: P. Baikov, K.G. Chetyrkin, P. Gambino, T. Hahn, W. Kilian, B.A. Kniehl, K. Melnikov, A. Pukhov, A. Rétey, T. van Ritbergen, T. Seidensticker, V.A. Smirnov, R.G. Stuart, M. Stuber, O.V. Tarasov, and G. Weiglein. This work was supported in part by *Graduiertenkolleg "Elementarteilchenphysik an Beschleunigern"* and *DFG-Forschergruppe "Quantenfeldtheorie, Computeralgebra und Monte-Carlo-Simulationen"* at the University of Karlsruhe, and the *Schweizer Nationalfond.*

References

[1] D. Karlen, talk presented at ICHEP '98, Vancouver, July 1998.

[2] W. Hollik, talk presented at ICHEP '98, Vancouver, July 1998.

[3] S.W. Herb *et al.*, *Phys. Rev. Lett.* **39** (1977) 252.

[4] F. Abe *et al.* (CDF Collaboration), *Phys. Rev. Lett.* **74** (1995) 2626;
S. Abachi *et al.* (D0 Collaboration), *Phys. Rev. Lett.* **74** (1995) 2632.

[5] R. Partridge, talk presented at ICHEP '98, Vancouver, July 1998.

[6] M. Veltman, *Act. Phys. Pol.* **B 8** (1977) 475; *Phys. Lett.* **B 70** (1977) 253.

[7] F. Teubert, Proceedings of the *IVth Int. Symp. on Radiative Corrections*, Barcelona, Sept. 8-12, 1998; hep-ph/9811414.

[8] P. McNamara, talk presented at ICHEP '98, Vancouver, July 1998.

[9] M. Veltman, *SCHOONSCHIP*, CERN preprint, 1967;
H. Strubbe, *Comp. Phys. Commun.* **8** (1974) 1.

[10] MACSYMA Reference manual version 9, MIT *Lab. Comp. Sci.* (1977).

[11] A.C. Hearn, *REDUCE user's manual Version 3.5* (RAND publication, Santa Monica 1993).

[12] S. Wolfram, *Mathematica — a system for doing mathematics by computer* (Addison-Wesley, New York, 1988).

[13] J.A.M. Vermaseren, *Symbolic Manipulation with FORM*, (Computer Algebra Netherlands, Amsterdam, 1991).

[14] G. Passarino and M. Veltman, *Nucl. Phys.* **B 160** (1979) 151.

[15] F.V. Tkachov, *Phys. Lett.* **B 100** (1981) 65;
K.G. Chetyrkin and F.V. Tkachov, *Nucl. Phys.* **B 192** (1981) 159.

[16] see, e.g.: A. Czarnecki, U. Kilian, and D. Kreimer, *Nucl. Phys.* **B 433** (1995) 259;
J. Fleischer *et al.*, *Eur. Phys. J.* **C 2** (1998) 747;
D.E. Soper, *Phys. Rev. Lett.* **81** (1998) 2638;
S. Groote, J.G. Körner, and A.A. Pivovarov, Rep. No. MZ-TH-98-09 (1998); hep-ph/9806402;
R. Kreckel, D. Kreimer, and K. Schilcher, *Eur. Phys. J.* **C 6** (1999) 693.

[17] K.G. Chetyrkin, A.L. Kataev, and F.V. Tkachov, *Phys. Lett.* **B 85** (1979) 277;
M. Dine and J. Sapirstein, *Phys. Rev. Lett.* **43** (1979) 668;
W. Celmaster and R.J. Gonsalves, *Phys. Rev. Lett.* **44** (1980) 560.

[18] L.R. Surguladze and M.A. Samuel, *Phys. Rev. Lett.* **66** (1991) 560; (E) ibid., 2416;
S.G. Gorishny, A.L. Kataev, and S.A. Larin, *Phys. Lett.* **B 259** (1991) 144.

[19] K.G. Chetyrkin, *Phys. Lett.* **B 391** (1997) 402.

[20] K.G. Chetyrkin, J.H. Kühn, and M. Steinhauser, *Phys. Lett.* **B 371** (1996) 93; *Nucl. Phys.* **B 482** (1996) 213; *Nucl. Phys.* **B 505** (1997) 40.

[21] T. Inami, T. Kubota, and Y. Okada, *Z. Phys.* **C 18** (1983) 69.

[22] A. Djouadi, M. Spira, and P.M. Zerwas, *Phys. Lett.* **B 264** (1991) 440.

[23] T. van Ritbergen, J.A.M. Vermaseren, and S.A. Larin, *Phys. Lett.* **B 400** (1997) 379.

[24] K.G. Chetyrkin, *Phys. Lett.* **B 404** (1997) 161;
J.A.M. Vermaseren, S.A. Larin, and T. van Ritbergen, *Phys. Lett.* **B 405** (1997) 327.

[25] R. Harlander, T. Seidensticker, and M. Steinhauser, *Phys. Lett.* **B 426** (1998) 125.

[26] S. Bauberger and G. Weiglein, *Phys. Lett.* **B 419** (1998) 333.

[27] E. Boos, H.-J. He, W. Kilian, A. Pukhov, C.-P. Yuan, and P Zerwas, *Phys. Rev.* **D 57** (1998) 1553.

[28] L. Avdeev, J. Fleischer, S. Mikhailov, and O. Tarasov, *Phys. Lett.* **B 336** (1994) 560; (E) ibid. **B 349** (1995) 597.

[29] K.G. Chetyrkin, J.H. Kühn, and M. Steinhauser, *Phys. Lett.* **B 351** (1995) 331.

[30] K.G. Chetyrkin, J.H. Kühn, and M. Steinhauser, *Phys. Rev. Lett.* **75** (1995) 3394.

[31] S.A. Larin, F.V. Tkachov, and J.A.M. Vermaseren, *Phys. Lett.* **B 272** (1991) 121.

[32] S.A. Larin, T. van Ritbergen, and J.A.M. Vermaseren, *Nucl. Phys.* **B 427** (1994) 41.

[33] S.A. Larin, P. Nogueira, R. van Ritbergen, and J.A.M. Vermaseren, *Nucl. Phys.* **B 492** (1997) 338.

[34] S.A. Larin, T. van Ritbergen, and J.A.M. Vermaseren, *Phys. Lett.* **B 362** (1995) 134;
K.G. Chetyrkin and A. Kwiatkowski, *Nucl. Phys.* **B 461** (1996) 3.

[35] R. Harlander and M. Steinhauser, *Phys. Rev.* **D 56** (1997) 3980.

[36] R. Harlander and M. Steinhauser, *Eur. Phys. J.* **C 2** (1998) 151.

[37] A. Denner, S. Dittmaier, and T. Hahn, *Phys. Rev.* **D 56** (1997) 117;
A. Denner and T. Hahn, *Nucl. Phys.* **B 525** (1998) 27.

[38] A. Czarnecki, B. Krause, and W.J. Marciano, *Phys. Rev. Lett.* **76** (1996) 3267.

[39] A. Czarnecki and B. Krause, *Phys. Rev. Lett.* **78** (1997) 4339.

[40] A. Frink, B.A. Kniehl, D. Kreimer, and K. Riesselmann, *Phys. Rev.* **D 54** (1996) 4548.

[41] A. Czarnecki and K. Melnikov, *Phys. Rev. Lett.* **78** (1997) 3630.

[42] A. Czarnecki and K. Melnikov, *Phys. Rev.* **D 59** (1999) 014036.

[43] A. Czarnecki and K. Melnikov, *Phys. Rev. Lett.* **80** (1998) 2531.

[44] M. Beneke and V.A. Smirnov, *Nucl. Phys.* **B 522** (1998) 321.

[45] A.H. Hoang and T. Teubner, *Phys. Rev.* **D 58** (1998) 114023.

[46] M. Beneke, A. Signer, and V.A. Smirnov, *Phys. Rev. Lett.* **80** (1998) 2535.

[47] E.E. Boos, M.N. Dubinin, V.A. Ilin, A.E. Pukhov, V.I. Savrin, *CompHEP: specialized package for automatic calculations of elementary particle decays and collisions*, Rep. No. SNUTP-94-116 (1994); hep-ph/9503280;
P.A. Baikov *et al.*, *10th International Workshop on High-energy Physics and Quantum Field Theory (NPI MSU 95), Zvenigorod, Russia, 19-26 Sep 1995*; hep-ph/9701412.

[48] G. 't Hooft and M. Veltman, *Nucl. Phys.* **B 44** (1972) 189.

[49] G. 't Hooft, *Nucl. Phys.* **B 61** (1973) 455.

[50] W.A. Bardeen, A.J. Buras, D.W. Duke, and T. Muta, *Phys. Rev.* **D 18** (1978) 3998.

[51] P. Breitenlohner and D. Maison, *Comm. Math. Phys.* **52** (1977) 11.

[52] D.J. Broadhurst, *Z. Phys.* **C 54** (1992) 599.

[53] L.V. Avdeev, *Comp. Phys. Commun.* **98** (1996) 15.

[54] D.J. Broadhurst, Rep. No. OUT-4102-72, hep-th/9803091.

[55] N. Gray, D.J. Broadhurst, W. Grafe, and K. Schilcher, *Z. Phys.* **C 48** (1990) 673.

[56] D.J. Broadhurst, N. Gray, and K. Schilcher, *Z. Phys.* **C 52** (1991) 111.

[57] S. Laporta and E. Remiddi, *Phys. Lett.* **B 379** (1996) 283

[58] T. van Ritbergen and R.G. Stuart, *Phys. Rev. Lett.* **82** (1999) 488.

[59] P.A. Baikov, *Phys. Lett.* **B 385** (1996) 404; *Nucl. Inst. Meth.* **A 389** (1997) 347.

[60] P.A. Baikov and M. Steinhauser, hep-ph/9802429; *Comp. Phys. Commun.* (in press).

[61] G. Weiglein, R. Scharf, and M. Böhm, *Nucl. Phys.* **B 416** (1994) 606.

[62] A. Denner, *Fortschr. Phys.* **41** (1993) 307.

[63] R.G. Stuart, *Comp. Phys. Commun.* **48** (1988) 367;
R.G. Stuart and A. Góngora-T, *Comp. Phys. Commun.* **56** (1990) 337;
R.G. Stuart, *Comp. Phys. Commun.* **85** (1995) 267; (E) ibid. **88** (1995) 347.

[64] G.J. van Oldenborgh and J.A.M. Vermaseren, *Z. Phys.* **C 46** (1990) 425.

[65] G. Devaraj and R.G. Stuart, *Nucl. Phys.* **B 519** (1998) 483.

[66] A.I. Davydychev, *Phys. Lett.* **B 263** (1991) 107.

[67] O.V. Tarasov, *Phys. Rev.* **D 54** (1996) 6479.

[68] O.V. Tarasov, *Nucl. Phys.* **B 502** (1997) 455.

[69] J. Fleischer, F. Jegerlehner, O.V. Tarasov, and O.L. Veretin, *Nucl. Phys.* **B 539** (1999) 671.

[70] O.V. Tarasov, *Act. Phys. Pol.* **B 29** (1998) 2655.

[71] K.G. Chetyrkin, S.G. Gorishny, and F.V. Tkachov, *Phys. Lett.* **B 119** (1982) 407.

[72] F.V. Tkachov, *Phys. Lett.* **B 125** (1983) 85.

[73] D.J. Broadhurst and S.C. Generalis, Rep. No. OUT-4102-12 (1984).

[74] K.G. Chetyrkin and V.P. Spiridonov, *Yad. Fiz.* **47** (1988) 818.

[75] G.B. Pivovarov and F.V. Tkachov, Rep. No. INR P-0370 (Moscow, 1984);
F.V. Tkachov, *Int. Journ. Mod. Phys.* **A 8** (1993) 2047;
G.B. Pivovarov and F.V. Tkachov, *Int. Journ. Mod. Phys.* **A 8** (1993) 2241.

[76] S.G. Gorishny, S.A. Larin, and F.V. Tkachov, *Phys. Lett.* **B 124** (1983) 217;
S.G. Gorishny and S.A. Larin, *Nucl. Phys.* **B 283** (1987) 452;
S.G. Gorishny, *Nucl. Phys.* **B 319** (1989) 633.

[77] K.G. Chetyrkin and V.A. Smirnov, Rep. No. INR P-518 (Moscow, 1987);
K.G. Chetyrkin, *Teor. Mat. Fiz.* **75** (1988) 26;
K.G. Chetyrkin, Rep. No. MPI-PAE/PhT 13/91 (Munich, 1991);
V.A. Smirnov, *Comm. Math. Phys.* **134** (1990) 109.

[78] V.A. Smirnov, *Renormalization and Asymptotic Expansion* (Birkhäuser, Basel, 1991).

[79] V.A. Smirnov, *Mod. Phys. Lett.* **A 10** (1995) 1485.

[80] A.I. Davydychev and J.B. Tausk, *Nucl. Phys.* **B 397** (1993) 123;
A.I. Davydychev, V.A. Smirnov, and J.B. Tausk, *Nucl. Phys.* **B 410** (1993) 325;
F.A. Berends, A.I. Davydychev, V.A. Smirnov, and J.B. Tausk, *Nucl. Phys.* **B 439** (1995) 536.

[81] V.A. Smirnov, *Phys. Lett.* **B 394** (1997) 205.

[82] A. Czarnecki and V.A. Smirnov, *Phys. Lett.* **B 394** (1997) 211.

[83] L.V. Avdeev and M.Yu. Kalmykov, *Nucl. Phys.* **B 502** (1997) 419.

[84] V.A. Smirnov, *Phys. Lett.* **B 404** (1997) 101.

[85] K.G. Chetyrkin and J.H. Kühn, *Nucl. Phys.* **B 432** (1994) 337.

[86] K.G. Chetyrkin, *Phys. Lett.* **B 307** (1993) 169.

[87] A.H. Hoang, M. Jeżabek, J.H. Kühn, and T. Teubner, *Phys. Lett.* **B 338** (1994) 330.

[88] T. Teubner, Ph. D. thesis, University of Karlsruhe (Shaker Verlag, Aachen, 1996).

[89] M. Steinhauser, Ph. D. thesis, University of Karlsruhe (Shaker Verlag, Aachen, 1996).

[90] Th. Seidensticker, Diploma thesis (University of Karlsruhe, 1998), unpublished.

[91] J.D. Bjorken and M.C. Chen, *Phys. Rev.* **154** (1967) 1335;
O. Reading Henry, *Phys. Rev.* **154** (1967) 1534.

[92] P. De Causmaecker, R. Gastmans, W. Troosts, and T.T. Wu, *Phys. Lett.* **B 105** (1981) 215; *Nucl. Phys.* **B 206** (1982) 53;
F.A. Berends, R. Kleiss, P. De Causmaecker, R. Gastmans, W. Troosts, and T.T. Wu, *Nucl. Phys.* **B 206** (1982) 61; *Nucl. Phys.* **B 239** (1984) 382 and 395; *Nucl. Phys.* **B 264** (1986) 243 and 265.

[93] F.A. Berends, P.H. Daverveldt, and R. Kleiss, *Nucl. Phys.* **B 253** (1985) 441;
R. Kleiss and W.H. Stirling, *Nucl. Phys.* **B 262** (1985) 235;
C. Mana and M. Martinez, *Nucl. Phys.* **B 287** (1987) 601.

[94] A. Ballestrero and E. Maina, *Phys. Lett.* **B 350** (1995) 225.

[95] J. Küblbeck, M. Böhm, and A. Denner, *Comp. Phys. Commun.* **60** (1990) 165;
H. Eck and J. Küblbeck, *Guide to FeynArts 1.0* (Würzburg, 1992);
H. Eck, *Guide to FeynArts 2.0* (Würzburg, 1995);
T. Hahn, *FeynArts 2.2 User's Guide* (Karlsruhe, 1998).

[96] P. Nogueira, *J. Comp. Phys.* **105** (1993) 279.

[97] S.G. Gorishny, S.A. Larin, and F.V. Tkachov, Rep. No. INR P-0330 (Moscow, 1984);
S.G. Gorishny, S.A. Larin, L.R. Surguladze and F.V. Tkachov, *Comp. Phys. Commun.* **55** (1989) 381.

[98] S.A. Larin, F.V. Tkachov, and J.A.M. Vermaseren, Rep. No. NIKHEF-H/91-18 (Amsterdam, 1991).

[99] J. Fleischer and O.V. Tarasov, *Comp. Phys. Commun.* **71** (1992) 193.

[100] R. Mertig, M. Böhm, and A. Denner, *Comp. Phys. Commun.* **64** (1991) 345.

[101] T. Hahn and M. Perez-Victoria, Rep. No. KA-TP-7-1998, hep-ph/9807565 (Karlsruhe, 1998);
T. Hahn, *FormCalc and LoopTools 1.2 User's Guide* (Karlsruhe, 1998).

[102] S. Bauberger, F.A. Berends, M. Böhm, and M. Buza, *Nucl. Phys.* **B 434** (1995) 383.

[103] G.J. van Oldenborgh, *FF — a package to evaluate one-loop Feynman diagrams*, Rep. No. NIKHEF-H/90-15 (Amsterdam, 1990).

[104] G. Degrassi, P. Gambino, and S. Fanchiotti, unpublished.

[105] L. Brücher, J. Franzkowski, and D. Kreimer, Rep. No. MZ-TH/97-35, hep-ph/9710484 (Mainz, 1997).

[106] A. Czarnecki, U. Kilian, and D. Kreimer, *Nucl. Phys.* **B 433** (1995) 259.

[107] B. W. Char *et al.*, *Maple V Language Reference Manual* (Springer, 1991).

[108] G.P. Lepage, *J. Comp. Phys.* **27** (1978) 192.

[109] R. Kreckel, *Comp. Phys. Commun.* **106** (1997) 258.

[110] R. Harlander, Ph. D. thesis, University of Karlsruhe (Shaker Verlag, Aachen, 1998).

[111] K.G. Chetyrkin and M. Steinhauser, unpublished.

[112] T. van Ritbergen, J.A.M. Vermaseren, S.A. Larin, and P. Nogueira, *Int. J. Mod. Phys.* **C 6** (1995) 513; J.A.M. Vermaseren, *Nucl. Inst. Meth.* **A 389** (1997) 350.

[113] T. van Ritbergen, unpublished.

[114] T. van Ritbergen, A.N. Schellekens, and J.A.M. Vermaseren, *Int. J. Mod. Phys.* **A 14** (1999) 41.

[115] T. Kaneko *et al.*, in: *New Computing Techniques in Physics Research II*, ed. D. Perret-Gallix (World Scientific, Singapore, 1992) 659;
T. Ishikawa *et al.*, *GRACE Manual*, KEK Rep. No. 92-19 (1993).

[116] M. Jimbo, Minami-Tateya collaboration, *JLC Workshop* 1995:0098-107.

[117] H. Tanaka, *Comp. Phys. Commun.* **58** (1990) 153;
H. Tanaka *et al.*, *Comp. Phys. Commun.* **64** (1991) 149.

[118] S. Kawabata, *Comp. Phys. Commun.* **41** (1986) 127; ibid. **88** (1995) 309.

[119] L.V. Avdeev, J. Fleischer, M.Yu. Kalmykov, and M.N. Tentyukov, *Nucl. Inst. Meth.* **A 389** (1997) 343; *Comp. Phys. Commun.* **107** (1997) 155.

[120] L.V. Avdeev, J. Fleischer, M.Yu. Kalmykov, and M.N. Tentyukov, *Comp. Phys. Commun.* **107** (1997) 155;
J. Fleischer, M. Tentyukov, and O.L. Veretin, *Act. Phys. Pol.* **B 28** (1997) 2333.

[121] E. Murayama, I. Watanabe, and K. Hagiwara, KEK Report 91-11, Jan. 1992.

[122] L.R. Surguladze, Rep. No. FERMILAB-PUB-92-191-T (1992);
L.R. Surguladze, *Int. J. Mod. Phys.* **C 5** (1994) 1089.

[123] A.V. Semenov, *Nucl. Inst. Meth.* **A 389** (1997) 293;
A.V. Semenov, *LanHEP – a package for automatic generation of Feynman rules from the Lagrangian*, Rep. No. INP MSU 98-2/503.

[124] L.R. Surguladze and F.V. Tkachov, *Comp. Phys. Commun.* **55** (1989) 205.

[125] T. Stelzer and W.F. Long, *Comp. Phys. Commun.* **81** (1994) 357.

[126] L. Brücher, J. Franzkowski, and D. Kreimer, *Comp. Phys. Commun.* **85** (1995) 153;
Comp. Phys. Commun. **107** (1997) 281.

[127] A. Ballestrero, Torino preprint, in preparation.

[128] D. Broadhurst, unpublished.

[129] E. Accomando, A. Ballestrero, and M. Pizzio, *Nucl. Phys.* **B 512** (1998) 19.

[130] D. Broadhurst, unpublished.

[131] R. Mertig and R. Scharf, *Comp. Phys. Commun.* **111** (1998) 265.

[132] M. Jamin and M.E. Lautenbacher, *Comp. Phys. Commun.* **74** (1993) 265.

[133] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *"Numerical Recipes in Fortran"*, Cambridge University Press (Cambridge, 1992).

[134] E. Accomando and A. Ballestrero, *Comp. Phys. Commun.* **99** (1997) 270.

[135] A.E. Pukhov, private communication.

[136] G. Källen and A. Sabry, *K. Dan. Videnk. Selsk. Mat.-Fys. Medd.* **29** (1955) No. 17;
see also: J. Schwinger, *Particles, Sources and Fields*, Vol. II, (Addison-Wesley, New York, 1973).

[137] R. Barbieri and E. Remiddi, *Nuovo Cim.* **13A** (1973) 99; (E) ibid. **A 13** (1973) 99.

[138] B.A. Kniehl, *Nucl. Phys.* **B 347** (1990) 86.

[139] D.J. Broadhurst, J. Fleischer, and O.V. Tarasov, *Z. Phys.* **C 60** (1993) 287.

[140] A.H. Hoang, J.H. Kühn, and T. Teubner, *Nucl. Phys.* **B 452** (1995) 173.

[141] A.H. Hoang, Ph. D. thesis, University of Karlsruhe (Shaker Verlag, Aachen, 1996).

[142] K.G. Chetyrkin, R. Harlander, and M. Steinhauser, *Phys. Rev.* **D 58** (1998) 014012.

[143] K.G. Chetyrkin, J.H. Kühn, and A. Kwiatkowski, *Phys. Reports* **277** (1996) 189.

[144] K.G. Chetyrkin, A.H. Hoang, J.H. Kühn, M. Steinhauser, and T. Teubner, *Eur. Phys. J.* **C 2** (1998) 137.

[145] P.A. Baikov and D.J. Broadhurst, *4th International Workshop on Software Engineering and Artificial Intelligence for High Energy and Nuclear Physics (AIHENP95)*, Pisa, Italy, 3-8 April 1995. Published in Pisa AIHENP (1995) 167.

[146] K.G. Chetyrkin, R. Harlander, J.H. Kühn, and M. Steinhauser, *Nucl. Inst. Meth.* **A 389** (1997) 354.

[147] K.G. Chetyrkin, R. Harlander, J.H. Kühn, and M. Steinhauser, *Nucl. Phys.* **B 503** (1997) 339.

[148] A. Akhundov, D. Bardin, and T. Riemann, *Nucl. Phys.* **B 276** (1986) 1;
J. Bernabeu, A. Pich, and A. Santamaria, *Phys. Lett.* **B 200** (1988) 569;
B.W. Lynn and R.G. Stuart, *Phys. Lett.* **B 252** (1990) 676.

[149] W. Beenakker and W. Hollik, *Z. Phys.* **C 40** (1988) 141.

[150] A. Czarnecki and J.H. Kühn, *Phys. Rev. Lett.* **77** (1996) 3955; (E) ibid. **80** (1998) 893.

[151] J. Fleischer, F. Jegerlehner, P. Rączka and O.V. Tarasov, *Phys. Lett.* **B 293** (1992) 437;
G. Buchalla and A. Buras, *Nucl. Phys.* **B 398** (1993) 285;
G. Degrassi, *Nucl. Phys.* **B 407** (1993) 271.

[152] K.G. Chetyrkin, A. Kwiatkowski and M. Steinhauser, *Mod. Phys. Lett.* **A 8** (1993) 2785.

[153] A. Kwiatkowski and M. Steinhauser, *Phys. Lett.* **B 344** (1995) 359.

[154] S. Peris and A. Santamaria, *Nucl. Phys.* **B 445** (1995) 252.

[155] D. Bardin *et al.*, *Reports of the working group on precision calculations for the Z resonance*, D. Bardin, W. Hollik, G. Passarino, (eds.), CERN 95-03, 31 Mar 1995.

[156] H. Kluberg-Stern and J.B. Zuber, *Phys. Rev.* **D 12** (1975) 467;
N.K. Nielsen, *Nucl. Phys.* **B 97** (1975) 527; *Nucl. Phys.* **B 120** (1977) 212.

[157] V.P. Spiridonov, Rep. No. INR P–0378 (Moscow, 1984).

[158] K.G. Chetyrkin, B.A. Kniehl, and M. Steinhauser, *Phys. Rev. Lett.* **78** (1997) 594; *Nucl. Phys.* **B 490** (1997) 19.

[159] K.G. Chetyrkin, B.A. Kniehl, and M. Steinhauser, *Phys. Rev. Lett.* **79** (1997) 353.

[160] K.G. Chetyrkin, B.A. Kniehl, M. Steinhauser, and W.A. Bardeen, *Nucl. Phys.* **B 535** (1998) 3.

[161] K.G. Chetyrkin, B.A. Kniehl, and M. Steinhauser, *Nucl. Phys.* **B 510** (1998) 61.

[162] K.G. Chetyrkin, *Phys. Lett.* **B 390** (1997) 309.

[163] A.A. Vladimirov, *Teor. Mat. Fiz.* **43** (1980) 210;
K.G. Chetyrkin, A.L. Kataev, and F.V. Tkachov, *Nucl. Phys.* **B 174** (1980) 345.

[164] K.G. Chetyrkin and F.V. Tkachov, *Phys. Lett.* **B 114** (1982) 340;
K.G. Chetyrkin and V.A. Smirnov, *Phys. Lett.* **B 144** (1984) 419.

[165] K.G. Chetyrkin, *Act. Phys. Pol.* **B 28** (1997) 725.

[166] K.G. Chetyrkin and M. Steinhauser, *Phys. Lett.* **B 408** (1997) 320.

[167] A.L. Kataev and V.T. Kim, Rep. No. ENSLAPP-A-407-92, hep-ph/9304282 (1993);
A.L. Kataev, *JETP Lett.* **66** (1997) 327.

[168] B.A. Kniehl and M. Steinhauser, *Nucl. Phys.* **B 454** (1995) 485; *Phys. Lett.* **B 365** (1996) 297.

[169] E. Accomando *et al.* (ECFA/DESY LC Physics Working Group), *Phys. Reports* **299** (1998) 1.

[170] A. Sirlin, *Phys. Rev.* **D 22** (1980) 971;
W.J. Marciano and A. Sirlin, *Phys. Rev.* **D 22** (1980) 2695; (E) ibid. **D 31** (1985) 213.

[171] G. Degrassi, P. Gambino, and A. Vicini, *Phys. Lett.* **B 383** (1996) 219.

[172] G. Degrassi, P. Gambino, and A. Sirlin, *Phys. Lett.* **B 394** (1997) 188.

[173] G. Degrassi, P. Gambino, M. Passera, and A. Sirlin, *Phys. Lett.* **B 418** (1998) 209.

[174] G. Weiglein, Proceedings of the *IVth Int. Symp. on Radiative Corrections*, Barcelona, Sept. 8-12, 1998.

[175] P. Gambino, Proceedings of the *IVth Int. Symp. on Radiative Corrections*, Barcelona, Sept. 8-12, 1998.

[176] G. Weiglein, *Act. Phys. Pol.* **B 29** (1998) 2735.