hep-ph 1006.0679

DESY 10-073
HEPTOOLS 10-020
SFB/CPP-10-43

Some variations of the reduction of one-loop Feynman tensor integrals

Jochem Fleischer
Fakultät für Physik, Universität Bielefeld, Universitätsstr. 25, 33615 Bielefeld, Germany
E-mail: Fleischer@physik.uni-bielefeld.de

Tord Riemann*
Deutsches Elektronen-Synchrotron, DESY, Platanenallee 6, 15738 Zeuthen, Germany
E-mail: Tord.Riemann@desy.de

We present a new algorithm for the reduction of one-loop *tensor* Feynman integrals with $n \leq 4$ external legs to *scalar* Feynman integrals I_n^D with $n = 3,4$ legs in D dimensions, where $D = d + 2l$ with integer $l \geq 0$ and generic dimension $d = 4 - 2\varepsilon$, thus avoiding the appearance of inverse Gram determinants $()_4$. As long as $()_4 \neq 0$, the integrals $I_{3,4}^D$ with $D > d$ may be further expressed by the usual dimensionally regularized scalar functions $I_{2,3,4}^d$. The integrals I_4^D are known at $()_4 \equiv 0$, so that we may extend the numerics to small, non-vanishing $()_4$ by applying a dimensional recurrence relation. A numerical example is worked out. Together with a recursive reduction of 6- and 5-point functions, derived earlier, the calculational scheme allows a stabilized reduction of n-point functions with $n \leq 6$ at arbitrary phase space points. The algorithm is worked out explicitely for tensors of rank $R \leq n$.

13th International Workshop on Advanced Computing and Analysis Techniques in Physics Research, ACAT2010
February 22-27, 2010
Jaipur, India

*Speaker.

© Copyright owned by the author(s) under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike Licence.

arXiv:1006.0679v1 [hep-ph] 3 Jun 2010

1. Introduction

The efficient and stable evaluation of tensor Feynman integrals,

$$I_n^{\mu_1\cdots\mu_R} = C(\varepsilon)\int\frac{d^dk}{i\pi^{d/2}}\;\frac{\prod_{r=1}^R k^{\mu_r}}{\prod_{j=1}^n c_j^{\nu_j}}, \tag{1.1}$$

with denominators c_j, having *indices* ν_j and *chords* q_j,

$$c_j = (k-q_j)^2 - m_j^2 + i\varepsilon, \tag{1.2}$$

is an important ingredient of precision calculations for collider physics. The normalization $C(\varepsilon)$ plays a role for divergent integrals only and is conventional, $C(\varepsilon) = (\mu)^{2\varepsilon}\Gamma(1-2\varepsilon)/[\Gamma(1+\varepsilon)\Gamma^2(1-\varepsilon)]$. Here, we use the generic dimension $d = 4-2\varepsilon$ and $\mu = 1$.

For $n \leq 4$, the problem was basically solved in the Seventies of the last century [1, 2] by tensor reduction with an ansatz of linear equations in terms of scalar integrals, and their evaluation in terms of logarithms and dilogarithms. For predictions of massive particle production at the LHC or ILC, one needs multi-dimensional phase space integrals over typically hundreds to thousands of Feynman diagrams with $n \leq 6$ external legs and tensor ranks $R \leq n$. Further, the so-called ε-expansion is needed in higher-order calculations. Over the years, a variety of papers appeared, and a comprehensive survey is well beyond the scope of this contribution. One of the approaches is purely algebraic and rests on the representation of single tensor integrals by scalar integrals, where the latter are defined in higher dimensions $D = d+2l$ and may have also higher indices $\nu_j + m$ [3], e.g.:

$$I_n^{\mu} = \int\frac{d^dk}{i\pi^{d/2}}k^{\mu}\prod_{r=1}^n c_r^{-1} = -\sum_{i=1}^n q_i^{\mu}\, I_{n,i}^{d+2}, \tag{1.3}$$

$$I_n^{\mu\nu} = \int\frac{d^dk}{i\pi^{d/2}}k^{\mu}k^{\nu}\prod_{r=1}^n c_r^{-1} = \sum_{i,j=1}^n q_i^{\mu}q_j^{\nu}n_{ij}\, I_{n,ij}^{d+4} - \frac{1}{2}g^{\mu\nu}\, I_n^{d+2}, \tag{1.4}$$

$$I_n^{\mu\nu\lambda} = \int\frac{d^dk}{i\pi^{d/2}}k^{\mu}k^{\nu}k^{\lambda}\prod_{r=1}^n c_r^{-1} = -\sum_{i,j,k=1}^n q_i^{\mu}q_j^{\nu}q_k^{\lambda}\, n_{ijk}\, I_{n,ijk}^{d+6} + \frac{1}{2}\sum_{i=1}^n g^{[\mu\nu}q_i^{\lambda]}I_{n,i}^{d+4}, \tag{1.5}$$

$$\begin{aligned} I_n^{\mu\nu\lambda\rho} &= \int\frac{d^dk}{i\pi^{d/2}}k^{\mu}k^{\nu}k^{\lambda}k^{\rho}\prod_{r=1}^n c_r^{-1} \\ &= \sum_{i,j,k,l=1}^n q_i^{\mu}q_j^{\nu}q_k^{\lambda}q_l^{\rho}\, n_{ijkl}\, I_{n,ijkl}^{d+8} - \frac{1}{2}\sum_{i,j=1}^n g^{[\mu\nu}q_i^{\lambda}q_j^{\rho]}\, n_{ij}I_{n,ij}^{d+6} + \frac{1}{4}g^{[\mu\nu}g^{\lambda\rho]}I_n^{d+4}. \end{aligned} \tag{1.6}$$

Tensors like $g^{[\mu\nu}q_i^{\lambda]}$ are completely symmetrized, and we define $n_{ij} = \nu_{ij} = 1+\delta_{ij}$, $n_{ijk} = \nu_{ij}\nu_{ijk}$, $\nu_{ijk} = 1+\delta_{ik}+\delta_{jk}$ etc., and:

$$I_{p,ijk\cdots}^{D,stu\cdots} = \int\frac{d^Dk}{i\pi^{D/2}}\prod_{r=1}^n\frac{1}{c_r^{1+\delta_{ri}+\delta_{rj}+\delta_{rk}+\cdots-\delta_{rs}-\delta_{rt}-\delta_{ru}-\cdots}}. \tag{1.7}$$

The $I_{n-1,ab}^{\{\mu_1,\cdots\},s}$ e.g. is obtained from $I_n^{\{\mu_1,\cdots\}}$ by shrinking line s and raising the powers of inverse propagators a,b ($s\neq a,b$); $I_{n,ab}^{\{\mu_1,\cdots\},a} = I_{n,b}^{\{\mu_1,\cdots\}}$.

For one-loop tensor integrals, recurrence relations have been derived allowing to represent the scalar tensor coefficients $I^D_{n,\{a\}}$ in terms of scalar integrals in generic dimension d and with natural indices, usually $\nu_j = 1$ [4, 5]. In the present work we need the relation to reduce index j and dimension D simultaneously and another relation to reduce dimension D only:

$$()_n \nu_j \left(\mathbf{j}^+ I^D_n\right) = -\binom{j}{0}_5 I^{D-2}_n + \sum_{k=1}^{n} \binom{j}{k}_n \left(\mathbf{k}^- I^{D-2}_n\right), \tag{1.8}$$

$$()_n (D+1-\sum_{i=1}^{n} \nu_i) I^D_n = \binom{0}{0}_n I^{D-2}_n - \sum_{k=1}^{n} \binom{0}{k}_n \left(\mathbf{k}^- I^{D-2}_n\right). \tag{1.9}$$

These relations hold for arbitrary index sets $\{\nu_i\}$. The integrals $\mathbf{k}^- I^D_n$ and $\mathbf{j}^+ I^D_n$ are obtained from I^D_n by replacing $\nu_k \to (\nu_k - 1)$ and $\nu_j \to (\nu_j + 1)$, respectively. For the definitions of Gram determinant $()_n$ and signed minors like $\binom{j}{k}_n$ we refer to [6, 7, 8]. If the Gram determinant vanishes, $()_n = 0$, relation (1.9) allows to express $I^D_{n,\{a\}}$ by simpler integrals, some of them with less external legs:

$$I^D_{n,\{a\}}|_{()_n=0} = \frac{1}{\binom{0}{0}_n} \sum_{k=1}^{n} \binom{0}{k}_n \left(\mathbf{k}^- I^D_{n,\{a\}}\right). \tag{1.10}$$

All this works fine, but for tensors of rank R one gets an intermediate scalar basis with dimensions up to $D = d + 2R$, and finally relations in generic dimension d with coefficients $(1/()_n)^R$. If $()_n$ becomes small, and this happens during phase space integrations, numerical instabilities will appear.

From other approaches it is well-known that the appearance of powers of inverse Gram determinants for 5-point functions may be avoided completely, and for 4-point functions one has to apply special measures if needed.

In [7], we have demonstrated for pentagons up to $R = 3$ that a clever use of properties of signed minors allows to cancel all the inverse Gram derminants $()_5$. In [9], we derived a recursive algorithm for the representation of (n,R) tensors by $(n,R-1)$ tensors and $(n-1,R-1)$ tensors, although with appearance of inverse Gram determinants $()_n$.

Here, we describe an algorithm which combines both approaches and allows an efficient evaluation of the tensor integrals in terms of scalar functions; the latter may be evaluated by packages like FF [10], LoopTools/FF[11, 10], QCDloop/FF [12, 10].

The algorithm has been worked out until $(n,R) = (6,6)$ tensors, but it is evident how to go beyond that.

2. A sample reduction free of $1/()_5$: the 5-point tensor $I^{\mu\nu\lambda}_5$

For details we have to refer to [8] and references therein. As an example, we consider a rank $R = 3$ pentagon $I^{\mu\nu\lambda}_5$. The rank $R = 3$ tensor was treated already in [7], but the calculational method applied here may be more easily extended to higher ranks. We apply the recurrence derived in [13]:

$$I^{\mu\nu\lambda}_5 = I^{\mu\nu}_5 Q^\lambda_0 - \sum_{s=1}^{5} I^{\mu\nu,s}_4 Q^\lambda_s, \tag{2.1}$$

where the auxiliary vectors contain inverse Gram determinants:

$$Q^\mu_s = \sum_{i=1}^{5} q^\mu_i \frac{\binom{s}{i}_5}{()_5}, \quad s = 0,\ldots,5. \tag{2.2}$$

The vector ($R=1$) is free of the inverse Gram determinant $()_5$. Starting the general recursion [9], as next the rank $R=2$ tensor is expressed by scalar 4-point functions, free of inverse $()_5$:

$$I_5^{\mu\nu} = \sum_{i,j=1}^{4} q_i^\mu q_j^\nu \frac{1}{\binom{0}{0}_5} \sum_{s=1}^{5} \left[\binom{0i}{sj}_5 I_4^{d+2,s} + \binom{0s}{0j}_5 I_{4,i}^{d+2,s} \right] + g^{\mu\nu} \left[-\frac{1}{2} \frac{1}{\binom{0}{0}_5} \sum_{s=1}^{5} \binom{s}{0}_5 I_4^{d+2,s} \right]. \quad (2.3)$$

After further involved manipulations, one may arrive at an expression where also $I_5^{\mu\nu\lambda}$ is expressed by scalar 4-point functions with higher indices and in higher dimensions:

$$I_5^{\mu\nu\lambda} = \sum_{i,j,k=1}^{4} q_i^\mu q_j^\nu q_k^\lambda E_{ijk} + \sum_{k=1}^{4} g^{[\mu\nu} q_k^{\lambda]} E_{00k}, \quad (2.4)$$

$$E_{ijk} = -\frac{1}{\binom{0}{0}_5} \sum_{s=1}^{5} \left\{ \left[\binom{0j}{sk}_5 I_{4,i}^{d+4,s} + (i \leftrightarrow j) \right] + \binom{0s}{0k}_5 \nu_{ij} I_{4,ij}^{d+4,s} \right\}, \quad (2.5)$$

$$E_{00j} = \frac{1}{\binom{0}{0}_5} \sum_{s=1}^{5} \left[\frac{1}{2} \binom{0s}{0j}_5 I_4^{d+2,s} - \frac{d-1}{3} \binom{s}{j}_5 I_4^{d+4,s} \right]. \quad (2.6)$$

The presentation is evidently free of $1/()_5$, and it is more compact than that given in our earlier paper [7].

3. The 4-point scalars and their Gram determinants

We have now to express efficiently the following scalar functions:

$$I_4^{d+2}, I_4^{d+4}, I_{4,i}^{d}, I_{4,i}^{d+2}, I_{4,i}^{d+4}, I_{4,ij}^{d}, I_{4,ij}^{d+2}, I_{4,ij}^{d+4}, \ldots \quad (3.1)$$

We can again treat only an example. The recurrence relation (1.8) applies e.g. to the scalar function $I_{4,ijk}^{d+6}$, appearing as tensor coefficient in (1.6):

$$\nu_{ij}\nu_{ijk} I_{4,ijk}^{d+6} = -\frac{\binom{0}{k}_4}{()_4} \nu_{ij} I_{4,ij}^{d+4} + \sum_{t=1,t\neq i,j}^{4} \frac{\binom{t}{k}_4}{()_4} \nu_{ij} I_{3,ij}^{d+4,t} + \frac{\binom{i}{k}_4}{()_4} I_{4,j}^{d+4} + \frac{\binom{j}{k}_4}{()_4} I_{4,i}^{d+4}. \quad (3.2)$$

Here, we see the appearance of the inverse sub-Gram determinant $\binom{s}{s}_5 \equiv ()_4$. For every dimensional shift, another inverse power of it will appear. In contrast to the case $n=5$, this may not be completely prevented, but the following strategy is quite useful: *Restrict the appearance of* $()_4$ *to terms related to* I_4^{d+2l}*, where they may be made implicit, and hold the scalar integrals* I_3, I_2, I_1 *free of them.*

A lengthy calculation yields:

$$\begin{aligned}
\nu_{ij}\nu_{ijk} I_{4,ijk}^{d+6} = & -\frac{\binom{0}{i}}{\binom{0}{0}} \frac{\binom{0}{j}}{\binom{0}{0}} \frac{\binom{0}{k}}{\binom{0}{0}} (d-1)d(d+1) I_4^{d+6} - \frac{\binom{0i}{0j}\binom{0}{k} + \binom{0i}{0k}\binom{0}{j} + \binom{0j}{0k}\binom{0}{i}}{\binom{0}{0}^2} (d-1) I_4^{d+4} \\
& + \frac{\binom{0}{j}}{\binom{0}{0}} \frac{\binom{0}{k}}{\binom{0}{0}} \frac{(d-1)d}{\binom{0}{0}} \sum_{t=1}^{4} \binom{0t}{0i} I_3^{d+4,t} - \frac{\binom{0}{k}}{\binom{0}{0}} \frac{d-1}{\binom{0}{0}} \sum_{t=1}^{4} \binom{0t}{0j} I_{3,i}^{d+4,t} \\
& + \sum_{t=1}^{4} \frac{\binom{0i}{0k}\binom{0t}{0j} + \binom{0j}{0k}\binom{0t}{0i}}{\binom{0}{0}^2} I_3^{d+2,t} + \frac{1}{\binom{0}{0}} \sum_{t=1,t\neq i,j}^{4} \binom{0t}{0k} \nu_{ij} I_{3,ij}^{d+4,t}.
\end{aligned} \quad (3.3)$$

This expression is free of inverse Gram determinants $()_4$. Although, by repeatedly applying (1.9), the expression of I_4^{d+6}, I_4^{d+4} in terms of scalar integrals in generic dimension d will introduce unavoidably such terms:

$$I_4^{d+2l} = \left[\frac{\binom{0}{0}_4}{()_4} I_4^{d+2(l-1)} - \sum_{t=1}^{4} \frac{\binom{t}{0}_4}{()_4} I_3^{d+2(l-1),t} \right] \frac{1}{d+2l-3}. \tag{3.4}$$

The same recurrence relation allows to express these scalar functions I_4^{d+2l} in terms of simpler ones at $()_4 \equiv 0$, if one writes the recurrence for $I_4^{d+2(l+1)}$; see (1.10). One may, however, rewrite (1.9) at arbitrary $()_4$:

$$I_4^D = \frac{1}{\binom{0}{0}_4} \left[()_4 (D-3) I_4^{D+2} + \sum_{k=1}^{4} \binom{0}{k}_4 I_3^{D,k} \right], \tag{3.5}$$

and apply that relation for small $()_4$. The I_4^{D+2} will be evaluated (*approximately*) at $()_4 = 0$. This should give a better approximation than (1.10) for the higher-dimensional functions at small $()_4$, and may be even further iterated.

In the next section, we will study a numerical example.

4. Numerical example: D_{111}

In order to investigate the stability of the method near a typical kinematical point of vanishing sub-Gram determinant, we consider an example given in [14], namely the tensor integrals related to a certain box diagram, which in LoopTools [11, 15] notations is:

$$\mathrm{D0i}(\mathrm{id}, 0, 0, s_3, s_4, s_{12}, s_{23}, 0, M^2, 0, 0). \tag{4.1}$$

The Gram determinant is:

$$()_4 = \Delta^{(3)} = \mathrm{Det}\left(2K_iK_j\right), \tag{4.2}$$

where K_i are the internal momenta, expressible by the (incoming) external momenta p_i: $K_1 = p_1, K_2 = K_1 + p_2, K_3 = K_2 + p_3, K_4 = 0$. Then, with $p_i^2 = s_i, (p_i + p_j)^2 = s_{ij}$, we get:

$$()_4 = -2s_{12}\left[s_{23}^2 + s_3 s_4 - s_{23}(s_3 + s_4 - s_{12})\right]. \tag{4.3}$$

This Gram determinant vanishes if:

$$s_4 \to s_{crit} = s_{23} \frac{(s_{23} - s_3 + s_{12})}{(s_{23} - s_3)}. \tag{4.4}$$

For $s_{23} = 2 \times 10^4$ GeV^2, $s_3 = 1 \times 10^4$ GeV^2, $s_{12} = -4 \times 10^4$ GeV^2, it is $s_{crit} = -6 \times 10^4$ GeV^2, and we look at the dependence of the tensor coefficients on

$$x = \frac{s_4}{s_{crit}} - 1 \to 0. \tag{4.5}$$

x	$\mathfrak{Re}\, D_{111}^{HD}$	$\mathfrak{Im}\, D_{111}^{HD}$	$\mathfrak{Re}\, D_{111}^{LT}$	$\mathfrak{Im}\, D_{111}^{LT}$
0.0 [0]	–**3.15407250453** E-10	–**3.31837792633** E-10	–	–
10^{-15} [0]	–3.15407250450 E-10	–3.31837792634 E-10	–	–
10^{-10} [0]	–3.15407189016 E-10	–3.31837792176 E-10	–	–
10^{-8} [lin]	–3.15407250056 E-10	–3.31837790700 E-10	–	–
10^{-7} [lin]	–3.15407246361 E-10	–3.31837773302 E-10	–	–
10^{-6} [lin]	–3.15407198445 E-10	–3.31837599320 E-10	–	–
10^{-5} [lin]	–3.15405621398 E-10	–3.31835860039 E-10	–	–
5×10^{-5} [lin]	–3.15374472234 E-10	–3.31828141644 E-10	–	–
5×10^{-4}	–3.**1544**5603308 E-10	–3.**3169**7733987 E-10	–3.15444393358 E-10	–3.31694238268 E-10
7×10^{-4}	–3.**1540**3131303 E-10	–3.**316**83867119 E-10	–3.15408833562 E-10	–3.31674350685 E-10
10^{-3}	–**3.15374**819252 E-10	–**3.3164**1477212 E-10	–3.15374217567 E-10	–3.31639655233 E-10
10^{-2}	–**3.1500800**0183 E-10	–**3.2991592**0672 E-10	–3.15007998895 E-10	–3.29915924109 E-10
10^{-1}	–**3.11226750695** E-10	–**3.13582331976** E-10	–3.11226750694 E-10	–3.13582331977 E-10

Table 1: D_{111} calculated with the algebraic approach in comparison with LoopTools numerics. Both calculations agree in digits shown in boldface. Numbers labeled with [0] rely on (1.10) with $()_4 = 0$, those with [lin] rely on (3.5), the others on (3.4).

.

In these variables:

$$()_4 = -2\,x\,s_{23}s_{12}(s_{23} - s_3 + s_{12}). \tag{4.6}$$

In LoopTools conventions, the tensor coefficients D_{ijl} are defined as follows:

$$D_{\mu\nu\lambda} \;=\; \sum_{i,j,l=1}^{3} K_{i\mu}K_{j\nu}K_{l\lambda}D_{ijl} + \sum_{i=1}^{3}(g_{\mu\nu}K_{i\lambda} + g_{\nu\lambda}K_{i\mu} + g_{\lambda\mu}K_{i\nu})D_{00i}, \tag{4.7}$$

and for our conventions, see (1.5). The inverse propagators are $c_j = [(k-q_j)^2 - m_j^2] = [(k+K_{j-1})^2 - m_j^2]$. Because we assume in our formulae $q_4 = 0$, and in LoopTools it is $K_1 = 0$, one has to care about specific correspondences; it is e.g.:

$$D_{111} \;=\; n_{222}\; I_{4,222}^{d+6}, \tag{4.8}$$

with $n_{222} = \nu_{22}\nu_{222} = 6$, and $\nu_{ij}\nu_{ijk}I_{4,ijk}^{d+6}$ given in (3.3). In the example, the tensor coefficients are finite. For tensors with non-vanishing $1/\varepsilon^n$ terms, there may arise modifications due to different normalizations.

We present in table 1 and figure 1 the sample numerics for D_{111}, for $M = 91.1876$. Our numbers for $I_{4,222}^{d+6}$ are evaluated with a Mathematica notebook and for comparison we also used LoopTools v.2.4, both approaches in normal double precision.

When $x < 10^{-3}$, the table shows that both the numbers from LoopTools and from a combined use of (3.3) and (3.4) (in figure: red line) loose precision significantly. With (3.3) and (1.10) we may determine the exact value of D_{111} at $x = 0$, and then interpolate to rising x (red broken line). The broken blue line uses (3.3) and (3.5), and it extends the region of validity of the broken red line considerably, although one does not reach the main region, where the red line is numerically safe. A simple numerical interpolation would close the gap quite satisfactory because D_{111} is a smooth function of its arguments near $x = 0$. With much more effort, namely going further with additional iterations a la (3.5), one may reach the same. But, one should have in mind the scales involved, and if an accuracy of, e.g., three or four digits is sufficient, simple interpolation schemes will suffice.

Figure 1: The tensor coefficient D_{111}, defined in (4.7), evaluated with (4.8), as a function of x, defined in (4.5). Red line: evaluation of D_{111} with (3.3), using reductions (3.4); red broken line: use of (1.10); blue broken line: use of (3.5).

5. Summary

We gave an introduction to our purely algebraic approach to tensor reduction of one-loop Feynman integrals. The treatment of inverse sub-Gram determinants has been refined, and a case study for vanishing $()_4$ has been presented. For not too small Gram determinants $()_4$, the algorithm has been realized in the Fortran package OLOTIC [16], which follows the recursive approach [13] for tensor integrals $n \leq 6, R \leq n$. The OLOTIC is being made an open source package. A C++ package is under development [17] and will evaluate the tensor integrals according to the scheme described here.

Acknowledgments

Work supported in part by Sonderforschungsbereich/Transregio SFB/TRR 9 of DFG “Computergestützte Theoretische Teilchenphysik" and by the European Community’s Marie-Curie Research Training Network MRTN-CT-2006-035505 “HEPTOOLS”. We thank F. Campanario and V. Yundin for useful discussions. J.F. likes to thank DESY for kind hospitality.

References

[1] G. ’t Hooft and M. Veltman, *Scalar one loop integrals*, *Nucl. Phys.* **B153** (1979) 365–401.

[2] G. Passarino and M. Veltman, *One loop corrections for e^+e^- annihilation into $\mu^+\mu^-$ in the Weinberg model*, *Nucl. Phys.* **B160** (1979) 151.

[3] A. I. Davydychev, *A simple formula for reducing Feynman diagrams to scalar integrals*, *Phys. Lett.* **B263** (1991) 107–111.

[4] O. Tarasov, *Connection between Feynman integrals having different values of the space-time dimension*, *Phys. Rev.* **D54** (1996) 6479–6490, [hep-th/9606018].

[5] J. Fleischer, F. Jegerlehner, and O. Tarasov, *Algebraic reduction of one-loop Feynman graph amplitudes*, *Nucl. Phys.* **B566** (2000) 423–440, [hep-ph/9907327].

[6] D. B. Melrose, *Reduction of Feynman diagrams*, *Nuovo Cim.* **40** (1965) 181–213.

[7] T. Diakonidis, J. Fleischer, J. Gluza, K. Kajda, T. Riemann, and J. Tausk, *A complete reduction of one-loop tensor 5- and 6-point integrals*, *Phys. Rev.* **D80** (2009) 036003, [arXiv:0812.2134].

[8] T. Riemann, *Some variations of the reduction of one-loop Feynman tensor integrals*, Transparencies of talk at this conference.

[9] T. Diakonidis, J. Fleischer, T. Riemann, and J. B. Tausk, *A recursive reduction of tensor Feynman integrals*, *Phys. Lett.* **B683** (2010) 69–74, [arXiv:0907.2115].

[10] G. J. van Oldenborgh, *FF: A Package to evaluate one loop Feynman diagrams*, *Comput. Phys. Commun.* **66** (1991) 1–15.

[11] T. Hahn and M. Perez-Victoria, *Automatized one-loop calculations in four and d dimensions*, *Comput. Phys. Commun.* **118** (1999) 153, [hep-ph/9807565].

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

[13] T. Diakonidis, J. Fleischer, T. Riemann, and B. Tausk, *A recursive approach to the reduction of tensor Feynman integrals*, *PoS* **RADCOR2009** (2009) 033, [arXiv:1002.0529].

[14] A. Denner, *Techniques and concepts for higher order calculations*, Introductory Lecture at DESY Theory Workshop on Collider Phenomenology, Hamburg, 29 Sep - 2 Oct 2009.

[15] T. Hahn, *LoopTools 2.5 User's Guide*, LT25Guide.pdf.

[16] T. Diakonidis and B. Tausk. Fortran package OLOTIC.

[17] V. Yundin. C++ package for one-loop tensor reduction, unpublished.