Master integrals and generalized polylogarithms: focus on fast and efficient evaluation

Hjalte Frellesvig
*Institute of Nuclear Physics, NCSR Demokritos, Agia Paraskevi, 15310, Greece
E-mail: frellesvig@inp.demokritos.gr

Costas G. Papadopoulos
*Institute of Nuclear Physics, NCSR Demokritos, Agia Paraskevi, 15310, Greece
E-mail: costas.papadopoulos@cern.ch

Damiano Tommasini*
*Institute of Nuclear Physics, NCSR Demokritos, Agia Paraskevi, 15310, Greece
E-mail: tommasini@inp.demokritos.gr

Christopher Wever
*Institute of Nuclear Physics, NCSR Demokritos, Agia Paraskevi, 15310, Greece & Institute for Theoretical Particle Physics (TTP), Engesserstraße 7, D-76128 Karlsruhe & Institute for Nuclear Physics (IKP), Hermann-von-Helmholtz-Platz 1, D-76344 Eggenstein-Leopoldshafen, Karlsruhe Institute of Technology, Germany
E-mail: christopher.wever@kit.edu

In the LHC era, precision QCD calculations have become a necessity. After the OPP method has been developed, the computation of particle cross sections and physical observables at NLO accuracy has become automated in many cases. Unfortunately, for some processes NLO accuracy is not enough and NNLO corrections are needed. In the recent period, substantial improvements in understanding amplitude reduction at two loops and computation of the necessary two-loop master integrals has been achieved. There is a general consensus that in the (near) future automation of NNLO calculations will be feasible and then the problem of reasonable computational cost of two loops contribution to NNLO cross section will become relevant. In this seminar I will focus on general features in evaluating master integrals, with emphasis on evaluating master integrals in different phase space regions (with Euclidean or physical particle momenta). Particular emphasis will be given on fast evaluation of generalized polylogarithms, since many master integrals are expressed in terms of such special functions.

12th International Symposium on Radiative Corrections (Radcor 2015) and LoopFest XIV (Radiative Corrections for the LHC and Future Colliders)
15-19 June 2015
UCLA Department of Physics & Astronomy Los Angeles, CA, USA
1. Introduction

In the recent decades, the Standard Model of particle physics has been established as extremely successful. In particular the LHC accelerator have reached the record-setting high energies of 13 TeV and the experiments recorded large amount of data letting very precise measurements. In order to keep up with the increasing experimental accuracy as more data is collected, more precise theoretical predictions and higher loop calculations will be required.

With the better understanding of reduction of one-loop amplitudes to a set of Master Integrals based on unitarity methods [1, 2] and at the integrand level via the OPP method [3, 4], one-loop calculations have been fully automated in many numerical tools (some reviews on the topic are [5, 6]). In the recent years, a lot of progress has been made towards the extension of these reduction methods to the two-loop order at the integral [7, 8, 9] as well as the integrand [10, 11, 12, 13] level. Contrary to the master integrals at one-loop, which have been known for a long time already [14], a complete library at two-loops is still missing. At the moment this seems to be the main obstacle to obtain a fully automated NNLO calculation framework similar to the one-loop case, that will satisfy the anticipated precision requirements at the LHC [15].

At one-loop all Feynman integrals (in $4-2\varepsilon$ dimensions) are expressible in terms of the logarithm $\log(x)$ and the dilogarithm $\text{Li}_2(x)$, up to the zeroth order in the $\varepsilon$ expansion [16], and these functions are special cases of so called generalized (or Goncharov) polylogarithms (GPLs) [17, 18, 19, 20, 21, 22]. At two or more loops many Feynman integrals can be likewise expressed in terms of GPLs [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33] (for further references, see [34, 35, 36]), but there are also integrals which are counter examples, such as notably that of the fully massive sunset graph [37, 38]. Certain graphs without massive propagators are also believed to be counter examples [9]. In this talk we will restrict the discussion to GPLs.

In [39] it was conjectured that all GPLs up to weight four, which includes all GPLs needed for two-loop calculations, can be expressed in terms of logarithms, the classical polylogarithms $\text{Li}_n(x)$ ($n \leq 4$), and one extra special function denoted by $\text{Li}_{2,2}(x,y)$. In the same reference it was explicitly shown that the conjecture holds true for a subset of GPLs, denoted harmonic polylogarithms (HPLs), up to weight four. A number of physical calculations of two-loop Feynman integrals with several scales, i.e. [40, 41, 42, 43, 44] have hinted at the truth of that conjecture.

Since the logarithm and classical polylogarithm are well-known functions, efficient algorithms for their numerical evaluation have been widely studied and developed. On the other hand, the GiNaC implementation of Vollinga and Weinzierl [45] is the only publicly available program which can efficiently evaluate the special function $\text{Li}_{2,2}(x,y)$ for any set of complex arguments. In this proceeding we discuss an independent algorithm for the efficient evaluation of $\text{Li}_{2,2}(x,y)$.

2. Planar double boxes and computational timing

Let’s consider some of the quite complicate two-loops master integrals, the planar double boxes. The have been firstly calculated by [29, 30] (HMS) and later by [41] (GMT) in the so called
xyz-parametrization as shown in Fig. 1.

**Figure 1:** The xyz-parametrization of external momenta for the planar double boxes contributing to pair production at the LHC. All external momenta are incoming.

The Lorentz invariants are expressed as follows:

\[
\begin{align*}
S &= (q_1 + q_2)^2 = (q_3 + q_4)^2, \\
T &= (q_1 - q_3)^2 = (q_2 - q_4)^2, \\
U &= (q_1 - q_4)^2 = (q_2 - q_3)^2, \\
q_3^2 &= M_3^2, \\
q_4^2 &= M_4^2, \quad (2.1)
\end{align*}
\]

where for the xyz parametrization the external momenta are given by different relations according to the family

\[
\begin{align*}
P_{12}: & \quad p_1 = -q_3, \ p_2 = -q_4, \ p_3 = q_1, \ p_4 = q_2; \\
P_{13}: & \quad p_1 = -q_3, \ p_2 = q_1, \ p_3 = -q_4, \ p_4 = q_2; \\
P_{23}: & \quad p_1 = q_2, \ p_2 = -q_4, \ p_3 = -q_3, \ p_4 = q_1; \quad (2.2)
\end{align*}
\]

and

\[
\begin{align*}
S &= \bar{m}^2(1 + \bar{x})^2, \\
p_3^2 &= \bar{m}^2\bar{x}^2(1 - \bar{y}^2), \\
T &= -\bar{m}^2\bar{x}((1 + \bar{y})(1 + \bar{y}\bar{x}) - 2\bar{y}(1 + \bar{x})), \\
p_4^2 &= \bar{m}^2(1 - \bar{x}^2\bar{y}^2). \quad (2.3)
\end{align*}
\]

The double-box master integrals have been calculated by [31] (PTW) in the so called x-parametrization as shown in Fig. 2. In this case there is a unique parametrization for the external momenta

\[
\begin{align*}
q_1 &= xp_1; \ q_2 = xp_2; \ q_3 = p_{123} - xp_{12}; \ q_4 = -p_{1234} \\
p_1^2 &= 0; \ s_{12} := p_{12}^2; \ s_{23} := p_{23}^2; \ q := s_{123}^2 \quad (2.4)
\end{align*}
\]

and thus

\[
\begin{align*}
S &= s_{12}x^2, \\
T &= q - (s_{12} + s_{23})x, \\
M_3^2 &= (1 - x)(q - s_{12}x), \\
M_4^2 &= q. \quad (2.5)
\end{align*}
\]
Clearly, even if there are different parametrization, they describe the same problem and they are equivalent. Thus there is a mapping between the variables, that for example is in case of $q < s_{12}$

\begin{align}
P12-P13: & \quad \bar{x} = s_{12}x/q - 1, \quad \bar{y} = q(1-x)/(q - s_{12}x), \quad \bar{z} = (q - qx + s_{23}x)/(q - s_{12}x); \\
P23: & \quad \bar{x} = s_{12}x/q - 1, \quad \bar{y} = q(1-x)/(q - s_{12}x), \quad \bar{z} = (q - (s_{12} + s_{23})x)/(q - s_{12}x).
\end{align}

Using different parameterizations and different integration strategies, the obtained results looks differently, even if they are clearly equivalent. In order to better understand we can provide a sample of the results for the $P12-29$ master integral

\begin{align}
G^{(1; x)}_{s_{12}} \left( G^{(s_{12} + s_{23}; x)} \left( G^{0, q/s_{12}; x} + G^{0, q/(q-s_{23}); x} \right) \right) + \ldots \quad x\text{-param.} \quad (2.7) \\
G^{(-1; \bar{x})} \left( i\pi - G^{0; \bar{y}} + 2G^{0; \bar{z}} \right) + i\pi G^{(-1/\bar{x}; \bar{x})} - i\pi G^{(-1/\bar{z}; \bar{z})} + \ldots \quad xyz\text{-param.}
\end{align}

Different parametrization implies different computational costs once we require numerical evaluation. We can consider some “random” phase space points as Tab. 1 and a measure their computational timing for a master integral, as Tab. 2.

\begin{table}[h]
\centering
\begin{tabular}{ |l|l|l|l|l| }
\hline
 & S & T & $p_1^2$ & $p_2^2$ \\
\hline
phys1a & 5.678 & -0.243 & 0.017 & 5 \\
phys1b & 8 & -2 & 3 & 1 \\
phys1c & 130.05 & -20.85 & 66.7 & 5.5 \\
phys2a & 5.646 & -1.52 & 0.213 & 3 \\
phys2b & 3 & -0.6 & 0.5 & 1 \\
phys2c & 226.875 & -25.16 & 175.125 & 2.3 \\
\hline
\end{tabular}
\caption{Phase space points chosen to test the computational costs of some master integrals.}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{ |l|l|l|l|l|l| }
\hline
point & HMS & PTW & PTW+fib.bas. & GMT trad. & GMT opt. \\
\hline
phys1a & 0.44 & 1600 & 25 & 63 & 1.1 \\
phys1b & 12 & 300 & 1.5 & 4.2 & 0.41 \\
phys1c & 17 & 230 & 0.86 & 31 & 0.58 \\
phys2a & 0.72 & 520 & 3.2 & 47 & 0.55 \\
phys2b & 4.8 & 670 & 6.1 & 35 & 0.66 \\
phys2c & 17 & 240 & 0.88 & 42 & 0.69 \\
\hline
\end{tabular}
\caption{Evaluation time in sec. for the integral $P12-30$, i.e. the integral $G^{1111111m0}$ with seven propagators and not trivial numerator. Different columns refers to different libraries, as explained in the text.}
\end{table}

In Tab. 2 the HMS column refers to the solution by [29, 30], the next is by PTW [31] using the provided result. Notice this result is valid in the Euclidean region. It is possible to perform analytic continuation of the result to extend its validity in the physical region. If this is done at the numerical evaluation step by GiNaC the computational cost raise considerably. On the other hand the analytic continuation can be done by fibrationBasis command in the HyperInt package suitable [47] and then the computational time (column PTW+fib.bas.) become of the order
of HMS. The GMT trad. column is by [41]. Up to now all the solutions are expressed by GPLs up to weight four. The last column is obtained by using the optimized results published by [41] where all the results are expressed as conjectured in [39] in terms of logarithms, classical polylogarithms $\text{Li}_n(x)$ ($n \leq 4$), and $\text{Li}_{2,2}(x,y)$. The computational cost is typically much faster than in the previous cases. We can also notice that the computational time change dramatically not only by changing solution and but also physical point. Furthermore those changes appear unrelated each other, in cases. We can also notice that the computational time change dramatically not only by changing physical point may be fast using one solution and slow using another solution, but changing point the situation may be is the opposite.

In order to understand this feature we need to look at the \texttt{GiNaC} algorithm. GPLs are defined as nested integrals

$$ G(z_1, \ldots, z_k; y) = \int_0^y \frac{dt_1}{t_1 - z_1} \int_0^{t_1} \frac{dt_2}{t_2 - z_2} \cdots \int_0^{t_{k-1}} \frac{dt_k}{t_k - z_k}. \quad (2.8) $$

For $|y| \leq |z_j|$ for all $j$ and $y \neq z_1$, $G$ can be evaluated by sum definition

$$ G(z_1, \ldots, z_k; y) = \sum_{j_1=1}^{\infty} \cdots \sum_{j_k=1}^{\infty} \frac{1}{j_1 + \cdots + j_k} \left( \frac{y}{z_1} \right)^{j_1} \frac{1}{j_2 + \cdots + j_k} \left( \frac{y}{z_2} \right)^{j_2} \cdots \frac{1}{j_k} \left( \frac{y}{z_k} \right)^{j_k}. \quad (2.9) $$

If $|y| > |z_j|$ then the analytic continuation is performed by algebraic expressions. One of the simplest is $G(z, y) = G(y, z) - G(0, z) + i\pi G(0, y)$. In general a single GPL is expressed as combination of several that can be evaluated by nested sums

$$ G(\ldots z_i; \ldots y) = \sum(\ldots)G(\ldots). \quad (2.10) $$

Note that if $|y/z_j| \sim 1$, even if the series is formally convergent, it can be slow convergent series. In order to improve the computational speed, the Hölder convolution (or other strategies) can be applied

$$ G(z_1, \ldots, z_k; 1) = \sum_{j=0}^{k} (-1)^j G(1 - z_j, 1 - z_{j-1}, \ldots, 1 - z_1; 1/p)G(z_{j+1}, \ldots, z_k; 1/p). \quad (2.11) $$

In practice the evaluation of a single GPL may require iterated use of eqs. (2.10-2.11), leading to the evaluation of several GPLs. For example the numerical evaluation of $G(1,2,3;0.001)$ is order of hundred times faster than $G(1,2,3;1.999)$. This is explaining the large differences in the computational costs as Tab. 2. Furthermore the theoretical achievement obtained by the optimized library of [41], namely the reduction of GPLs of weight four to the previously mentioned basis of function can be spoiled by the use of Hölder convolution: the $\text{Li}_{2,2}(x,y)$ function is mapped back to generic GPLs

$$ \text{Li}_{2,2}(x,y) = G(0, 1/x, 0, 1/(xy); 1) = G(1 - 1/(xy), 1, 1/x, 1; 1/p) + \ldots \quad (2.12) $$

The main message is the following: once a library of two loops master integrals will be obtained, in order to have a fast and efficient numerical evaluation it is important to develop fast algorithms. In the next section we present a preliminary discussion about an optimized algorithm to evaluate the $\text{Li}_{2,2}(x,y)$ function.
3. Proposal for and algorithm to evaluate the $\text{Li}_{2,2}(x,y)$ function

The algorithm is going to be presented here is inspired by the algorithm Crandall [48] proposed to evaluate polylogarithms $\text{Li}_n(z)$. It is worthwhile to quickly summarize that algorithm before explaining the main idea.

The polylogarithm is evaluated with different formulas according to the value of the variable complex $z$.

- If $z = 1$ we just return the special value $\text{Li}_n(1) = \zeta(n)$,
- if $|z| \leq 1/2$ the polylogarithm is evaluated by the definition $\text{Li}_n(z) = \sum_{i=1}^{\infty} z^n/n^2$,
- if $|z| \geq 2$ we can apply inversion relation to map the problem to a convergent region by
  $$\text{Li}_n(1/z) = (-1)^n \left( -\text{Li}_n(z) + \frac{(2i\pi)^n}{n!} B_n(x) + 2i\Theta(z) \frac{\log^{n-1} z}{(n-1)!} \right)$$
  where $B_n(x)$ is the Bernoulli polynomial with $x = \frac{\log z}{2\pi}$ and $\Theta(z) = 1$ if ($\mathcal{R}(z) < 0$ or $\mathcal{R}(z) > 1$) and $\Theta(z) = 0$ otherwise;
- finally if $|z| \sim 1$ the series is slowly convergent, then we can use a logarithmic expansion of the polylogarithm that captures the singularity structure for $|z| \sim 1$ and is fast convergent
  $$\text{Li}_n(z) = \sum_{m=0, m \neq n}^{\infty} \frac{\zeta(n-m)}{m!} \log^m z + \frac{\log^{n-1} z}{(n-1)!} (H_{n-1} - \log(-\log z)). \quad (3.1)$$

Analogously the $\text{Li}_{2,2}(x,y)$ function can be evaluated\(^3\) by several expressions according to the numerical values of its complex arguments $(x,y)$.

As starting point, $\text{Li}_{2,2}$ can be calculated by the following sum

$$\text{Li}_{2,2}(x,y) = \sum_{i,j \geq 0, i \neq j} \frac{x^i y^j}{i^2 j^2} = \sum_{i=1, j=1}^{\infty} \frac{x^i}{(i+j)^2} \frac{(xy)^j}{j^2}, \quad (3.2)$$

which converges whenever $|x| \leq 1$ and $|xy| \leq 1$.

Outside the region of convergence of eq. (3.2) two relations are needed in order to map to the convergent region. One is obtained by the stuffle property [21, 35] and becomes

$$\text{Li}_{2,2}(x,y) = -\text{Li}_{2,2}(y,x) - \text{Li}_4(xy) + \text{Li}_2(x)\text{Li}_2(y), \quad (3.3)$$

and which is seen to effectively swap the two arguments. Furthermore we need inversion relation

$$\text{Li}_{2,2}(x,y) = \text{Li}_{2,2}(1/x, 1/y) - \text{Li}_4(xy) + 3 \left( \text{Li}_4(1/x) + \text{Li}_4(y) \right) + 2 \left( \text{Li}_3(1/x) - \text{Li}_3(y) \right) \log(-xy)
+ \text{Li}_2(1/x) \left( \frac{x^2}{6} + \frac{\log^2(-xy)}{2} \right) + 2 \text{Li}_2(y) \left( \log^2(-xy) - \log^2(-x) \right), \quad (3.4)$$

which is our generalization of the inversion relation for the case of $\text{Li}_n$, to $\text{Li}_{2,2}$. As for other similar relations, this inversion relation requires non-zero imaginary parts on $x$, $y$, and $xy$ in order to be guaranteed correct.

\(^3\)The full algorithm has been presented in a recent paper [49].
Table 3: A procedure for mapping $\text{Li}_{2,2}(x,y)$ to the convergent region. For the case of equalities both cases are in principle applicable.

The main algorithm is presented in Tab. 3. In principle it is enough to evaluate $\text{Li}_{2,2}(x,y)$ for any complex value of the variables, but can be quite slow convergent for $|x| \sim |y| \sim 1$. In order to improve the convergence of the sum in this problematic region, some further expressions are needed. In particular we can start from the one-dimensional integral

$$\text{Li}_{2,2}(x,y) = \int_0^1 \frac{\log(z) \text{Li}_{2}(xyz)}{z - 1/x} dz,$$

(3.5)

expand each term in the integrand and perform integration term by term. This is leading to some expressions that are generalization of eq. (3.1) and fast convergent. Finally we can plug the special value $\text{Li}_{2,2}(1,1) = \pi^4/120.$

Acknowledgments

The authors would like to thank Thomas Gehrmann, Andreas von Manteuffel, Erik Panzer, Narek Martirosyan, Yang Zhang, Francesco Coradeschi for useful discussions.

This work was primarily performed at NCSR Demokritos. CW and DT were primarily supported by the Research Funding Program ARISTEIA, HOCTools (co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF)). The work of HF is supported by the European Commission through the HiggsTools Initial Training Network PITN-GA-2012-316704.

References


Master integrals and generalized polylogarithms: focus on fast and efficient evaluation  

Damiano Tommasini


