Automated one-loop calculations: a proof of concept

A. van Hameren
The H. Niewodniczański Institute of Nuclear Physics
Polish Academy of Sciences
Radzikowskiego 152, 31-342 Cracow, Poland

C.G. Papadopoulos
Institute of Nuclear Physics, NCSR Demokritos, 15310 Athens, Greece

R. Pittau
Departamento de Física Teórica y del Cosmos
Centro Andaluz de Física de Partículas Elementales (CAFPE)
Universidad de Granada, E-18071 Granada, Spain

Abstract: An algorithm, based on the OPP reduction method, to automatically compute any one-loop amplitude, for all momentum, color and helicity configurations of the external particles, is presented. It has been implemented using the tree-order matrix element code HELAC and the OPP reduction code CutTools. As a demonstration of the potential of the current implementation, results for all sub-processes included in the 2007 Les Houches wish list for LHC, are presented.

Keywords: NLO, radiative corrections, LHC
Contents

1. Introduction 2
2. One-Loop amplitudes 4
3. Results 12
4. Summary and Outlook 16
A. OneLoop for evaluating scalar loop integrals 17
1. Introduction

The forthcoming LHC data will hopefully improve our current understanding of particle physics. The discovery of the Higgs particle will be of primary importance. Moreover, new particles and interactions are expected to be within reach. In order to be able to discover new physics, the precise description of multi-parton final states is necessary [1].

At the leading order in perturbation theory, many tools are already available that are able to simulate any scattering process involving up to several partons, among them Alpgen [2], MadEvent [3], Sherpa [4], HELAC-PHEGAS [5–8], WHIZARD [9]. These tools are highly automated and they have been widely used [10]. The advance of algorithms based on recursive equations [11–13] to calculate multi-parton scattering amplitudes as opposed to the traditional Feynman graph approach, has been proven a very important factor in order to build up fast and reliable computer codes.

At the next-to-leading order the situation is currently less advanced. At the conceptual level, one has to deal with an integration over the loop momentum which results to ultraviolet and infrared divergencies. Dimensional regularization [15, 16] is needed in order to produce meaningful results. The amplitude can be cast in the form of a linear combination of known scalar integrals [17] – boxes, triangles, bubbles and tadpoles – multiplied by coefficients that are rational functions of the external momenta and polarization vectors, plus a remainder which is also a rational function of the latter. At the practical level, one has to devise an efficient algorithm to calculate all these ingredients. Starting with the scalar integrals, the problem is considered solved: there exist several implementations, covering all cases of interest [18, 19]. We will also present such an implementation in Appendix A. As far as the full one-loop amplitude is concerned the situation is less satisfactory for the moment. On the one hand there are tools, like MCFM [20], that are able to produce results at NLO accuracy, for specific scattering processes, based on analytic calculations. On the other hand the only automatic tool available for some time now was FeynCalc [21] and FormCalc [22]. These tools rely heavily on the use of computer algebra programmes, notably Mathematica\footnote{http://www.wolfram.com/products/mathematica/index.html} and FORM [23], and are based on the traditional Passarino-Veltman [24–27] (PV) reduction of Feynman graphs, that are generated automatically (FeynArts [28] or QGRAF [29]). In order to produce numerical results, tensor coefficients functions are calculated using LoopTools [30]. For processes with two particles in the final state, their performance is very satisfactory. For the time being five-point rank four is the highest available option. It should be noticed though that there exist several important calculations, that make use of these automatic packages such as FeynArts, QGRAF and FormCalc, producing results with up to four particles in the final state [31–35], but for the moment no publicly available automatic tool exists. Recently a programme called GOLEM [36] has been presented, that it is also able to perform reduction of tensor integrals.
with up to six eternal legs. It will also provide an alternative to compute automatically one-loop amplitudes [37]. Alternatives to the PV reduction scheme relevant for our discussion have been also presented in the literature, including the van Oldenborgh-Vermaseren scheme [38] and the reduction at the integrand level technique [39,40].

In a very different line of thinking, starting from the pioneering work of Bern, Dixon, Dunbar, and Kosower [41,42], a new approach has been set forward, known under the name of unitarity approach. Unitarity has been proven very powerful in computing multi-parton amplitudes in QCD [43–45] that seemed to be impossible with the traditional Feynman graph approach. The reason is that within the unitarity approach, one-loop amplitudes may be calculated by using tree-order building blocks, that are either known analytically with very compact expressions, or can be evaluated using fast recursive equations. Nevertheless a systematic framework to develop a generic computation of any one-loop amplitude was missing, limiting the applicability of the method.

Few years ago, Britto, Cachazo and Feng [46] made a very important discovery: introducing the so-called quadruple cut of one-loop amplitudes, they were able to reproduce directly, known results regarding the box coefficients. It was still unclear though how to get in a systematic way all the coefficients of the scalar integrals. This problem has been first solved by Ossola, Papadopoulos and Pittau [47,48] (OPP), who introduced a systematic framework, in order to calculate all coefficients of the scalar integrals, as well as the rational part of the integral originating from the reduction process of a four-dimensional numerator, called $R_1$ in their approach. The part of the rational remainder that originates from the explicit dependence of the numerator function on the dimension of the loop momentum, called $R_2$, can be reproduced by counter-terms encoded in tree-like Feynman rules involving up to four fields [49]. Therefore, the OPP method provides a self-contained framework for the evaluation of the full one-loop amplitude. Ellis, Giele, Kunszt and Melnikov [50,51] used the OPP reduction approach within the so-called generalized unitarity approach [46,52] in order to get also the full rational part of the amplitude, paying the price to work with tree-amplitudes in higher dimensions.

The systematic extraction of all coefficients and of the rational term, opened the road for the construction of tools that are able to compute one-loop amplitudes with any number of particles. BlackHat [53] and Rocket [54] were the first tools to realize such a possibility: based on either on-shell recursive equations [55] or Berends-Giele ones, were able to compute multi-gluon one-loop primitive amplitudes with as many as 20 gluons in the final state. Primitive one-loop amplitudes with massless and massive quarks as well as electroweak bosons, were also added later [56]. Moreover, realistic calculations of leading color NLO corrections to $W + 3 \text{ jets}$ have been achieved recently [57,58].

In this paper, we report on the development of a new algorithm based on the tree-order amplitude computation code HELAC [5,6,8] and the OPP reduction code CutTools [59]. Within this approach, it is shown how the full one-loop amplitude can be computed. Results
for the full color and helicity summed squared matrix elements for (basically) all $2 \to 4$ (sub-)processes included in the Les Houches wish list [1], are presented.

2. One-Loop amplitudes

For a one-loop $n$-particle amplitude, a set of $d$-dimensional denominators

$$D_i(q) = (\bar{q} + pt_i)^2 - m_i^2, \quad i = 0, 1, \ldots, n - 1$$

(2.1)
can generally be identified such that the amplitude can be represented in the form

$$A = \sum_{I \subset \{0, 1, \ldots, n-1\}} \int \frac{\mu^{4-d}d^d\bar{q}}{(2\pi)^d} \frac{\tilde{N}_I(\bar{q})}{\prod_{i \in I} D_i(\bar{q})},$$

(2.2)

where $\bar{q}$ is the loop momentum in $d$ dimensions and $\tilde{N}_I(\bar{q})$ is the numerator calculated also in $d$ dimensions.

It is a well known fact that when $d \to 4$ limit is taken, the amplitude can be cast into the form

$$A = \sum_i d_i \text{Box}_i + \sum_i c_i \text{Triangle}_i + \sum_i b_i \text{Bubble}_i + \sum_i a_i \text{Tadpole}_i + R,$$

(2.3)

where $\text{Box}$, $\text{Triangle}$, $\text{Bubble}$ and $\text{Tadpole}$ refer to the well known scalar one-loop functions and $R = R_1 + R_2$ is the so-called rational term.

The reduction of eq. (2.2) to eq. (2.3) is the first ingredient of any approach aiming in the calculation of virtual corrections. In the following we will follow the so called reduction at the integrand level, developed by Ossola, Papadopoulos and Pittau [47]. The main idea is that any numerator function can be written as

$$N(q) = \sum_{i_0 < i_1 < i_2 < i_3} \left[ d(i_0 i_1 i_2 i_3) + \tilde{d}(q; i_0 i_1 i_2 i_3) \right] \prod_{i \neq i_0, i_1, i_2, i_3} D_i$$

$$+ \sum_{i_0 < i_1 < i_2} \left[ c(i_0 i_1 i_2) + \tilde{c}(q; i_0 i_1 i_2) \right] \prod_{i \neq i_0, i_1, i_2} D_i$$

$$+ \sum_{i_0 < i_1} \left[ b(i_0 i_1) + \tilde{b}(q; i_0 i_1) \right] \prod_{i \neq i_0, i_1} D_i$$

$$+ \sum_{i_0} \left[ a(i_0) + \tilde{a}(q; i_0) \right] \prod_{i \neq i_0} D_i$$

$$+ \tilde{P}(q) \prod_i D_i,$$

(2.4)

where now $N(q)$ and $D_i$ are the four-dimensional versions of $\tilde{N}(\bar{q})$ and $\tilde{D}_i(\bar{q})$. The coefficients $d$, $c$, $b$ and $a$ appearing in eq. (2.4) are independent of the loop momentum and the
same as the ones in eq. (2.3), whereas the new coefficients \( \tilde{d}, \tilde{c}, \tilde{b}, \tilde{a} \) and \( \tilde{P}(q) \), called also spurious terms, are depending on the loop momentum and they integrate to zero.

Depending on the reduction method used, the calculation of any one-loop amplitude is placed in a very different perspective. For instance eq. (2.4) can be solved in the unitarity way namely by computing the numerator functions for specific values of the loop momentum, that are solutions of equations of the form

\[
D_i(q) = 0, \quad \text{for } i = 0, \ldots, m - 1
\]  

(2.5)

It is customary to refer to these equations as quadruple \((m = 4)\), triple \((m = 3)\), double \((m = 2)\) and single \((m = 1)\) cuts.

Calculating the numerator function for specific values of the loop momentum, opens the possibility to use tree-level amplitudes as building blocks. The reason is rather obvious: the numerator function is nothing but a sum of individual Feynman graphs with the given structure of denominators and as we will see in a while, it is part of a tree amplitude with \(n + 2\) particles. This is by itself a very attractive possibility, since one can use existing algorithms and tools that perform tree-order amplitude calculations, exploiting their automation, simplicity and speed. Indeed in the sequel we will describe how using HELAC, a programme that is capable to compute any tree-order amplitude, we can also compute any one-loop amplitude.

The existing public version of HELAC was the first implementation of the Dyson-Schwinger (DS) recursive equations for the full Standard Model. During initialization (first phase) HELAC performs a solution of the DS equations, expressing sub-amplitudes with \(k\) external particles, in terms of sub-amplitudes with \(k - 1, k - 2, \ldots, 1\) external particles. The solution is represented by a sequence of integer arrays, encoding the information that is needed for the calculation of each sub-amplitude. Since this information is the minimal required for the calculation of the full amplitude, once particle momenta are available, HELAC provides a fast and efficient tool to compute any scattering amplitude.

The idea to proceed to the one loop level is rather straightforward. The aim is to collect all contributions with a given denominator structure. This will allow to calculate the corresponding numerator function \(N(q)\) in four dimensions. We will illustrate the procedure with a 6-particle amplitude.

The input of the calculation is as usual the flavor of the 6 external particles. In HELAC a binary representation will be used to order the external particles, called also level one sub-amplitudes. For instance in a 6-particle amplitude external momenta are labeled by the number 1, 2, 4, 8, 16 and 32. In the one-loop case, also the flavor of the allowed particles in the loop has to be taken into account as an input information. The first step is the construction of all topologically inequivalent partitions of the external particles into the highest possible number of sets (blobs), namely 6 in our case. One such contribution
is schematically represented as

The labels $f$ and $c$ refers to the possible flavor and color of the internal particles. This construction will continue to include also pentagon-topologies, tetragon-topologies, triangle-topologies, and so on. A typical collection of possible contributions, looks like

If for instance, the color degree of freedom is omitted (we will come back to this below), as is indeed the case for amplitudes involving colorless particles, the selection of all these contributions is enough for the calculation of the one-loop amplitude. Practically each numerator contribution, will be calculated as part of the $n+2$ tree-order amplitude subject to the constraint that the attached blobs, will contain no propagator depending on the loop momentum.
and no denominator will be used for the internal loop propagators. Cutting now the line connecting the blob containing the particle number 1 and the last blob, it is easy to see that we have nothing more that a part of the \( n + 2 \) amplitude. The ‘cut’ particles with flavor \( f \), will now acquire their usual numbering \( 2^n \) and \( 2^{n+1} \), \( (64 \) and \( 128 \) for \( n = 6 \)).

**HELAC** will know how to reconstruct all information needed for the calculation and store it as a sequence of sub-amplitudes, exactly as in the tree-order calculation.

One important aspect of the one-loop calculations is also the treatment of the color degrees of freedom. In **HELAC** the color connection representation is used. External quarks and anti-quarks are represented as usually by a color or anti-color index. Gluons are also represented by a pair of color/anti-color indices. This is achieved by multiplying the amplitude with a matrix in the fundamental representation \( t^a_i \) and summing over the adjoint index \( a \) for all gluons. In this way any amplitude, with any number of gluons and quarks(anti-quarks) can be represented as

\[
M^{i_1,i_2,...,i_k}_{j_1,j_2,...,j_k} = \sum_{\sigma} \delta_{i_{\sigma_1},j_1} \delta_{i_{\sigma_2},j_2} \ldots \delta_{i_{\sigma_k},j_k} A_\sigma
\]

with \( i \) and \( j \) referring to color/anti-color indices, taking values from 1 to \( N_c \) for \( SU(N_c) \) \((N_c = 3 \) for QCD). If \( n_g \) is the number of gluons and \( n_q \) the number of quarks (equal also the number of anti-quarks) then \( k = n_g + n_q \). Moreover the amplitude can now be decomposed as

\[
M^{i_1,i_2,...,i_k}_{j_1,j_2,...,j_k} = \sum_{\sigma} \delta_{i_{\sigma_1},j_1} \delta_{i_{\sigma_2},j_2} \ldots \delta_{i_{\sigma_k},j_k} A_\sigma
\]

where the sum is running over all permutations \( \sigma \) of the set \( \{1, 2, 3, \ldots, n_l\} \). Using the Feynman rules described in ref. [6], **HELAC** calculates the color-stripped amplitudes \( A_\sigma \). Namely for given flavor of external particles, **HELAC** generates all color connections, and for each one of them reconstruct the integer arrays allowing the calculation of all necessary sub-amplitudes compatible with the color connection under consideration. For instance for a 6-gluon tree-level amplitude, out of the 720 (=6!) color connections, the DS recursive equations automatically and correctly single out the 120 (=5!) that are non-zero. This is a mere fact of the Feynman rules and leads to what is called ‘planar’ topologies.

The color summed matrix element squared is given by

\[
\sum_{\{i\},\{j\}} |M^{i_1,i_2,...,i_k}_{j_1,j_2,...,j_k}|^2
\]
which can be written also as
\[ \sum_{\sigma, \sigma'} A_{\sigma}^* C_{\sigma, \sigma'} A_{\sigma'} \]
where the color matrix \( C_{\sigma, \sigma'} \) is defined by
\[ C_{\sigma, \sigma'} \equiv \sum_{\{i\}, \{j\}} \delta_{i_{\sigma_1}, j_1} \delta_{i_{\sigma_2}, j_2} \ldots \delta_{i_{\sigma_k}, j_k} \delta_{i_{\sigma'_1}, j_1} \delta_{i_{\sigma'_2}, j_2} \ldots \delta_{i_{\sigma'_k}, j_k} \]

In practice HELAC uses the following representation for the color connection: a gluon is represented by a two-element array \((x, y)\), incoming quarks (outgoing anti-quarks) with \((x, 0)\) and outgoing quarks (incoming anti-quarks) with \((0, y)\). So for any process \((x_1, y_1) \ldots (x_n, y_n)\)

where \(y_i\) take the values \(\{1, 2, \ldots, n_l\}\) if \(i\) is a gluon or an outgoing quark (incoming anti-quark) otherwise \(y_i = 0\), whereas \(x_i\) take the values \(\{\sigma_1, \sigma_2, \ldots, \sigma_n\}\) if \(i\) is a gluon or an incoming quark (outgoing anti-quark) otherwise \(x_i = 0\). So for instance for a \(q\bar{q} \rightarrow gg\) process, \(n_l = 3\) and a possible color connection is given by \((3, 0)(0, 1)(1, 2)(2, 3)\)

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{color_connection1.png}
\caption{Possible color connection for \(q\bar{q} \rightarrow gg\).}
\end{figure}

whereas for \(gg \rightarrow ggg\), \(n_l = 5\) and a possible color connection is given by \((2, 1)(3, 2)(4, 3)(5, 4)(1, 5)\)

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{color_connection2.png}
\caption{Possible color connection for \(gg \rightarrow ggg\).}
\end{figure}

As is evident the second element which corresponds to the anti-color index is always in the nominal order, whereas the first element which corresponds to the color index inherits
the order of the permutation \( \sigma \). The color matrix element is given by

\[
C_{\sigma, \sigma'} = N_c^{m(\sigma, \sigma')}
\]

where \( m(\sigma, \sigma') \) count the number of common cycles of the two permutations.

The extension at the one-loop level is straightforward. Since after the one-particle cutting one has to deal with an \( n + 2 \) tree-order matrix element, the same Feynman rules apply [6]. For planar topologies, in a 6-gluon amplitude for example, what is called color-ordered or primitive,

the solution is quite obvious, with the mere addition of two more color lines. For the non-planar ones the one-particle cutting results into a tree-order calculation with a different ordering. Obviously, a relabeling of the color lines is necessary after the one-particle cutting, and this is what is done in this case.

**HELAC** second phase is synonymous of the 'real' computation of the amplitude, namely assigning a complex number to it. To this end the information on the external particle momenta should be provided. This is done by default, using the phase-space generator
PHEGAS, but this is not exclusive. Once the momenta of the external particles are given, their polarizations are defined and therefore their wave functions are computed. At the tree order this is all we need, and the programme returns the value of the amplitude for each color connection. At the one-loop level though, slightly more steps are required. First of all, the numerator functions have to be analyzed by a reduction algorithm. To this end the programme CutTools is used. It returns, for each of the numerator function the corresponding cut-constructible and $R_1$ contribution, fully interfaced to the scalar loop functions. For the scalar one-loop functions, two possible interfaces are available for the moment, namely QCDLoop [19] and OneLOop(Appendix A). In order to compute the numerator function, the polarization vectors of the two extra 'external' particles, after the one-particle cut, are also calculated. Within the Feynman gauge for gauge bosons, the sum over four different (4-dimensional) polarizations, that satisfy $\sum_i \epsilon_i^\mu \epsilon_i^\nu = g^{\mu \nu}$ is performed. Ghost particles are also included. Finally for fermions, four vectors in spinor space, satisfying $\sum_i u^{(i)}_\alpha (x_{n_i+1}) u^{(i)}_\beta (y_{n_i+1}) = (\not{q} + m)_{\alpha \beta}$ are used.

Up to now we have described how HELAC and CutTools are able to compute the cut-constructible and $R_1$ part of the rational term for any amplitude and for any color connection. It turns out that the calculation of the contribution to the so-called $R_2$ part of the rational term is an even easier task. The reason is very simple: to calculate $R_2$ part one has to calculate tree-order contributions, with given extra Feynman rules [60]. This is completely similar to the calculation of the counter-term contributions, needed in any case in order to obtain renormalized amplitudes. This task for HELAC is completely straightforward. All tree-order contributions, including only one of these extra vertices are also generated in the same format described so far. It is worth to emphasize at this point that the actual computation of $R_2$ costs 'nothing' compared to the computation of the cut-constructible and $R_1$ parts, as it is expected, being a fully tree-order computation.

On the same footing the ultra-violet (UV) counter-term contributions are included. At the present stage, with NLO QCD corrections in mind, the counter-term included in the same automatic way are: first the gauge-coupling and the wave function renormalization, which are trivially proportional to the tree-order amplitudes, and second the one related to the mass renormalization, $\delta m$. They give contributions both to the $\epsilon^0$ and $\epsilon^{-1}$ terms.

Finally, as part of the whole project to compute NLO QCD corrections, the infrared 'counter-terms' have been also included automatically. This is based on the formulae provided by Catani and Seymour [61] and Catani, Dittmaier and Trocsanyi [62] and cover the full range of interest. It is worth to emphasize that the color correlations have a natural interpretation within the color connection representation. Indeed the color correlation matrix, which accounts for the gluon emission from particle $i$ and absorption from particle $j$, can be reconstructed as follows: taken any two color connections of the tree-order configuration, $I = \{(x_k,y_k)\}_{k=1,\ldots,n_i}$ and $J = \{(x'_k,y'_k)\}_{k=1,\ldots,n_j}$ we add to both an extra gluon $(x_{n_i+1},y_{n_i+1})$ and $(x'_{n_j+1},y'_{n_j+1})$, with the rules, $x_{n_i+1} = x_i$, $y_{n_i+1} = n_i + 1$ and
replacing \( x_i = n_l + 1 \), and the same for \( x'_{n_l+1} = x'_i \), \( y_{n_l+1} = n_l + 1 \) and \( x'_i = n_l + 1 \). The same rules also apply for anti-colors, with an appropriate relabeling in order to comply with the nominal order in the anti-color indices. If we label the two new color connections with the extra gluon as \( I' \) and \( J' \) then the color-correlated matrix element is given by \( C_{I',J'}^{(i,j)} = (N^m_{c(I',J')}) - \frac{1}{N_c} N^m_{c(I,J)}) p_{ij} \) with \( p_{ij} = -1 \) if the emission is from the color line of particle \( i \) and the absorption from the anti-color line of particle \( j \) (and vice-versa) and \( p_{ij} = 1 \) otherwise. In this way the integrated dipole counter-term is also included in the calculation in a fully automated way.

Summarizing the procedure to calculate one-loop amplitudes, fully automated, at this stage is as follows:

1. Construction of all numerator functions using HELAC format, namely minimal information for the calculation of all sub-amplitudes, within the \( n + 2 \) tree-order matrix element. All flavors within the SM can be included either as external or internal (loop) particles. All particles can have arbitrary masses.

2. Each numerator function is reduced using CutTools. The cut-constructible and \( R_1 \) part of the rational term is obtained.

3. Construction of all counter-term contributions needed for the calculation of \( R_2 \) part of the rational part. At this stage of the implementation of our method, all extra \( R_2 \)-vertices needed for NLO QCD calculations have been included.

4. Construction of all UV counter-term contributions needed to renormalize the amplitude.

5. Construction of all IR counter-term contributions needed to obtain a finite expression after the cancelation of the \( \epsilon^{-2} \) and \( \epsilon^{-1} \) poles.

As far as testing of our calculations is concerned, there are several tools in our disposal. The first is the Ward Identities test, whenever a massless gauge boson is present as external particle. Replacing the wave function with its momentum, one expects the answer to be nullified. This is a very powerful and useful test. The second is the cancelation of infrared poles. Having implemented in a fully automated way, the infrared poles of the amplitude based on the tree-level color-correlated matrix elements squared – the \( I(\epsilon) \) operator \([61, 62]\) – we are able to test the reliability of our calculations.

Finally, in order to better clarify possible common features or differences with other approaches, we would like to make the following remarks:

1. As far as the 'classical' way of doing one-loop computations is concerned there are two main differences concerning the reduction method used and the organization of the calculation. The PV-reduction, imposes the use of computer algebra programmes, in
order to produce manageable expressions. We would like to emphasize that in our approach the whole setup is purely numerical. Moreover, calculating the integrand for specific values of the loop momentum, instead of manipulating products of momenta, Dirac matrices and metric tensors, gives us the possibility to use the tree-order matrix elements as building blocks. To the best of our knowledge, also the automation of the whole procedure becomes a much more simple task in our case.

2. The other commonly used approach is the unitarity one. As far as the reduction is concerned for the cut-constructible part of the amplitude, this is nowadays identical to our method. There is still a different way in the organization of the calculation. The first obvious difference is that in the unitarity approach the blobs attached to the loop contain the full contribution, whereas in our case we have split it in order to avoid the presence of denominators depending on the loop-momentum. This is not a matter of principle, but just a different bookkeeping. On the other hand the use of on-shell blobs, instead of DS equations, induces a summation over intermediate polarizations. In any case we believe that these differences are minor. For the moment the more obvious difference is related to the color treatment. As we have shown, all color connections for any scattering amplitude, are straightforwardly accessible within our framework.

3. Finally a more recent approach, named ‘generalized unitarity’ aims in unifying the computation of cut-constructible and rational terms. As far as the organization of the calculation is concerned, the remarks of the previous paragraph apply as well. As far as the reduction is concerned, we notice that in the ‘generalized unitarity’ case, more coefficients are needed in the expansion of the one-loop integrand, coming from the use of internal particles in higher dimensions, and the use of pentuple cuts. In contrast within our method all calculations are performed in four dimensions. Nevertheless, to our opinion, the relative merit of the two methods, has still to be judged within actual calculations. After all, as is usually the case, a hybrid of all different methods used so far, may at the end provide the optimal solution, not neglecting of course the possibility of new breakthroughs in the field.

3. Results

The current implementation allows to calculate any one-loop virtual matrix element, for all color and helicity configurations, with any external particle and with particles in the loop that can be either gluons (ghosts) or quarks of any flavor. Moreover the cut-constructible part can be obtained for any internal particle. When the rational counterterms for the full Standard Model will be implemented, then any one-loop amplitude will be obtainable. The calculation is done in a fully automatic way, and it is purely numerical.
In this section we will present indicative results for a given phase-space point, for subprocesses that basically exhaust the so called wish list given in ref. [1]. In all cases we use a fixed strong coupling $g_{\text{QCD}} = 1$, and a renormalization scale $\mu = \sqrt{s}$. The result refers to the squared matrix element, fully summed and averaged over colors and helicities, including all fermion loop contributions with $N_f = 6$ flavors out of which five are considered as massless. Mass renormalization is also included in the result, both at order $\epsilon^{-1}$ and $\epsilon^0$, according to the conventions of ref. [31]. The top-quark mass is taken $m_{\text{top}} = 174$ GeV. An overall normalization factor is as usually considered

$$C_T = \frac{(4\pi)^\epsilon \Gamma(1 + \epsilon)\Gamma^2(1 - \epsilon)}{16\pi^2 \Gamma(1 - 2\epsilon)}$$

In all tables following below, $I(\epsilon)$ represents the result as predicted by the color-correlated tree-order matrix elements [61, 62], including also the $\epsilon^{-1}$ contributions of the coupling constant and wave function renormalizations. Therefore the $I(\epsilon)$-result should agree with that of HELAC-1L for $\epsilon^{-2}$ and $\epsilon^{-1}$. Momenta are given in GeV and all results are in the 't Hooft-Veltman scheme [15]. The electroweak parameters are taken from the default values used by HELAC, namely

$$m_Z = 91.188 \text{ GeV}, \quad m_W = 80.419 \text{ GeV}, \quad \sin^2 \theta_W = 1 - \frac{m_W^2}{m_Z^2}, \quad G_F = 1.1663910^{-5} \text{ GeV}^{-2}$$

whereas the electromagnetic coupling constant is given by

$$\alpha_{em} = \sqrt{2} G_F m_W^2 \sin^2 \theta_W / \pi$$

| subprocess | $\epsilon^{-2}$ | $\epsilon^{-1}$ | $\epsilon^0$ |
|------------|----------------|----------------|-------------|
| $pp \to t\bar{t}bb$ | HELAC-1L | -2.347908989000179E-07 | -2.082520105681483E-07 | 3.909384299635230E-07 |
| $u\bar{u} \to t\bar{t}bb$ | HELAC-1L | -2.347908989000179E-07 | -2.082520105665445E-07 | 3.909384299635230E-07 |
| $gg \to t\bar{t}bb$ | HELAC-1L | -1.435108168334016E-06 | -2.08507077363073E-06 | 3.616343483497464E-06 |
| $I(\epsilon)$ | HELAC-1L | -1.435108168334035E-06 | -2.085070773651439E-06 |

The momenta used to obtain the above result are

| | $p_x$ | $p_y$ | $p_z$ | $E$ |
|----------------|-------|-------|-------|-----|
| $u(g)$ | 0 | 0 | 250 | 250 |
| $\bar{u}(g)$ | 0 | 0 | -250 | 250 |
| $t$ | 12.99421901255723 | -9.591511769543683 | 75.05543670827210 | 190.184556191092 |
| $\bar{t}$ | 53.73271578143694 | -0.2854146459513714 | 17.68101382654795 | 182.9642163285034 |
| $b$ | -41.57664370692741 | 3.8955311350980977 | -91.94931862397770 | 100.9874727883170 |
| $\bar{b}$ | -25.15029108706678 | 5.981395280396083 | -0.7871319108423604 | 25.86375471407044 |
\[ pp \rightarrow VVbb \text{ and } pp \rightarrow VV + 2 \text{ jets} \]

\[ uu \rightarrow W^+W^-bb \]

| HELAC-1L | \( \epsilon^{-2} \) | \( \epsilon^{-1} \) | \( \epsilon^0 \) |
|----------|-----------------|-----------------|-----------------|
| -2.49391693959002E-07 | -4.885901774740355E-07 | 1.59253853368835E-07 |

| \( I(\epsilon) \) | -2.49391693959001E-07 | -4.885901774752593E-07 |

\[ gg \rightarrow W^+W^-b\overline{b} \]

| HELAC-1L | \( \epsilon^{-2} \) | \( \epsilon^{-1} \) | \( \epsilon^0 \) |
|----------|-----------------|-----------------|-----------------|
| -2.68631059222120E-07 | -6.078623164346466E-07 | 2.43162440346638E-07 |

| \( I(\epsilon) \) | -2.686310592221206E-07 | -6.07862340168020E-07 |

The momenta used to obtain the above result are

\[ pp \rightarrow bbbb \]

\[ uu \rightarrow bbbb \]

\[ gg \rightarrow bbbb \]

\[ pp \rightarrow bbbb \]

\[ uu \rightarrow bbbb \]

\[ gg \rightarrow bbbb \]

The momenta used to obtain the above result are
\[ pp \rightarrow V + 3 \text{ jets} \]

\[ ud \rightarrow W^+ ggg \]

| HELAC-1L | \( \epsilon^{-2} \) | \( \epsilon^{-1} \) | \( \epsilon^0 \) |
|----------|----------------|----------------|----------------|
| -1.995636628164684E-05 | -5.935610843551600E-05 | -5.323285370666314E-05 |
| \( I(\epsilon) \) | -1.995636628164686E-05 | -5.935610843566534E-05 |

\[ uu \rightarrow Z ggg \]

| HELAC-1L | \( \epsilon^{-2} \) | \( \epsilon^{-1} \) | \( \epsilon^0 \) |
|----------|----------------|----------------|----------------|
| -7.148261887172997E-06 | -2.142170009323704E-05 | -1.906378375774021E-05 |
| \( I(\epsilon) \) | -7.148261887172976E-06 | -2.142170009540120E-05 |

The momenta used to obtain the above result are

\[
\begin{array}{cccc}
  p_x & p_y & p_z & E \\
  u & 0 & 0 & 250 & 250 \\
  \bar{d} & 0 & 0 & -250 & 250 \\
  W^+ & 23.90724239064912 & -17.64681636854432 & 138.0897548661186 & 162.5391101447744 \\
  g & 98.8594281236483 & -0.5251163702879512 & 32.53017998659339 & 104.0753327455388 \\
  g & -76.49423931754684 & 7.167141557113385 & -169.1717405928078 & 185.8004692730082 \\
  g & -46.27243119673712 & 11.00479118171890 & -1.448194259904179 & 47.58508783667868 \\
  \bar{u} & 0 & 0 & 250 & 250 \\
  \bar{u} & 0 & 0 & -250 & 250 \\
  Z & 23.61417669184427 & -17.43049377950531 & 136.3969887224391 & 166.6758570722832 \\
  g & 97.64756491862407 & -0.5186792583242352 & 32.13141045164495 & 102.7995306425180 \\
  g & -75.55653862694311 & 7.09283521688509 & -1.430441645200695 & 47.00176848969281 \\
  g & -45.70520298352523 & 10.86988951614105 & -1.448194259904179 & 47.58508783667868 \\
\end{array}
\]

\[ pp \rightarrow \bar{t} \bar{t} + 2 \text{ jets} \]

\[ uu \rightarrow \bar{t} \bar{t} ggg \]

| HELAC-1L | \( \epsilon^{-2} \) | \( \epsilon^{-1} \) | \( \epsilon^0 \) |
|----------|----------------|----------------|----------------|
| -6.127108113312741E-05 | -1.874963444741646E-04 | -3.305349683690902E-04 |
| \( I(\epsilon) \) | -6.127108113312702E-05 | -1.874963445081074E-04 |

\[ gg \rightarrow \bar{t} \bar{t} ggg \]

| HELAC-1L | \( \epsilon^{-2} \) | \( \epsilon^{-1} \) | \( \epsilon^0 \) |
|----------|----------------|----------------|----------------|
| -3.838786514961561E-04 | -9.76116889507888E-04 | -5.225385984750410E-04 |
| \( I(\epsilon) \) | -3.838786514961539E-04 | -9.761168898436521E-04 |

The momenta used to obtain the above result are
As far as the speed is concerned, it is comparable with the results presented in \([53, 54, 63]\). We would like to emphasize though that this is only a measure of the efficiency of the current implementation of the algorithm and any extrapolation to a real calculation is to a very large extent misleading. The reason is that for a real calculation the integration over phase space will use the tree-order matrix-element squared for optimization, which is much cheaper in CPU time, along with a Monte-Carlo sampling over colors and helicities for the one-loop virtual corrections. Therefore the overall speed and efficiency should be assessed within this sampling approach. The same is true for the numerical stability. Therefore we will postpone a detailed discussion for these subjects, when a full calculation for those processes, including the real corrections based on our current implementation will be performed. Needless to notice that improvements of the current implementation are under study, both in calculating the cut-constructible and \( R_1 \) contributions in CutTools, as well as in using of HELAC at one loop.

4. Summary and Outlook

In this paper we have presented an algorithm, fully automated, to evaluate any one-loop amplitude. We have implemented this algorithm using HELAC and CutTools. The current implementation supports any one-loop amplitude with any number and species of external particles, but with colored particles running in the loop: for the cut-constructible part even this last restriction does not apply. It is able to produce results for all color connections, therefore is not restricted to primitive amplitudes or large\( -N_c \) approximation. Generic tests of the correctness of our results, include Ward Identities, and IR poles structure. We have also tested the full result against available calculations. In the near future we plan to use this implementation to perform realistic calculations for LHC processes, including real corrections within the dipole formalism \([61, 64]\).

Acknowledgments

We would like to thank T. Binoth and T. Reiter for providing us results on 6-quark amplitudes in order to cross check our code. We would like also to thank M. Czakon, A. Denner, P. Draggiotis, A. Lazopoulos, P. Mastrolia and G. Ossola, for helpful discussions.
A.vH. and R.P. acknowledge the financial support of the ToK Program “ALGOTOOLS” (MTKD-CD-2004-014319). Research was also partially supported by the RTN European Programme MRTN-CT-2006-035505 (HEPTOOLS, Tools and Precision Calculations for Physics Discoveries at Colliders). The research of R.P. was also supported by the MEC project FPA2008-02984.

Appendices

A. OneLOop for evaluating scalar loop integrals

The program deals with all finite and IR-divergent scalar 4-point, 3-point, 2-point and 1-point functions for all relevant real mass combinations and all relevant regions of phase-space, and the IR-divergent cases are dealt with within dimensional regularization. The implementations of the IR-divergent scalar functions with all internal masses equal to zero are based on the formulas from [40,65,66]. The implementations of the IR-divergent scalar functions with non-zero internal masses are based on the formulas from [19,31,67–69]. The implementation of the finite 4-point scalar function is based on the formulas from [70], and the finite 3-point function is based on the formulas obtained from these by taking one of the masses to infinity. The 2-point scalar function, finally, is based on the formula as found in [25].

Some details worth mentioning are, firstly, that the IR-divergent 4-point functions have consistently been expressed in terms of the variables as defined in [70] for the finite 4-point function. These are

\[ k_{ij} = \frac{m_i^2 + m_j^2 - (p_i - p_j)^2}{2m_im_j} \]  

(A.1)

where \(1 \leq i < j \leq 4\) and where \(p_i, m_i\) are the momentum and mass associated with propagator \(i\). If one of the masses is zero, its appearance in the denominator is replaced by another scale, also as suggested in [70]. Furthermore, we used the variables \(r_{ij}\) defined in [70] such that

\[ k_{ij} = r_{ij} + 1/r_{ij}, \]  

(A.2)

ez essentially instead of the function \(K(z, m, m')\) used in [19,68].

Secondly, formulas have been numerically stabilized as much as possible by expressing them in terms of the functions

\[ \frac{\log(x)}{1-x}, \quad \frac{\operatorname{Li}_2(1-x) - \operatorname{Li}_2(1-y)}{x-y}, \]  

(A.3)

and by using stable implementations of these.

In order to achieve the analytic continuation of the \(\operatorname{Li}_2\)-function, we used the formula (B.3) from [68], also given in [19]. It can be formulated as follows: understanding the meaning of the formula

\[ \log(e^{2ni\pi}) = \log(z) + 2ni\pi \]  

(A.4)
for complex number $z$ and integer $n$, the formula from [68] can be expressed as

$$
\text{Li}_2(1 - z e^{2ni\pi}) = \text{Li}_2(1 - z) - 2ni\pi \left\{ \log(1 - z) + \theta(|z| - 1) \left[ ni\pi + \log(z) - \log(-z) \right] \right\}.
$$

(A.5)

This formula tells us how to evaluate the Li$_2$-functions on different Riemann sheets. In order to evaluate Li$_2(1 - z)$ when $z$ is a product of several complex numbers, we should now keep track of the overall phase of this product. This can be conveniently achieved by writing complex numbers as

$$
z = c(z) e^{n(z)i\pi}
$$

(A.6)

where $n(z)$ is an integer, and $c(z)$ is a complex number with a positive real part. For real numbers within an $i\varepsilon$-prescription we have

$$
x + i\varepsilon \rightarrow |x| e^{\theta(-x)\text{sign}(\varepsilon)i\pi}
$$

(A.7)

One can keep track of the overall phase by applying the multiplication rule

$$
c(yz) = \text{sign}(\text{Re} c(y)c(z)) c(y)c(z)
$$

(A.8)

$$
n(yz) = n(y) + n(z) + \theta(-\text{Re} c(y)c(z)) \text{sign}(\text{Im} c(y))
$$

(A.9)

Notice that the step-function is only non-zero if \text{sign}(\text{Im} c(y)) = \text{sign}(\text{Im} c(z))$. If the phase $n$ of the final complex number at which the Li$_2$-functions has to be evaluated is odd, one should evaluate

$$
\text{Li}_2\left( 1 + ce^{(n+\text{sign}(\text{Im} c))i\pi} \right).
$$

(A.10)

Finally, we use, as suggested in [19], the more practical relation

$$
\text{Li}_2(1 - z e^{2ni\pi}) + \text{Li}_2(1 - e^{-2ni\pi}/z) = -\frac{1}{2} \left( \log(z e^{2ni\pi}) \right)^2
$$

instead of Eq.(A.3) when $|z| > 1$.

**References**

[1] Z. Bern et al. [NLO Multileg Working Group], arXiv:0803.0494 [hep-ph].

[2] M. L. Mangano, M. Moretti, F. Piccinini, R. Pittau and A. D. Polosa, JHEP 0307 (2003) 001 [arXiv:hep-ph/0206293].

[3] F. Maltoni and T. Stelzer, JHEP 0302 (2003) 027 [arXiv:hep-ph/0208156].

[4] T. Gleisberg, S. Hoche, F. Krauss, A. Schalicke, S. Schumann and J. C. Winter, JHEP 0402 (2004) 056 [arXiv:hep-ph/0311263].

[5] A. Kanaki and C. G. Papadopoulos, Comput. Phys. Commun. 132 (2000) 306 [arXiv:hep-ph/0002082].

[6] A. Kanaki and C. G. Papadopoulos, arXiv:hep-ph/0012004.
[7] C. G. Papadopoulos, Comput. Phys. Commun. 137 (2001) 247 [arXiv:hep-ph/0007335].
[8] A. Cafarella, C. G. Papadopoulos and M. Worek, arXiv:0710.2427 [hep-ph].
[9] W. Kilian, T. Ohl and J. Reuter, arXiv:0708.4233 [hep-ph].
[10] J. Alwall et al., Eur. Phys. J. C 53 (2008) 473 [arXiv:0706.2569 [hep-ph]].
[11] F. A. Berends and W. T. Giele, Nucl. Phys. B 306 (1988) 759.
[12] F. Caravaglios and M. Moretti, Phys. Lett. B 358 (1995) 332 [arXiv:hep-ph/9507237].
[13] P. Draggiotis, R. H. P. Kleiss and C. G. Papadopoulos, Phys. Lett. B 439 (1998) 157 [arXiv:hep-ph/9807207].
[14] P. D. Draggiotis, R. H. P. Kleiss and C. G. Papadopoulos, Eur. Phys. J. C 24 (2002) 447 [arXiv:hep-ph/0202201].
[15] G. ’t Hooft and M. J. G. Veltman, Nucl. Phys. B 44 (1972) 189.
[16] G. ’t Hooft, Nucl. Phys. B 62 (1973) 444.
[17] G. ’t Hooft and M. J. G. Veltman, Nucl. Phys. B 153 (1979) 365.
[18] G. J. van Oldenborgh, Comput. Phys. Commun. 66 (1991) 1.
[19] R. K. Ellis and G. Zanderighi, JHEP 0802 (2008) 002 [arXiv:0712.1851 [hep-ph]].
[20] R. K. Ellis, Nucl. Phys. Proc. Suppl. 160 (2006) 170.
[21] R. Mertig, M. Bohm and A. Denner, Comput. Phys. Commun. 64 (1991) 345.
[22] T. Hahn, arXiv:0901.1528 [hep-ph].
[23] J. A. M. Vermaseren, arXiv:math-ph/0010025.
[24] G. Passarino and M. J. G. Veltman, Nucl. Phys. B 160 (1979) 151.
[25] A. Denner, Fortsch. Phys. 41 (1993) 307 [arXiv:0709.1075 [hep-ph]].
[26] A. Denner and S. Dittmaier, Nucl. Phys. B 658 (2003) 175 [arXiv:hep-ph/0212259].
[27] A. Denner and S. Dittmaier, Nucl. Phys. B 734 (2006) 62 [arXiv:hep-ph/0509141].
[28] T. Hahn, Comput. Phys. Commun. 140 (2001) 418 [arXiv:hep-ph/0012260].
[29] P. Nogueira, J. Comput. Phys. 105 (1993) 279.
[30] T. Hahn, Nucl. Phys. Proc. Suppl. 89 (2000) 231 [arXiv:hep-ph/0005029].
[31] W. Beenakker, S. Dittmaier, M. Kramer, B. Plumber, M. Spira and P. M. Zerwas, Nucl. Phys. B 653 (2003) 151 [arXiv:hep-ph/0211352].
[32] A. Denner, S. Dittmaier, M. Roth and L. H. Wieders, Phys. Lett. B 612 (2005) 223 [arXiv:hep-ph/0502063].
[33] S. Dittmaier, P. Uwer and S. Weinzierl, Phys. Rev. Lett. 98 (2007) 262002 [arXiv:hep-ph/0703120].
[34] S. Dittmaier, S. Kallweit and P. Uwer, Phys. Rev. Lett. 100 (2008) 062003 [arXiv:0710.1577 [hep-ph]].
[35] A. Bredenstein, A. Denner, S. Dittmaier and S. Pozzorin, JHEP 0808 (2008) 108 [arXiv:0807.1248 [hep-ph]].

[36] T. Binoth, J. P. Guillet, G. Heinrich, E. Pilon and T. Reiter, arXiv:0810.0992 [hep-ph].

[37] T. Reiter, arXiv:0903.0947 [hep-ph].

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

[39] F. del Aguila and R. Pittau, JHEP 0407 (2004) 017 [arXiv:hep-ph/0404120].

[40] A. van Hameren, J. Vollinga and S. Weinzierl, Eur. Phys. J. C 41 (2005) 361 [arXiv:hep-ph/0502165].

[41] Z. Bern, L. J. Dixon, D. C. Dunbar and D. A. Kosower, Nucl. Phys. B 435 (1995) 59 [arXiv:hep-ph/9409265];

[42] Z. Bern, L. J. Dixon, D. C. Dunbar and D. A. Kosower, Nucl. Phys. B 425, 217 (1994);

[43] Z. Bern, L. J. Dixon and D. A. Kosower, Phys. Rev. Lett. 70 (1993) 2677 [arXiv:hep-ph/9302280].

[44] Z. Bern, L. J. Dixon and D. A. Kosower, Nucl. Phys. B 513 (1998) 3 [arXiv:hep-ph/9708239].

[45] Z. Bern, L. J. Dixon and D. A. Kosower, Nucl. Phys. B 437 (1995) 259 [arXiv:hep-ph/9409393].

[46] R. Britto, F. Cachazo and B. Feng, Nucl. Phys. B 725 (2005) 275 [arXiv:hep-th/0412103].

[47] G. Ossola, C. G. Papadopoulos and R. Pittau, Nucl. Phys. B 763 (2007) 147 [arXiv:hep-ph/0609007].

[48] G. Ossola, C. G. Papadopoulos and R. Pittau, JHEP 0707 (2007) 085 [arXiv:0704.1271 [hep-ph]].

[49] G. Ossola, C. G. Papadopoulos and R. Pittau, JHEP 0805 (2008) 004 [arXiv:0802.1876 [hep-ph]].

[50] W. T. Giele, Z. Kunszt and K. Melnikov, JHEP 0804 (2008) 049 [arXiv:0801.2237 [hep-ph]].

[51] R. K. Ellis, W. T. Giele, Z. Kunszt and K. Melnikov, arXiv:0806.3467 [hep-ph].

[52] C. Anastasiou, R. Britto, B. Feng, Z. Kunszt and P. Mastrolia, Phys. Lett. B 645 (2007) 213 [arXiv:hep-ph/0609191].

[53] C. F. Berger et al., Phys. Rev. D 78 (2008) 036003 [arXiv:0803.4180 [hep-ph]].

[54] W. T. Giele and G. Zanderighi, arXiv:0805.2152 [hep-ph].

[55] R. Britto, F. Cachazo and B. Feng, Nucl. Phys. B 715 (2005) 499 [arXiv:hep-th/0412308].

[56] R. K. Ellis, W. T. Giele, Z. Kunszt, K. Melnikov and G. Zanderighi, JHEP 0901 (2009) 012 [arXiv:0810.2762 [hep-ph]].

[57] R. K. Ellis, K. Melnikov and G. Zanderighi, arXiv:0901.4101 [hep-ph].

[58] C. F. Berger et al., arXiv:0902.2760 [hep-ph].
[59] G. Ossola, C. G. Papadopoulos and R. Pittau, JHEP 0803 (2008) 042 [arXiv:0711.3596 [hep-ph]].

[60] P. Draggiotis, M. V. Garzelli, C. G. Papadopoulos and R. Pittau, arXiv:0903.0356 [hep-ph].

[61] S. Catani and M. H. Seymour, Nucl. Phys. B 485 (1997) 291 [Erratum-ibid. B 510 (1998) 503] [arXiv:hep-ph/9605323].

[62] S. Catani, S. Dittmaier and Z. Trocsanyi, Phys. Lett. B 500 (2001) 149 [arXiv:hep-ph/0011222].

[63] A. Lazopoulos, arXiv:0812.2998 [hep-ph].

[64] S. Catani, S. Dittmaier, M. H. Seymour and Z. Trocsanyi, Nucl. Phys. B 627 (2002) 189 [arXiv:hep-ph/0201036].

[65] Z. Bern, L. J. Dixon and D. A. Kosower, Nucl. Phys. B 412 (1994) 751 [arXiv:hep-ph/9306240].

[66] G. Duplancic and B. Nizic, Eur. Phys. J. C 20 (2001) 357 [arXiv:hep-ph/0006249].

[67] W. Beenakker, H. Kuijf, W. L. van Neerven and J. Smith, Phys. Rev. D 40, 54 (1989).

[68] W. Beenakker and A. Denner, Nucl. Phys. B 338 (1990) 349.

[69] E. L. Berger, M. Klasen and T. M. P. Tait, Phys. Rev. D 62, 095014 (2000) [arXiv:hep-ph/0005196].

[70] A. Denner, U. Nierste and R. Scharf, Nucl. Phys. B 367, 637 (1991).