10-28-2010

Tensorial reconstruction at the integrand level

G. Heinrich
University of Durham

Giovanni Ossola
CUNY New York City College of Technology

T. Reiter
Nikhef

F. Tramontano
CERN

Recommended Citation

Heinrich, G.; Ossola, Giovanni; Reiter, T.; and Tramontano, F., "Tensorial reconstruction at the integrand level" (2010). *CUNY Academic Works.*
https://academicworks.cuny.edu/ny_pubs/405

Tensorial reconstruction at the integrand level

G. Heinrich,[a,b] **G. Ossola,**[c] **T. Reiter**[d] **and F. Tramontano**[e]

[a]*IPPP, Department of Physics, University of Durham,*
Durham DH1 3LE, U.K.

[b]*Institute for Theoretical Physics, University of Zürich,*
Winterthurerstr. 190, 8057 Zürich, Switzerland

[c]*Physics Department, New York City College of Technology, City University Of New York,*
300 Jay Street, Brooklyn NY 11201, U.S.A.

[d]*Nikhef, Science Park 105,*
1098XG Amsterdam, The Netherlands

[e]*Theory Group, Physics Department, CERN,*
CH-1211 Geneva 23, Switzerland

E-mail: gudrun.heinrich@durham.ac.uk, GOssola@citytech.cuny.edu,
thomasr@nikhef.nl, francesco.tramontano@cern.ch

ABSTRACT: We present a new approach to the reduction of one-loop amplitudes obtained by reconstructing the tensorial expression of the scattering amplitudes. The reconstruction is performed at the integrand level by means of a sampling in the integration momentum. There are several interesting applications of this novel method within existing techniques for the reduction of one-loop multi-leg amplitudes: to deal with numerically unstable points, such as in the vicinity of a vanishing Gram determinant; to allow for a sampling of the numerator function based on real values of the integration momentum; to optimize the numerical reduction in the case of long expressions for the numerator functions.

Contents

1   Introduction

In the last few years we observed enormous progress in the computation of one-loop virtual corrections for processes involving many particles.

Commonly, all these calculations rely on the decomposition of the tensor integrals in terms of scalar master integrals which are known analytically, and can be computed by means of public libraries, like LoopTools [1], QCDloop [2, 3], Golem 95 [4] or OneLOop [5, 6]. The task of reducing tensor integrals to scalar integrals can be achieved by following a fully analytic procedure, according to the well established Passarino-Veltman reduction [7, 8], which proved its effectiveness for reactions involving a small number of particles. Improved tensor reduction methods led to the development of tools [4, 9–11] which are able to deal in an efficient and numerically stable way with processes of high complexity such as the EW corrections to $e^+e^- \to 4f$ [12] and the QCD corrections to $pp \to t\bar{t}b\bar{b}$ [13–15], or $q\bar{q} \to b\bar{b}b\bar{b}$ [16].

In recent years, inspired by unitarity arguments [17, 18], alternative approaches have emerged, aiming at the direct determination of the coefficients of the master integrals. A novel powerful framework for one-loop calculations was developed by merging the idea

of exploiting the kinematic cuts of the scattering amplitudes in four dimensions [19, 20] with a systematic analysis of the structure of their integrands [21], leading to a framework by now known as OPP method [22, 23]. This approach has been further extended to the generalized d-dimensional unitarity-based method [24, 25] for the reconstruction of scattering amplitudes in dimensional regularization [26].

Different versions of the new reduction techniques have been implemented in various codes, such as CutTools [27] and SAMURAI [28], which are publicly available, BlackHat [29], and Rocket [30]. Among several results obtained with these new approaches [31–34], the state of the art is represented by the numerical calculation of extremely challenging $2 \to 4$ processes, like $pp \to W + 3$ jet production [35–38], $pp \to Z + 3$ jet production [39], $pp \to t\bar{t}b\bar{b}$ [40], $pp \to t\bar{t}jj$ [41], and $pp \to W^+W^+jj$ [42].

Controlling the quality of the numerical reconstruction is crucial for the new approaches to one-loop calculations [27, 28, 43]: with respect to the tensorial reduction, the new approaches require a particular attention to the issues of numerical efficiency and the control over numerical instabilities, which are some of the strong features of purely algebraic techniques. In addition to standard tests, commonly used within all the reduction methods to monitor the quality of the final results (such as the checks on the coefficients of the UV and IR poles or the comparison between results using different levels of numerical precision), the OPP/d-dimensional unitarity framework provides internal-consistency checks on the quality of the reconstructed coefficients. The phase space points for which the reconstruction fails the tests are in general (re)processed using routines that provide higher numerical precision which demand longer computing time. Multi-precision procedures are implemented either in dedicated routines or by doubling the original algorithm within a copy running in higher floating point precision.

In this paper we present a new simple strategy for the numerical evaluation of one-loop amplitudes, based on the reconstruction of the tensorial representation of their integrands. Accordingly, any integrand can be expressed in a basis of tensors formed by products of the integration momentum, each multiplied by a tensorial coefficient. The tensorial decomposition is achieved by sampling the integrand for *real* values of the components of the loop momentum. Finally, the reconstructed coefficients can be contracted numerically with the corresponding tensor integrals, evaluated by means of dedicated libraries such as Golem 95 [4].

This technique shows its virtues in at least three cases.

First, it is suitable to treat unstable configurations in the proximity of vanishing Gram determinants. The reduction of tensor integrals to express them in terms of scalar ones is responsible for the appearance of spurious, potentially singular coefficients, which might spoil the numerical accuracy in phase space regions where the Gram determinants tend to zero. Using a tensor basis, rather than a scalar one, this problem is bypassed completely. Although the evaluation of the tensor integrals demands more time than the computation of the scalar integrals, it might still be competitive when compared to a complete reduction in higher floating point precision. Moreover, a "rescue-system" based on the tensorial reconstruction of the integrand should be performed only for a small fraction of points when integrating over the full phase space.

Second, the tensorial reconstruction combined with the sampling by *real* momenta yields the automatic generation of the integrand available directly from tree-level generators like MadGraph [44] and HELAC [45, 46], which provide tree amplitudes for real values of external momenta. This possibility can be an alternative to the unitarity-based generation of the integrand, which, by construction, requires tree-level amplitudes evaluated at complex (yet on-shell) external momenta.

A third possibility is represented by the use of the reconstructed tensor integrand as "preprocessed" integrand for the reduction algorithm. Namely, the tensorial reconstruction of the integrand, by definition, amounts to disentangling the part of the integrand depending on the loop momentum from the one depending only on the kinematic variables. The dependence on the kinematics is carried by the tensorial coefficients, which have to be evaluated only once per phase-space point. Therefore, given any one-loop integrand, one can reconstruct its tensorial structure at a given phase-space point, and use the reconstructed expression as input for the reduction procedure at the integrand level. The numerical evaluation of the preprocessed integrand can be more effective than the evaluation of the original integrand, because the kinematic information stored in the tensorial coefficients is constant during the reduction algorithm, as it evolves simply through the variation of the loop-momentum used for the sampling. Within the framework of the preprocessed numerator, the tensorial coefficients play the role of an algebraic alternative to a caching system.

The paper is organized as follows. The reconstruction algorithm is discussed in section 2. Section 3 describes the various applications of the method, with particular emphasis on the "rescue-system" option. In this section we also describe an example of implementation by means of existing codes such as SAMURAI and Golem 95. Finally in section 4 we present our conclusions.

2 Tensorial reconstruction algorithm

2.1 Algorithm

Within the dimensional regularization scheme, any one-loop n-point amplitude can be written as

$$\mathcal{A}_n = \int d^d \bar{q} \frac{\mathcal{N}(\bar{q}, \epsilon)}{\bar{D}_0 \bar{D}_1 \cdots \bar{D}_{n-1}},$$

$$\bar{D}_i = (\bar{q} + p_i)^2 - m_i^2 = (q + p_i)^2 - m_i^2 - \mu^2. \tag{2.1}$$

We use a bar to denote objects living in $d = 4 - 2\epsilon$ dimensions, following the prescription $\bar{\slashed{q}} = \slashed{q} + \slashed{\mu}$ with $\bar{q}^2 = q^2 - \mu^2$. Further, we use the notation $f(\bar{q})$ as short-hand notation for $f(q, \mu^2)$.

Our task is to rewrite the numerator function $\mathcal{N}(\bar{q})$ as a linear combination of tensors of increasing rank, up to the maximum power of the integration momentum appearing in the numerator.

We first focus on the part of the numerator $\mathcal{N}(q)$ that depends only on the 4-dimensional part of the loop momentum q. Let's assume that the numerator has at most R powers in

the integration momentum q. We aim at building numerically the tensorial representation which reproduces $\mathcal{N}(q)$ before integration for each value of q. In the most general case, we have an expression of the form

$$\mathcal{N}(q) = \sum_{r=0}^{R} C_{\mu_1 \ldots \mu_r} q_{\mu_1} \ldots q_{\mu_r}, \tag{2.2}$$

where for each r the set of coefficients $C_{\mu_1 \ldots \mu_r}$ forms a contravariant tensor (we keep lower indices for convenience) and the contraction is performed with the Euclidean metric $(1, 1, 1, 1)$. For $r = 0$, we indicate the constant term as C_0. We observe that the reconstruction of eq. (2.2), namely the calculation of all coefficients $C_{\mu_1 \ldots \mu_r}$, is independent of the number of denominators $\bar{D}_i$ appearing in the original amplitude. It will however depend on the specific phase space point or helicity configuration that we want to process, in the same way as the numerator functions of other numerical methods.

In order to determine all the coefficients, we can simply evaluate both sides of eq. (2.2) for an arbitrary set of values of the integration momentum. Those values can be chosen as real four-momenta, thus allowing the treatment of numerators depending on a real integration momentum.

It is useful to rewrite $\mathcal{N}(q)$ by separating the tensorial components. Each of the terms in eq. (2.2) can be written as a multivariate polynomial in the components of q, where q_4 denotes the energy component

$$C_{\mu_1 \ldots \mu_r} q_{\mu_1} \cdots q_{\mu_r} = \sum_{(i_1, i_2, i_3, i_4) \vdash r} \hat{C}^{(r)}_{i_1 i_2 i_3 i_4} \cdot (q_1)^{i_1} (q_2)^{i_2} (q_3)^{i_3} (q_4)^{i_4}. \tag{2.3}$$

Here, the notation $\vdash$ indicates that the indices i_j have to form an integer partition of r.

We can now compute the coefficients $\hat{C}^{(r)}$; the conversion from $\hat{C} \mapsto C$ is easy since each component of $C_{\mu_1 \ldots \mu_r}$ contributes to one particular $\hat{C}^{(r)}_{i_1 \ldots i_4}$ where the tuple $(\mu_1, \ldots, \mu_r)$ contains exactly i_j occurrences of the number j. Conversely, when contracting the tensor $C_{\mu_1 \ldots \mu_r}$ with a tensor integral of rank r, one has to take into account that $\hat{C}^{(r)}_{i_1 \ldots i_4}$ already sums over

$$\frac{\left(\sum_{j=1}^{4} i_j \right)!}{\prod_{j=1}^{4} (i_j)!}$$

symmetric components of C. For both C and $\hat{C}^{(r)}$ the number of independent components in four dimensions is

$$n_r = \binom{4 + r - 1}{r} \tag{2.4}$$

and therefore $n_r = \{1, 4, 10, 20, 35, 56, 82\}$ for $r = 0, 1, \ldots 6$ respectively.

To reconstruct the tensorial coefficients of eq. (2.2) we will then consider the numerator for four-dimensional loop momentum q, $\mathcal{N}(q) = \mathcal{N}(x, y, z, w)$, as a multivariate polynomial of degree R in the four variables x, y, z and w, being the components of q, i.e. $q_\mu = (x, y, z, w)$. We will determine all the coefficients by sampling q, (hence its four components) in a bottom-up approach, starting from the simplest choice of q, namely the null-vector, and proceeding with sets of momenta q where one variable at a time is different from zero.

2.1.1 Level-0

The coefficient C_0 can be immediately determined from the identity:

$$\mathcal{N}(0,0,0,0) \equiv \mathcal{N}^{(0)} = C_0 \tag{2.5}$$

where $q = (0,0,0,0)$. This trivially completes the calculation of the constant term.

2.1.2 Level-1

We can subtract $\mathcal{N}^{(0)}$ from $\mathcal{N}(q)$, and define

$$\mathcal{N}^{(1)}(q) \equiv \mathcal{N}(q) - \mathcal{N}^{(0)} . \tag{2.6}$$

In this case we may consider four polynomial systems, generated respectively by choosing q with only one non-vanishing component, whose solution leads to the determination of $l_1 = 4R$ coefficients.

Using $q = (x,0,0,0)$, we extract only the R coefficients that are proportional to the first component of q. Therefore our system reduces to

$$\mathcal{N}^{(1)}(x,0,0,0) \equiv x\, C_1 + x^2\, C_{11} + \ldots + x^R\, C_{\underbrace{11\ldots1}_{R \text{ times}}}, \tag{2.7}$$

and can be easily solved by choosing R different values for x.

We repeat the same operation for $q = (0,y,0,0)$,

$$\mathcal{N}^{(1)}(0,y,0,0) \equiv y\, C_2 + y^2\, C_{22} + \ldots + y^R\, C_{\underbrace{22\ldots2}_{R \text{ times}}}, \tag{2.8}$$

and proceed analogously with the remaining two choices of q with only one component different from zero.

2.1.3 Level-2

After completing Level-0 and Level-1, we know $4R + 1$ coefficients, and we can again subtract the corresponding terms from $\mathcal{N}(q)$, to obtain

$$\mathcal{N}^{(2)}(q) \equiv \mathcal{N}^{(1)}(q) - \sum_{j=1}^{4} C_j\, q_j - \sum_{j=1}^{4} C_{jj}\, q_j^2 - \ldots - \sum_{j=1}^{4} C_{\underbrace{jj\ldots j}_{R \text{ times}}}\, q_j^R. \tag{2.9}$$

We proceed with the calculation of the $l_2 = 3R(R-1)$ coefficients multiplying two non-vanishing components of q. They will be found from six systems, each containing $R(R-1)/2$ coefficients.

We begin with $q = (x,y,0,0)$, and with this choice of q, $\mathcal{N}^{(2)}(q)$ reduces to:

$$\mathcal{N}^{(2)}(x,y,0,0) \equiv x\,y\,C_{12} + x^2\,y\,C_{112} + x\,y^2\,C_{122} + \ldots + \sum_{a,b:(a+b=R)} x^a\,y^b\,\hat{C}^{(R)}_{ab00}, \tag{2.10}$$

where a and b are different from zero. To extract the coefficients, we solve straightforwardly a system generated by sampling $R(R-1)/2$ pairs (x,y).

Then we choose $q = (x, 0, z, 0)$, and get the generating polynomial for the second system,

$$\mathcal{N}^{(2)}(x, 0, z, 0) \equiv x\, z\, C_{13} + x^2\, z\, C_{113} + x\, z^2\, C_{133} + \ldots + \sum_{a, b:(a+b=R)} x^a\, z^b\, \hat{C}^{(R)}_{a0b0}\,, \quad (2.11)$$

which yields again $R(R-1)/2$ coefficients. We repeat the same sampling algorithm for the remaining four cases.

2.1.4 Level-3

The first step of Level-3 is the subtraction of the known coefficients from $\mathcal{N}(q)$, defining the polynomial

$$\mathcal{N}^{(3)}(q) \equiv \mathcal{N}^{(2)}(q) - \sum_{r=2}^{R} \sum_{a, b:(a+b)=r} \sum_{k<j=1}^{4} C_{\underbrace{j\ldots j}_{a \text{ times}}\underbrace{k\ldots k}_{b \text{ times}}}\, q_j^a q_k^b \quad (2.12)$$

This time, we proceed with the calculation of the $l_3 = 2R(R-1)(R-2)/3$ coefficients by multiplying three non-vanishing components of q. They will be found from four systems, each containing $R(R-1)(R-2)/6$ coefficients.

We begin with $q = (x, y, z, 0)$, so that $\mathcal{N}^{(3)}(q)$ reduces to:

$$\mathcal{N}^{(3)}(x, y, z, 0) \equiv xyz\, C_{123} + x^2 yz\, C_{1123} + \ldots + \sum_{a, b, c:(a+b+c=R)} x^a y^b z^c\, \hat{C}^{(R)}_{abc0}\,, \quad (2.13)$$

where a, b and c are different from zero. By sampling $R(R-1)(R-2)/6$ triplets (x, y, z), we generate a system for the extraction of the coefficients. Analogous to level-2, we continue with the other three polynomials generated when $\mathcal{N}^{(3)}(q)$ is evaluated respectively for $q = (x, y, 0, w)$, $q = (x, 0, z, w)$, and $q = (0, y, z, w)$. The solutions of the associated systems complete the reconstruction of all coefficients multiplying three non-vanishing components of q.

2.1.5 Level-4

The final step is represented by the determination of the $l_4 = R(R-1)(R-2)(R-3)/24$ coefficients multiplying four non-vanishing components of q. They form the polynomial defined as

$$\mathcal{N}^{(4)}(q) \equiv \mathcal{N}^{(3)}(q) - \sum_{r=3}^{R} \sum_{a, b, c:(a+b+c)=r} \sum_{m<k<j=1}^{4} C_{\underbrace{j\ldots j}_{a \text{ times}}\underbrace{k\ldots k}_{b \text{ times}}\underbrace{m\ldots m}_{c \text{ times}}}\, q_j^a q_k^b q_m^c\,. \quad (2.14)$$

This polynomial is evaluated at $q = (x, y, z, w)$,

$$\mathcal{N}^{(4)}(x, y, z, w) \equiv xyzw\, C_{1234} + \ldots + \sum_{a, b, c, d:(a+b+c+d=R)} x^a y^b z^c w^d\, \hat{C}^{(R)}_{abcd}\,, \quad (2.15)$$

(a, b, c and d are different from zero) and sampled $R(R-1)(R-2)(R-3)/24$ times, to generate the system for the extraction on the last set of unknown coefficients.

At the end of Level-4, the tensorial reconstruction is complete: the original numerator $\mathcal{N}(q)$ is rewritten as a combination of tensors formed by products of loop momenta q. The tensorial coefficients multiplying each tensor store the information depending on the kinematics. We will denote the reconstructed numerator by $\langle N(q) \rangle$. For example, in the case of a numerator $\mathcal{N}(q)$ of rank six, the reconstruction of $\langle N(q) \rangle$ requires the calculation of 210 coefficients, obtained by evaluating $\mathcal{N}(q)$ 210 times. More details on the combinatorics are given in appendix A. We would like to point out that by construction the proposed algorithm does not introduce any spurious sources of instabilities.

As already mentioned, $\langle N(q) \rangle$ can be employed in several ways.

First, it can be used as *rescue system* to treat unstable configurations in the proximity of vanishing Gram determinants, because the use of a tensor basis, rather than a scalar one, avoids the appearance of spurious potentially singular coefficients.

Second, as $\langle N(q) \rangle$ can be constructed from products of trees evaluated at *real* momenta only, it matches well with the automatic generation of the integrand directly from available tree-level generators like MadGraph [44] and HELAC [45, 46], which provide tree amplitudes for real values of external momenta. Third, the reconstructed tensor integrand $\langle N(q) \rangle$ can be used as an integrand which is more suitable for further processing, as will be explained in detail in section 3.2.

2.2 Example of the reconstruction for rank two

If the numerator has at most two powers of q, eq. (2.2) reads

$$\mathcal{N}(q) = C_0 + C_\mu q^\mu + C_{\mu\nu} q^\mu q^\nu \, . \tag{2.16}$$

We have a total of 15 independent coefficients to determine, because $q_\mu q_\nu$ selects the symmetric part of $C_{\mu\nu}$. According to the algorithm outlined in the previous section, to reconstruct the unknown coefficient we have to solve the following systems.

- Level-0.

  We start from $\mathcal{N}(q)$ and evaluate it at the null-vector,

  $$\mathcal{N}(0,0,0,0) = \mathcal{N}^{(0)} = C_0 \, , \tag{2.17}$$

  obtaining trivially the constant C_0.

- Level-1.

  Subtracting the known term, we define

  $$\mathcal{N}^{(1)}(q) \equiv \mathcal{N}(q) - \mathcal{N}^{(0)} \, , \tag{2.18}$$

  and evaluate it for the following four choices of q,

  $$\mathcal{N}^{(1)}(x,0,0,0) = x\, C_1 + x^2\, C_{11} \, , \tag{2.19}$$

  $$\mathcal{N}^{(1)}(0,y,0,0) = y\, C_2 + y^2\, C_{22} \, , \tag{2.20}$$

  $$\mathcal{N}^{(1)}(0,0,z,0) = z\, C_3 + z^2\, C_{33} \, , \tag{2.21}$$

  $$\mathcal{N}^{(1)}(0,0,0,w) = w\, C_4 + w^2\, C_{44} \, . \tag{2.22}$$

Sampling each polynomial for two different *real* values of the variable, and solving the corresponding systems, yields the determination of the $l_1 = 4R = 8$ unknown coefficients C_i and C_{ii} with $i = 1, \dots, 4$.

- Level-2.

  We again subtract the determined coefficients, define

  $$\mathcal{N}^{(2)}(q) \equiv \mathcal{N}^{(1)}(q) - \sum_{j=1}^{4} C_j \ q_j - \sum_{j=1}^{4} C_{jj} \ q_j^2 \qquad (2.23)$$

  and evaluate it for the following six choices of q,

  $$\mathcal{N}^{(2)}(x, y, 0, 0) = xy \ C_{12}, \qquad (2.24)$$
  $$\mathcal{N}^{(2)}(x, 0, z, 0) = xz \ C_{13}, \qquad (2.25)$$
  $$\mathcal{N}^{(2)}(x, 0, 0, w) = xw \ C_{14}, \qquad (2.26)$$
  $$\mathcal{N}^{(2)}(0, y, z, 0) = yz \ C_{23}, \qquad (2.27)$$
  $$\mathcal{N}^{(2)}(0, y, 0, w) = yw \ C_{24}, \qquad (2.28)$$
  $$\mathcal{N}^{(2)}(0, 0, z, w) = zw \ C_{34}. \qquad (2.29)$$

  The extraction of the $l_2 = 3R(R-1) = 6$ coefficients is trivial in this case as well, because each monomial has to be sampled only once.

At the end of Level-2, we have found the numerical values of the 15 coefficients appearing in eq. (2.16).

2.3 Reconstruction of the μ^2-dependence

To account for the μ^2-dependence of the numerator $\mathcal{N}(\bar{q})$, where μ^2 is the radial integration variable in the $(d-4)$-dimensional subspace, the tensorial decomposition of the numerator requires more terms than in the 4-dimensional case previously discussed. The extended decomposition reads

$$\mathcal{N}(\bar{q}) = \langle N(q) \rangle + G^{(1)} \mu^2 + G^{(2)} \mu^4 + G^{(3)}_\alpha q^\alpha \mu^2 + G^{(4)}_{\alpha\beta} q^\alpha q^\beta \mu^2. \qquad (2.30)$$

This form can be derived from the fact that rational terms only come from the combination of $(d-4)$-dimensional terms with UV divergent integrals, and the listed numerators are the only ones leading to UV divergent integrals in addition to $\langle N(q) \rangle$ (in a renormalisable gauge). Note that the μ^2 terms can be inferred from the general relation

$$\int \frac{d^d q}{i\pi^{\frac{d}{2}}} (\mu^2)^\alpha f(q^\mu, q^2) = \frac{\Gamma(\alpha + \frac{d}{2} - 2)}{\Gamma(\frac{d}{2} - 2)} \int \frac{d^{d+2\alpha} q}{i\pi^{\frac{d}{2} + \alpha}} f(q^\mu, q^2). \qquad (2.31)$$

In calculating the terms proportional to both powers of q and μ^2, we should consider that not all terms will contribute to the final result. We will therefore limit our procedure to non-vanishing terms plus additional vanishing terms which are required to compute other pieces. For example, by power counting, we can exclude the presence of the term $G^{(2)} \mu^4$

from bubble and triangle diagrams. On the other hand, we cannot neglect the term $G^{(1)}\mu^2$ for box diagrams, even if we know that it will give a vanishing contribution: since we are operating at the integrand level, its presence is necessary in order to compute other non-vanishing terms in eq. (2.34), starting from $G^{(2)}$. Moreover, only the diagonal terms of $G^{(4)}$ in eq. (2.34) are needed, since the non-vanishing part of the resulting integral is proportional to $g^{\alpha\beta}$. Therefore, for n denominators, the numerator function $\mathcal{N}_n(\bar{q})$ of eq. (2.30) will reduce to:

$$\mathcal{N}_2(\bar{q}) = \langle N(q) \rangle + G^{(1)}\mu^2 \,, \tag{2.32}$$

$$\mathcal{N}_3(\bar{q}) = \langle N(q) \rangle + G^{(1)}\mu^2 + G^{(3)}_\alpha q^\alpha \mu^2 \,, \tag{2.33}$$

$$\mathcal{N}_4(\bar{q}) = \langle N(q) \rangle + G^{(1)}\mu^2 + G^{(2)}\mu^4 + G^{(3)}_\alpha q^\alpha \mu^2 + G^{(4)}_{\alpha\alpha} q^\alpha q^\alpha \mu^2 \,, \tag{2.34}$$

respectively.

The presence of μ^2 amounts to dealing with multivariate polynomials with one more variable than in the four-dimensional case. We can reconstruct all the coefficients $G^{(i)}$ by applying a technique similar to the one employed in the previous section, therefore, we will not repeat the discussion on the adopted sampling.

The integrals in d dimensions necessary for the evaluation of these terms have been given in several papers, see e.g. [47–50]; we list them here for completeness:

$$\int d^d\bar{q} \frac{\mu^2}{\bar{D}_i \bar{D}_j} = -\frac{i\pi^2}{2}\left[m_i^2 + m_j^2 - \frac{(p_i - p_j)^2}{3}\right] + \mathcal{O}(\epsilon)\,,$$

$$\int d^d\bar{q} \frac{\mu^2}{\bar{D}_i \bar{D}_j \bar{D}_k} = -\frac{i\pi^2}{2} + \mathcal{O}(\epsilon)\,,$$

$$\int d^d\bar{q} \frac{\mu^4}{\bar{D}_i \bar{D}_j \bar{D}_k \bar{D}_l} = -\frac{i\pi^2}{6} + \mathcal{O}(\epsilon)\,,$$

$$\int d^d\bar{q} \frac{\mu^2 q^\mu}{\bar{D}_i \bar{D}_j \bar{D}_k} = \frac{i\pi^2}{6}(p_i + p_j + p_k) + \mathcal{O}(\epsilon)\,,$$

$$\int d^d\bar{q} \frac{\mu^2 q^\mu q^\nu}{\bar{D}_i \bar{D}_j \bar{D}_k \bar{D}_l} = -\frac{i\pi^2}{12} g^{\mu\nu} + \mathcal{O}(\epsilon)\,. \tag{2.35}$$

Their multiplication by the coefficients $G^{(i)}$ completes the calculation of the rational part corresponding to the evaluation of R_2 in refs. [50–52]. After this step is done, we can set $\mu^2 = 0$ in all expressions and retain only the four-dimensional numerator in the remainder of the calculation.

3 Examples of applications

The tensorial reconstruction, paired with an efficient program for the evaluation of tensor integrals, can represent a simple and efficient way of computing one-loop virtual corrections. This method is particularly efficient for processes involving integrands of maximum rank equal to four: in this case the number of coefficients to compute is small and routines for the evaluation of tensor integrals very efficient. Important processes such as $pp \to b\bar{b}b\bar{b}$ or $pp \to t\bar{t}b\bar{b}$ belong to this category.
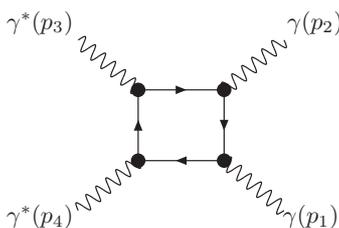
Figure 1. For the example in this section we use the above diagram in QED with two massless and two off-shell photons attached to a massless fermion loop.

In this section however we explore different uses of the tensorial decomposition, in particular the advantages that this method can bring when combined with other advanced approaches for the reduction of one-loop amplitudes, such as OPP/d-dimensional unitarity.

3.1 Dealing with unstable points

One of the major challenges for the new reduction methods, in particular when compared to algebraic reduction techniques, is to deal efficiently and automatically with phase space points which are numerically unstable. This is typically in the proximity of a vanishing Gram determinant.

All the points that do not pass the reconstruction/stability test within any chosen reduction algorithm can be reprocessed by using the technique presented here. The tensorial reconstruction avoids the reduction to scalar integrals and thus the emergence of Gram determinants; on the other hand, the technique requires the evaluation of tensor integrals, which is in general more time consuming.

In order to approach continuously a kinematic configuration that is numerically unstable, we consider a 4-point rank 4 diagram made of a fermion loop with two massless and two massive vector particles attached to it. We approach the phase space configuration of a vanishing Gram determinant by taking the limit $Q \to 0$ within the kinematics

$$p_{1,2} = (E, 0, 0, \pm E) \qquad\qquad p_{1,2}^2 = 0$$
$$p_{3,4} = (E, 0, \pm Q \sin\theta, \pm Q \cos\theta) \qquad p_{3,4}^2 = m^2$$

where $E = \sqrt{m^2 + Q^2}$ changes with Q, while θ and m^2 are kept constant (in the following plots we set $\theta = \frac{35}{180}\pi$ and $m^2 = 7$). The Gram determinant is given by $\det G = 32\, E^4 Q^2 \sin^2\theta$, while $\det S$, the modified Cayley determinant, goes to a constant as $Q \to 0$.

The calculation is performed with SAMURAI, using a standard d-dimensional integrand-level reduction; the results are compared to the improved tensorial technique where we employ `Golem 95` for the evaluation of the tensor integrals.

We present the case of a four-point function because this is typically the case in which the Gram determinant issue shows up. We would like to note that for kinematics far from a vanishing Gram determinant, the new method based on tensorial reconstruction and
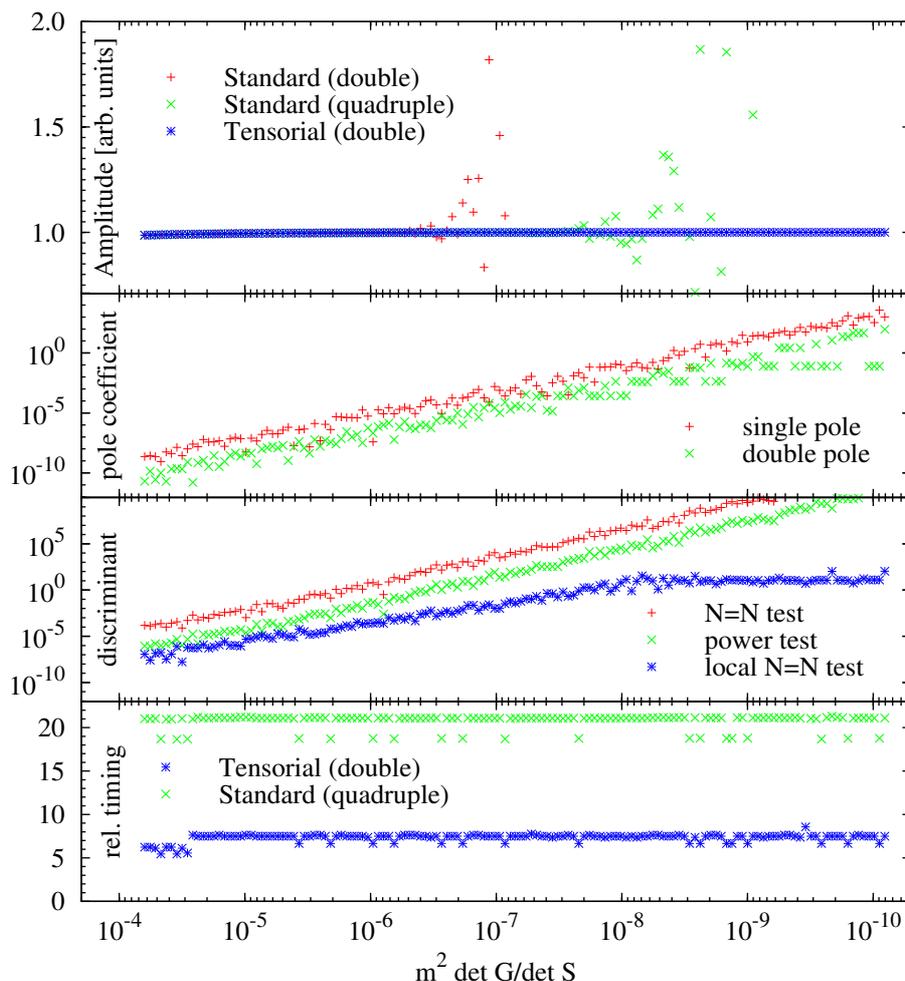
Figure 2. Top panel: Comparison between standard reduction at the integrand level (Standard) and tensorial reconstruction combined with an evaluation of the tensor integrals (Tensorial). The standard method starts deviating from the correct result at $\det G/\det S \approx 10^{-7}$, the standard method with quadruple precision (see the text) starts deviating at $\det G/\det S \approx 10^{-8}$, while the tensorial method remains stable over the whole range. Middle panels: the behaviour of tests that trigger the detection of instabilities. Bottom panel: timing of tensorial reduction in double precision versus standard reduction in quadruple precision, normalized by the timing of standard reduction in double precision.

SAMURAI have similar performances; however, for diagrams with more than four legs, the performance of SAMURAI with "standard" reduction at the integral level is naturally better in these "safe" phase space regions.

As shown in the top panel of figure 2, at a certain value of $\det G/\det S$, the standard reduction technique starts deviating from the stable result obtained by the improved tensor reconstruction, exhibiting the sensitivity to the vanishing Gram determinant.

The price to pay for the improved stability is an increased run time. The bottom panel of figure 2 shows the timing evaluated for the tensorial reconstruction method in double

precision, compared to an implementation in quadruple precision of the standard reduction at the integral level, each normalized to the standard reduction in double precision.

To make a comparison with quadruple or multiple precision we follow a strategy similar to the one proposed in [43]: we generate the kinematics in double precision, the scalar integral evaluation is performed in double precision as well. Then we upgrade formally the double precision kinematics to quadruple and go through the algorithm evaluating the numerator with multiple precision.

This can certainly give some benefit because near a configuration of vanishing Gram determinant the situation is such that numerator and denominator vanish, and with quadruple precision one can succeed to get a cancellation for small values of the Gram determinant. Such an improvement is evident from figure 2 (top panel); nevertheless, within this construction we do not find further improvements increasing the precision of the algorithm from quadruple to multiple. Hence, this implementation of quadruple precision only delays the Gram determinant problem, thus reducing the affected part of the phase space rather than curing the problem at its root. One might argue that a little improvement here would mean a big improvement in terms of statistics when extensive Monte Carlo runs are performed. On the other hand, in our implementation, examining the timing indicates that quadruple precision is more time consuming then tensorial reconstruction, at least for the example considered here.

The performance of the tensorial reconstruction relies on the underlying implementation of the tensor integrals: for this purpose we employed the `Golem 95` library. The timing depends on the value of $\left|\frac{\det G}{\det S}\right|$, as expected given the internal structure of the `Golem 95` library. When $\left|\frac{\det G}{\det S}\right|$ is above a certain threshold a fast, analytic reduction is used. For points where the ratio is below the threshold, the library switches to a method using the numerical quadrature of one-dimensional integral representations of the tensor integral form factors, thus avoiding the introduction of inverses of Gram determinants, but increasing the run time by up to a factor seven as compared to the standard reduction at integrand level, for this particular example.

Considering that the number of points where the standard approach fails in a cross-section calculation typically amounts to a few per mil, the additional cost is relatively small. Moreover, within the set of diagrams contributing to a specific process, only a subset will suffer from the instabilities and requires the additional evaluation time.

We have so far only discussed how the proposed method can cure numerical instabilities related to small Gram determinants, and will now elucidate strategies for an a priori detection of points requiring the application of this reduction technique. One would like to keep the fraction of points which are classified as unstable as low as possible to avoid switching unnecessarily to a slower method. In the remainder of this section we will study the performance in detecting such points for various reconstruction tests that have been proposed in the literature.

Cancellation of the poles. The cancellation of the infrared and ultraviolet poles is often used as a first indicator to identify points suffering from numerical instabilities. The panel below the top one of figure 2 shows that there is indeed a correlation between the

size of the Gram determinant and the deviation of the poles from the expected result, the latter being obtained from the tensorial method which gives a stable answer. However, one also observes a sizable spread in the values, which hampers the choice of a good threshold to discriminate unstable from stable points.

The "$N = N$" local test. The local test only examines the reconstructed polynomials corresponding to specific cuts. In the example we have plotted at each point the bubble cut which shows the maximum discriminant, the latter being defined as the relative difference between the reconstructed and the original polynomial. We find a steep slope in the region where the standard reduction method fails, however, the curve flattens out at smaller values of the Gram determinant.

The "$N = N$" global test. A stronger correlation can be found by looking at the global "$N = N$" reconstruction test. The idea behind this test is to compare the original numerator function to the reconstructed one for an arbitrary value of the integration momentum. The global test shows a clear correlation to the Gram determinant. By comparing the top plot of figure 2 with the benchmark from the "$N = N$" global test of figure 2, we can select an appropriate threshold: in this particular example, a threshold of $O(1)$ in the discriminant is sufficient for the classification of points as "unstable".

The *power test*. The so called power test was first presented in [28]. It uses the fact that in the reconstruction of the numerator one typically allows for a higher power of the integration momentum in the reconstructed function than one would expect in the original numerator function. One can therefore find certain combinations of coefficients which should sum to zero if the reconstruction was successful. Moreover, with respect to other reconstruction tests, the power test has the advantage of being totally independent from the choice of the integration momentum. Similar to the case of the global $N = N$ test we find a strong correlation between the goodness of these zeroes and the size of the Gram determinant and therefore find another trustworthy criterion for the discrimination of numerically unstable points in the presence of degenerate kinematics.

3.2 Improving the numerator sampling

Apart from dealing with numerical instabilities, even for stable phase-space points there are at least two possible applications of this tensorial reconstruction.

After the tensorial reconstruction is performed, we can feed the reconstructed numerator rather than the original one to the reduction algorithm. The added time needed to reconstruct the numerator can be compensated by a faster evaluation of the numerator during the reduction. For this reason, in the case of long expressions for the numerator function, processing the reconstructed tensor can be more efficient than the original numerator.

In order to address this issue, we performed a simple test: given a phase space point far from singularities, we consider a 6-point one-loop amplitude with a dummy numerator of rank R made of one line of code by means of sums and powers of various scalar products. We can increase linearly the complexity of the numerator by adding identical copies of the same line and monitor the run time for the evaluation of the amplitude with or without a

# Lines	Time ratio "hybrid"/standard	
N	Rank = 4	Rank = 6
1	1.3	1.6
10	1.1	1.4
100	0.51	0.85
1000	0.30	0.59
10000	0.27	0.55

Table 1. Timing for a rank 4 and rank 6 six-point function with a numerator of N lines. We display the ratio between the run times of the "hybrid" method and the standard reduction. In the "hybrid" method we compute the reconstructed tensor $\langle N(q) \rangle$ and use it for the reduction in place of the original numerator: already for a numerator of about $N = 100$ lines, we get a significant improvement in computation time.

preliminary reconstruction of the tensor integrand $\langle N(q) \rangle$. The results for the time ratio between the two methods for increasing sizes of the numerator are given in table 1. It is interesting to observe that, in the case of rank $R = 4$, we can easily gain a factor of 2 in the overall timing.

As a further application, the method allows for a sampling based on real values of the integration momentum: this feature can be used to facilitate the adaptation of tree-level generators to the task of producing one-loop numerator functions.

3.3 Implementation with SAMURAI and Golem 95

For all the examples and calculations contained in this paper, the new techniques have been implemented using SAMURAI for a d-dimensional integrand-level reduction, and Golem 95 for the evaluation of the tensor integrals. However the validity of the methods presented goes beyond our specific implementation and can be used in several alternative frameworks.

Coming back to our implementation, it required updates within SAMURAI which will be part of a future release of the package. In particular:

Rescue system: unstable points, detected by means of one of the available tests described above, will be automatically reprocessed using the tensorial decomposition. The evaluation of tensor integrals is performed by Golem 95.

Tensorial numerator: a flag will allow to reconstruct the tensorial integrand $\langle N(q) \rangle$, to be processed by the reduction algorithm in place of the original numerator. This is useful in the case of long expressions for the numerator function.

4 Conclusions and outlook

We presented a new approach to the reduction of one-loop amplitudes in which we reconstruct the tensorial expression of the scattering amplitudes by means of a sampling in the integration momentum at the integrand level.

This approach could be used within existing techniques for the reduction of one-loop multi-leg amplitudes in several ways; for example to deal with numerically unstable points, to allow for a sampling of the numerator function based on real values of the integration momentum, and to optimize the numerical reduction in the case of long expressions for the numerator functions.

We also illustrated how the existing reconstruction tests can be employed efficiently to detect potentially unstable phase space points. We believe that this method can represent a viable option as a "rescue-system" to deal with unstable phase space configurations: the increase in computation time required by the evaluation of tensorial "master" integrals appears to be less costly than reprocessing the unstable points with quadruple- or multi-precision routines.

The tensorial reconstruction has been implemented and successfully tested within SAMURAI and Golem 95 and will be part of the next release of these programs.

Acknowledgments

A Combinatorics for tensor components

The following table lists the block sizes and multiplicities of the systems for $d = 4$ up to rank 6. As explained in section 2, for rank r, at each level i, we have $l_i^r = \binom{4}{i}\binom{r}{i}$ coefficients. The numbers sum up to $n(R) = \sum_{i=0}^{R} l_i^r = \sum_{r=0}^{R} n_r$.

$$n(0) = 1 = 1 \times \binom{4}{0} \tag{A.1a}$$

$$n(1) = 5 = 1 \times \binom{4}{0} + 1 \times \binom{4}{1} \tag{A.1b}$$

$$n(2) = 15 = 1 \times \binom{4}{0} + 2 \times \binom{4}{1} + 1 \times \binom{4}{2} \tag{A.1c}$$

$$n(3) = 35 = 1 \times \binom{4}{0} + 3 \times \binom{4}{1} + 3 \times \binom{4}{2} + 1 \times \binom{4}{3} \tag{A.1d}$$

$$n(4) = 70 = 1 \times \binom{4}{0} + 4 \times \binom{4}{1} + 6 \times \binom{4}{2} + 4 \times \binom{4}{3} + 1 \times \binom{4}{4} \qquad \text{(A.1e)}$$

$$n(5) = 126 = 1 \times \binom{4}{0} + 5 \times \binom{4}{1} + 10 \times \binom{4}{2} + 10 \times \binom{4}{3} + 5 \times \binom{4}{4} \qquad \text{(A.1f)}$$

$$n(6) = 210 = 1 \times \binom{4}{0} + 6 \times \binom{4}{1} + 15 \times \binom{4}{2} + 20 \times \binom{4}{3} + 15 \times \binom{4}{4} \qquad \text{(A.1g)}$$

Hence, the largest system appears at rank six and is of size 20.

Open Access. This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

[1] T. Hahn, *Feynman diagram calculations with FeynArts, FormCalc and LoopTools*, arXiv:1006.2231 [SPIRES].

[2] R.K. Ellis and G. Zanderighi, *Scalar one-loop integrals for QCD*, *JHEP* **02** (2008) 002 [arXiv:0712.1851] [SPIRES].

[3] G.J. van Oldenborgh and J.A.M. Vermaseren, *New algorithms for one loop integrals*, *Z. Phys.* **C 46** (1990) 425 [SPIRES].

[4] T. Binoth, J.P. Guillet, G. Heinrich, E. Pilon and T. Reiter, *Golem95: a numerical program to calculate one-loop tensor integrals with up to six external legs*, *Comput. Phys. Commun.* **180** (2009) 2317 [arXiv:0810.0992] [SPIRES].

[5] A. van Hameren, C.G. Papadopoulos and R. Pittau, *Automated one-loop calculations: a proof of concept*, *JHEP* **09** (2009) 106 [arXiv:0903.4665] [SPIRES].

[6] A. van Hameren, *OneLoop: for the evaluation of one-loop scalar functions*, arXiv:1007.4716 [SPIRES].

[7] G. 't Hooft and M.J.G. Veltman, *Scalar one loop integrals*, *Nucl. Phys.* **B 153** (1979) 365 [SPIRES].

[8] G. Passarino and M.J.G. Veltman, *One loop corrections for e^+e^- annihilation into $\mu^+\mu^-$ in the Weinberg model*, *Nucl. Phys.* **B 160** (1979) 151 [SPIRES].

[9] T. Binoth, J.P. Guillet, G. Heinrich, E. Pilon and C. Schubert, *An algebraic/numerical formalism for one-loop multi-leg amplitudes*, *JHEP* **10** (2005) 015 [hep-ph/0504267] [SPIRES].

[10] A. Denner and S. Dittmaier, *Reduction schemes for one-loop tensor integrals*, *Nucl. Phys.* **B 734** (2006) 62 [hep-ph/0509141] [SPIRES].

[11] A. Denner and S. Dittmaier, *Scalar one-loop 4-point integrals*, arXiv:1005.2076 [SPIRES].

[12] A. Denner, S. Dittmaier, M. Roth and L.H. Wieders, *Electroweak corrections to charged-current $e^+e^- \to 4$ fermion processes: technical details and further results*, *Nucl. Phys.* **B 724** (2005) 247 [hep-ph/0505042] [SPIRES].

[13] A. Bredenstein, A. Denner, S. Dittmaier and S. Pozzorini, *NLO QCD corrections to $pp \to t\bar{t}b\bar{b} + X$ at the LHC*, *Phys. Rev. Lett.* **103** (2009) 012002 [arXiv:0905.0110] [SPIRES].

[14] A. Bredenstein, A. Denner, S. Dittmaier and S. Pozzorini, *NLO QCD corrections to top anti-top bottom anti-bottom production at the LHC: 2. Full hadronic results*, *JHEP* **03** (2010) 021 [`arXiv:1001.4006`] [SPIRES].

[15] A. Bredenstein, A. Denner, S. Dittmaier and S. Pozzorini, *Production of $t\bar{t}b\bar{b}$ at the LHC at NLO QCD*, `arXiv:1006.2653` [SPIRES].

[16] T. Binoth et al., *Next-to-leading order QCD corrections to $pp \to b\bar{b}b\bar{b} + X$ at the LHC: the quark induced case*, *Phys. Lett.* **B 685** (2010) 293 [`arXiv:0910.4379`] [SPIRES].

[17] Z. Bern, L.J. Dixon, D.C. Dunbar and D.A. Kosower, *One-loop n-point gauge theory amplitudes, unitarity and collinear limits*, *Nucl. Phys.* **B 425** (1994) 217 [`hep-ph/9403226`] [SPIRES].

[18] Z. Bern, L.J. Dixon, D.C. Dunbar and D.A. Kosower, *Fusing gauge theory tree amplitudes into loop amplitudes*, *Nucl. Phys.* **B 435** (1995) 59 [`hep-ph/9409265`] [SPIRES].

[19] R. Britto, F. Cachazo and B. Feng, *Generalized unitarity and one-loop amplitudes in $N = 4$ super-Yang-Mills*, *Nucl. Phys.* **B 725** (2005) 275 [`hep-th/0412103`] [SPIRES].

[20] Z. Bern, L.J. Dixon and D.A. Kosower, *On-shell methods in perturbative QCD*, *Annals Phys.* **322** (2007) 1587 [`arXiv:0704.2798`] [SPIRES].

[21] F. del Aguila and R. Pittau, *Recursive numerical calculus of one-loop tensor integrals*, *JHEP* **07** (2004) 017 [`hep-ph/0404120`] [SPIRES].

[22] G. Ossola, C.G. Papadopoulos and R. Pittau, *Reducing full one-loop amplitudes to scalar integrals at the integrand level*, *Nucl. Phys.* **B 763** (2007) 147 [`hep-ph/0609007`] [SPIRES].

[23] G. Ossola, C.G. Papadopoulos and R. Pittau, *Numerical evaluation of six-photon amplitudes*, *JHEP* **07** (2007) 085 [`arXiv:0704.1271`] [SPIRES].

[24] R.K. Ellis, W.T. Giele and Z. Kunszt, *A numerical unitarity formalism for evaluating one-loop amplitudes*, *JHEP* **03** (2008) 003 [`arXiv:0708.2398`] [SPIRES].

[25] W.T. Giele, Z. Kunszt and K. Melnikov, *Full one-loop amplitudes from tree amplitudes*, *JHEP* **04** (2008) 049 [`arXiv:0801.2237`] [SPIRES].

[26] R.K. Ellis, W.T. Giele, Z. Kunszt and K. Melnikov, *Masses, fermions and generalized D-dimensional unitarity*, *Nucl. Phys.* **B 822** (2009) 270 [`arXiv:0806.3467`] [SPIRES].

[27] G. Ossola, C.G. Papadopoulos and R. Pittau, *CutTools: a program implementing the OPP reduction method to compute one-loop amplitudes*, *JHEP* **03** (2008) 042 [`arXiv:0711.3596`] [SPIRES].

[28] P. Mastrolia, G. Ossola, T. Reiter and F. Tramontano, *Scattering amplitudes from unitarity-based reduction algorithm at the integrand-level*, *JHEP* **08** (2010) 080 [`arXiv:1006.0710`] [SPIRES].

[29] C.F. Berger et al., *An automated implementation of on-shell methods for one-loop amplitudes*, *Phys. Rev.* **D 78** (2008) 036003 [`arXiv:0803.4180`] [SPIRES].

[30] W.T. Giele and G. Zanderighi, *On the numerical evaluation of one-loop amplitudes: the gluonic case*, *JHEP* **06** (2008) 038 [`arXiv:0805.2152`] [SPIRES].

[31] T. Binoth, G. Ossola, C.G. Papadopoulos and R. Pittau, *NLO QCD corrections to tri-boson production*, *JHEP* **06** (2008) 082 [`arXiv:0804.0350`] [SPIRES].

[32] S. Actis, P. Mastrolia and G. Ossola, *NLO QED corrections to Hard-Bremsstrahlung emission in Bhabha scattering*, *Phys. Lett.* **B 682** (2010) 419 [`arXiv:0909.1750`] [SPIRES].

[33] K. Melnikov and M. Schulze, *NLO QCD corrections to top quark pair production in association with one hard jet at hadron colliders*, *Nucl. Phys.* **B 840** (2010) 129 [arXiv:1004.3284] [SPIRES].

[34] S. Biswas, K. Melnikov and M. Schulze, *Next-to-leading order QCD effects and the top quark mass measurements at the LHC*, *JHEP* **08** (2010) 048 [arXiv:1006.0910] [SPIRES].

[35] C.F. Berger et al., *Precise predictions for W + 3 jet production at hadron colliders*, *Phys. Rev. Lett.* **102** (2009) 222001 [arXiv:0902.2760] [SPIRES].

[36] C.F. Berger et al., *Next-to-leading order QCD predictions for W + 3-jet distributions at hadron colliders*, *Phys. Rev.* **D 80** (2009) 074036 [arXiv:0907.1984] [SPIRES].

[37] R.K. Ellis, K. Melnikov and G. Zanderighi, *Generalized unitarity at work: first NLO QCD results for hadronic W^+ 3jet production*, *JHEP* **04** (2009) 077 [arXiv:0901.4101] [SPIRES].

[38] R.K. Ellis, K. Melnikov and G. Zanderighi, *W + 3 jet production at the Tevatron*, *Phys. Rev.* **D 80** (2009) 094002 [arXiv:0906.1445] [SPIRES].

[39] C.F. Berger et al., *Next-to-leading order QCD predictions for $Z, \gamma^* + 3$-jet distributions at the Tevatron*, *Phys. Rev.* **D 82** (2010) 074002 [arXiv:1004.1659] [SPIRES].

[40] G. Bevilacqua, M. Czakon, C.G. Papadopoulos, R. Pittau and M. Worek, *Assault on the NLO wishlist: $pp \to ttbb$*, *JHEP* **09** (2009) 109 [arXiv:0907.4723] [SPIRES].

[41] G. Bevilacqua, M. Czakon, C.G. Papadopoulos and M. Worek, *Dominant QCD backgrounds in Higgs boson analyses at the LHC: a study of $pp \to t\bar{t} + 2$ jets at next-to-leading order*, *Phys. Rev. Lett.* **104** (2010) 162002 [arXiv:1002.4009] [SPIRES].

[42] T. Melia, K. Melnikov, R. Rontsch and G. Zanderighi, *Next-to-leading order QCD predictions for $W + W + jj$ production at the LHC*, arXiv:1007.5313 [SPIRES].

[43] R. Pittau, *Testing and improving the numerical accuracy of the NLO predictions*, *Comput. Phys. Commun.* **181** (2010) 1941 [arXiv:1006.3773] [SPIRES].

[44] J. Alwall et al., *MadGraph/MadEvent v4: the new web generation*, *JHEP* **09** (2007) 028 [arXiv:0706.2334] [SPIRES].

[45] A. Kanaki and C.G. Papadopoulos, *HELAC: a package to compute electroweak helicity amplitudes*, *Comput. Phys. Commun.* **132** (2000) 306 [hep-ph/0002082] [SPIRES].

[46] A. Cafarella, C.G. Papadopoulos and M. Worek, *Helac-Phegas: a generator for all parton level processes*, *Comput. Phys. Commun.* **180** (2009) 1941 [arXiv:0710.2427] [SPIRES].

[47] Z. Bern and A.G. Morgan, *Massive loop amplitudes from unitarity*, *Nucl. Phys.* **B 467** (1996) 479 [hep-ph/9511336] [SPIRES].

[48] A. van Hameren, J. Vollinga and S. Weinzierl, *Automated computation of one-loop integrals in massless theories*, *Eur. Phys. J.* **C 41** (2005) 361 [hep-ph/0502165] [SPIRES].

[49] T. Binoth, J.P. Guillet and G. Heinrich, *Algebraic evaluation of rational polynomials in one-loop amplitudes*, *JHEP* **02** (2007) 013 [hep-ph/0609054] [SPIRES].

[50] P. Draggiotis, M.V. Garzelli, C.G. Papadopoulos and R. Pittau, *Feynman rules for the rational part of the QCD 1-loop amplitudes*, *JHEP* **04** (2009) 072 [arXiv:0903.0356] [SPIRES].

[51] G. Ossola, C.G. Papadopoulos and R. Pittau, *On the rational terms of the one-loop amplitudes*, *JHEP* **05** (2008) 004 [arXiv:0802.1876] [SPIRES].

[52] M.V. Garzelli, I. Malamos and R. Pittau, *Feynman rules for the rational part of the electroweak 1-loop amplitudes*, *JHEP* **01** (2010) 040 [arXiv:0910.3130] [SPIRES].