Scattering Amplitudes via Computational Algebraic Geometry

Yang Zhang
(ETH Zürich)

yang.zhang@phys.ethz.ch

Introduction

The scattering amplitude is the central quantity for the prediction of particle scattering reactions, its modulus squared is directly proportional to the cross-section measured at a collider. To find signals of new particles or new fundamental interactions, the signal rate but in particular the background reactions have to be known to high precision. Run II of the Large Hadron Collider (LHC) puts urgent demands on scattering amplitude computations within Quantum Chromodynamics (QCD) and the Standard Model (SM).

Scattering amplitudes are traditionally calculated from Feynman diagrams. The method becomes arduous at the high-precision frontier, however, i.e. at multi-loop order. Some SM scattering amplitudes, crucial for LHC Run II, are presently uncomputable in the Feynman-diagram approach.

What makes multi-loop Feynman diagrams so difficult mathematically are the multiple complex variables and multifold integrals involved. The traditional one-variable complex analysis is not powerful enough for the computation of multi-loop Feynman diagrams. The most convenient mathematical tool for this purpose is Computational Algebraic Geometry (CAG). In recent years, several computational methods from CAG, such as Gröbner bases, syzygies, module computation, or genus evaluation, have been successfully applied to scattering amplitudes.

This article presents several typical applications of CAG in scattering amplitudes,

1. Multi-loop integrand reduction. We use Gröbner basis methods to obtain the minimal integrand automatically.

2. Multi-loop unitarity. Frequently, multi-loop unitarity generates degenerate multivariate residues, which cannot be reduced to one-variable residues. This problem can be solved by the transformation law and Bezoutian matrix methods.

3. Integration-by-parts (IBP) relations. We apply syzygy computations of polynomial modules to restrict the size of IBP relations, and efficiently reduce Feynman integrals to a minimal set.

Integrand Reduction

The integrals appearing in multi-loop Feynman diagrams are of the form

$$\int \frac{d^D\ell_1}{i\pi^{D/2}} \cdots \frac{d^D\ell_L}{i\pi^{D/2}} \frac{N(\ell_1,\ldots,\ell_L)}{D_1 \cdots D_k} \tag{1}$$

where L is the number of loops, ℓ_i are the loop momenta, and $D_i = p_i^2 - m_i^2$ are the denominators of Feynman propagators, where p_i and m_i are the momentum and mass of the particle running on the propagator and p_i is in general a linear combination of loop momenta ℓ_i and external momenta. The numerator N is a polynomial in the components of the ℓ_i.

A natural question on these objects is, *Is there a way to reduce the total number of terms before integration?* More specially, we need an algorithm to decompose the numerator as

$$N = \Delta + \sum_i f_i D_i \,, \tag{2}$$

so that all terms in the numerator which cancel denominators are identified, and the integrand is then *maximally* reduced. By maximal reduction we mean that the algorithm shall produce $\Delta = 0$ if N can be written as $N = \sum_i f_i D_i$.

For a one-loop amplitude, $L = 1$, the decomposition can be achieved by the OPP method [1, 2]. For $L > 1$, the naive generalization of OPP method does not work, however, because there are too many variables and maximal reduction cannot be achieved in general. Hence we introduce the CAG method *Integrand reduction via Gröbner basis*.

The decomposition (2) has the form of a synthetic polynomial division over the polynomials, and maximal reduction in particular is automatically satisfied by the properties of Gröbner basis. We define the ideal $I = \langle D_1, \ldots, D_k \rangle$, and calculate its Gröbner basis $G(I)$ with respect to some monomial order. By dividing the numerator N towards $G(I)$ we obtain the desired Δ and quotients f_i. In practice, we find it efficient to use GLex or GRevLex monomial orders. This algorithm [3, 4] works for an arbitrary number of loops and is realized in the Mathematica package BASISDET [3].

It is interesting to see the relation between *integrand reduction* and *unitarity*. In the context of a quantum

field theory, 'unitarity' means that a loop amplitude on the *unitarity cut* can be written as the product of lower-order loop and tree amplitudes. The unitarity cut is the affine algebraic set

$$V(I) = V(\{D_1, \ldots, D_k\}) \tag{3}$$

of solutions of $D_1 = \ldots = D_k = 0$. For a multi-loop amplitude it is typically very complicated to obtain the coefficients f_i in N from Feynman rules, so we would like to reconstruct N from unitarity, by merging lower-order loop and tree amplitudes. The question is, *How much information can be obtained from unitarity?* or more specifically, *What is the ansatz for the numerator from the unitarity cut?*

This question can be answered by integrand reduction via the Gröbner-basis method and Hilbert's Nullstellensatz [3]. Let $J = \sqrt{I}$ be the radical of I. Hilbert's Nullstellensatz states that if a polynomial g vanishes on the unitarity cut, i.e. vanishes on $V(I)$, then $g \in J$. Hence the unitarity-reconstructible part is the quotient ring, R/J. In particular we could prove [5] that in dimensional regularization scheme I is a radical ideal, $I = \sqrt{I}$, which means that, if we start from a numerator N with all possible terms restricted by renormalization conditions and indetermined coefficients, the division output Δ provides the ansatz for unitarity reconstruction.

Consider a simple example, the two-loop massless double box (dbox) diagram 1 in 4 dimensions. There are 7 inverse propagators $D_{1\ldots7}$,

$$\begin{array}{c} \ell_1^2, \quad (\ell_1 - p_1)^2, \quad (\ell_1 - p_1 - p_2)^2 \\ \ell_2^2, \quad (\ell_2 - p_4)^2, \quad (\ell_2 - p_3 - p_4)^2, \quad (\ell_1 + \ell_2)^2. \end{array} \tag{4}$$

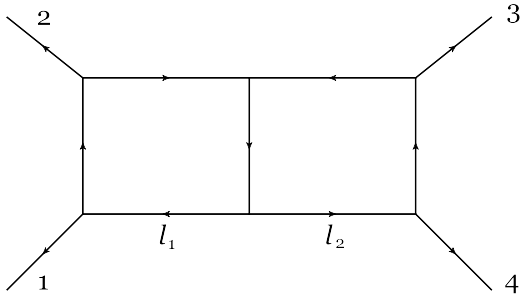

Figure 1: *Two-loop double box diagram.*

The ideal $I = \langle D_1, \ldots, D_7 \rangle$ is a one-dimensional ideal which is the intersection of 6 prime ideals. On rather general grounds (renormalizability) a term in the numerator of the form

$$(\ell_1^0)^{m_0}(\ell_1^1)^{m_1}(\ell_1^2)^{m_2}(\ell_1^3)^{m_3}(\ell_2^0)^{n_0}(\ell_2^1)^{n_1}(\ell_2^2)^{n_2}(\ell_2^3)^{n_3}$$

obeys $\sum_{i=0}^3 m_i \leq 4$, $\sum_{i=0}^3 n_i \leq 4$, $\sum_{i=0}^3 (m_i + n_i) \leq 6$. A simple counting shows that there are 160 monomials in N, and the division of N towards $G(I)$ generates a Δ with 32 monomials. Since parity-odd terms like $\epsilon_{\mu\nu\rho\sigma}\ell_i^\mu p_1^\nu p_2^\rho p_3^\sigma$ integrate to zero, we may drop another 16 monomials from the integrand and have hence reduced the original 160 terms in the numerator to 16 terms at the integrand level.

Several complicated multi-loop amplitudes were computed by this method. One distinguished achievement is the complete integrand for the two-loop all-plus-helicities five-gluon amplitude in QCD [5, 6, 7]. This integrand cannot be tackled in the traditional Feynman diagram approach with present compute resources, but it can be obtained efficiently from recursive integrand reduction method with Gröbner-basis computation.

Unitarity and Residue computation

Besides the unitarity-based integrand reduction we can also use unitarity in the residue approach [8, 9, 10]. An L-loop scattering amplitude can generally be decomposed into a basis $\{I_k\}$ of so-called master integrals (MI) and a remainder which is free of integrals,

$$A_n^{L\text{-loop}} = \sum_i c_i I_i + \text{rational terms}. \tag{5}$$

The *maximal unitarity method* introduces a further split,

$$\begin{array}{rl} A_n^{L\text{-loop}} = \sum_j c_j I_j & + \text{simpler integrals} \\ & + \text{rational terms} \end{array} \tag{6}$$

where the sum now extends only over master integrals with exactly k propagators, where k is the largest number of propagators in the MI basis. "Simpler integrals" stands for integrals with fewer than k propagators. The coefficients c_j can be obtained by maximal unitarity as follows: first replace a generic Feynman integral by a contour integral [8]

$$\begin{array}{l} \displaystyle\int \frac{d^D\ell_1}{(2\pi)^D} \cdots \frac{d^D\ell_L}{(2\pi)^D} \frac{N(\ell_1, \ldots, \ell_L)}{D_1^{\alpha_1} \cdots D_k^{\alpha_k}} \rightarrow \\ \displaystyle\oint \frac{d^D\ell_1}{(2\pi)^D} \cdots \frac{d^D\ell_L}{(2\pi)^D} \frac{N(\ell_1, \ldots, \ell_L)}{D_1^{\alpha_1} \cdots D_k^{\alpha_k}} = \sum_j w_j \oint_{\mathcal{C}_j} \omega \end{array}$$

where ω is a differential form on $V(I)$, and again $I = \langle D_1, \ldots, D_k \rangle$. The contours $\mathcal{C}_n$ are around the poles of ω and also around *nontrivial cycles* of V and w_n are weights of these contours.

To extract the coefficients c_j in (), we can find a special set of weights $w_n^{\{j\}}$ [8] such that

$$c_j = \sum_n w_n^{\{j\}} \oint_{\mathcal{C}_n} \omega\,. \tag{7}$$

This method obtains the coefficients in an elegant way, since usually it is much simpler to evaluate contour integrals than the original Feynman integrals. For multi-loop cases, however, the contour integrals are multivariate, and can be complicated in cases. Also here we can turn to CAG methods to evaluate the contour integrals.

A multivariate residue is mathematically defined as follows [22]: consider a residue at $(z_1, \ldots, z_n) = (\xi_1, \ldots, \xi_n) \equiv \xi$. Let U be the ball $||z - \xi|| < \epsilon$ and assume that the functions $f_1(z), \ldots, f_n(z)$ are holomorphic in U and have only one isolated common zero ξ in

U. Let $h(z)$ be a holomorphic function in a neighborhood of $\bar{U}$. Then for the differential form

$$\omega = \frac{h(z)dz_1 \wedge \cdots \wedge dz_n}{f_1(z)\cdots f_n(z)} \tag{8}$$

the residue at ξ is defined to be

$$\mathrm{Res}_{\{f_1,\ldots,f_n\},\xi}(\omega) = \left(\frac{1}{2\pi i}\right)^n \oint_\Gamma \omega\,. \tag{9}$$

The contour is defined by the real n-cycle, $\Gamma = \{z : |f_i(z)| = \epsilon_i\}$, with orientation specified by $d(\arg f_1) \wedge \cdots \wedge d(\arg f_n)$. If the Jacobian of $\{f_1,\ldots,f_n\}$ at ξ is nonzero, we call the residue *non-degenerate*. In this case, the value of residue is simply

$$\mathrm{Res}_{\{f_1,\ldots,f_n\},\xi}(\omega) = \frac{h(\xi)}{J(\xi)}. \tag{10}$$

by Cauchy's theorem. For multi-loop amplitudes, the residues can be degenerate, however, so Cauchy's theorem does not apply.

There are many different algorithms of calculating multivariate residues in algebraic geometry. We mainly apply two methods,

1. Transformation law. The original denominators can be transformed to a new set of denominators, namely, $h_i = \sum_j a_{ij} f_j$, where a_{ij} are locally holomorphic functions. Then the transformation law states that

$$\mathrm{Res}_{\{f_1,\ldots,f_n\},\xi}(\omega) = \mathrm{Res}_{\{g_1,\ldots,g_n\},\xi}(\det A\omega) \tag{11}$$

where A is the matrix (a_{ij}). In particular, if $f_1,\ldots,f_n$ are polynomials, we can calculate the Gröbner basis for $I = \langle f_1,\ldots f_n\rangle$ in Lex or elimination orders to obtain a set of polynomials h_i where each h_i is univariate, $h_i(z) = h_i(z_i)$. The r.h.s. of (11) then becomes a product of univariate residues.

2. Bezoutian matrix. The multivariate residue has local and global dualities. Define symmetric forms

$$\langle N_1, N_2\rangle_\xi \equiv \mathrm{Res}_{\{f_1,\ldots,f_n\},\xi}\left(\frac{N_1 N_2}{f_1\cdots f_n}\right), \tag{12}$$

$$\langle N_1, N_2\rangle \equiv \sum_{\xi\in V(I)} \langle N_1, N_2\rangle_\xi\,. \tag{13}$$

Eq. (13) is non-degenerate in $\mathbb{C}[z_1,\ldots,z_n]/I$, so we can find a $\mathbb{C}$-basis $\{e_1,\ldots e_d\}$ by Gröbner-basis computation. Then by duality theorems, there is a dual basis $\{\Delta_1,\ldots,\Delta_n\}$ such that

$$\langle e_i, \Delta_j\rangle = \delta_{ij} \tag{14}$$

Explicitly, the dual basis can be found as follows: first, calculate the Bezoutian matrix $B = (b_{ij})$,

$$b_{ij} = \frac{f_i(y_1,\ldots,y_{j-1},z_j,\ldots,z_n)}{z_j - y_j} - \frac{f_i(y_1,\ldots,y_j,z_{j+1},\ldots,z_n)}{z_j - y_j}\,. \tag{15}$$

Then we divide the determinant $\det B$ over the double copy of the Gröbner basis $G(I)$. The remainder can be expanded as

$$\sum_{i=1}^{d} \Delta_i(y) e_i(z), \tag{16}$$

hence both the basis and dual basis are obtained. The sum of residues can be easily found by standard linear algebra analysis of the non-degenerate inner product.

Finally, for each individual residue ξ we can find a polynomial s_ξ with properties

$$\sum_{\xi\in V(I)} s_\xi = 1, \quad s_\xi^2 = s_\xi, \quad s_{\xi_i} s_{\xi_j} = 0\ (i\neq j) \tag{17}$$

by means of which the individual residue is extracted from the sum of all residues,

$$\mathrm{Res}_{\{f_1,\ldots,f_n\},\xi_i}(\omega) = \sum_{\xi\in V(I)} \mathrm{Res}_{\{f_1,\ldots,f_n\},\xi}(s_\xi\omega). \tag{18}$$

This method is highly efficient for calculating the sum of residues. For an individual residue, it is also often faster than the first method since one does not need to compute a Gröbner basis in Lex or Elimination order.

For example, using the first method, we calculated all 64 residues from the maximal unitarity cut of a three-loop triple box diagram [11]. All residues are bivariate and 6 of them are degenerate. We also used both methods to get the two-loop unitarity cuts with doubled propagators [12, 13]. In these cases, all residues are degenerate.

The residue computation methods also apply to the computation of scattering equations [19]. In particular, the scattering equation requires the sum of residues only, so we can apply the highly efficient method 2 to get the amplitude directly.

IBP reduction and Syzygy computation

Integration-by-parts (IBP) identities [14, 15] arise from the vanishing integration of total derivatives. Schematically, the relations take the form

$$\int \prod_{i=1}^{L} \frac{d^D\ell_i}{\pi^{D/2}} \sum_{j=1}^{L} \frac{\partial}{\partial \ell_j^\mu} \frac{v_j^\mu\, P}{D_1^{a_1}\cdots D_t^{a_k}} = 0\,, \tag{19}$$

where P and the vectors v_j^μ are polynomials in the internal and external momenta, the D_k denote inverse propagators, and $a_i \geq 1$ are integers. IBP relations reduce the set of all integrals to master integrals.

In the computation of multi-loop scattering amplitudes, IBP reduction is a necessary but difficult step. The difficulty comes from the large number of choices for v_i^μ: there are too many IBP relations and too many integrals involved. After obtaining IBP relations, we need to use a linear reduction to find an independent set of IBPs. This process usually takes a lot of computer time and memory. The current standard IBP-generating algorithm is the one of Laporta [16], of which several public implementations exist.

One way to improve the IBP-generating efficiency is pick up suitable v_i^μ such that (19) contains no doubled propagator [17]. Overall, this choice is equivalent to finding *polynomial tangent fields of a given affine hypersurface*, or *Kähler differentials*.

We briefly review the *polynomial tangent field*. Let $F(z)$ be a polynomial from $R = \mathbb{C}[z_1, \ldots, z_n]$. The set of polynomial tangent fields is the submodule M_f in R^n such that, for each $(a_1, \ldots, a_n) \in M$,

$$\sum_{i=1}^{n} a_i \frac{\partial F}{\partial z_i} = b(z) F(z) \tag{20}$$

for some polynomial $b(z)$. This definition is dual to the *Kähler differential*. If the surface $F(z) = 0$ is nonsingular, the tangent field is in general simple. For example, let F be $y^2 - x^3 - 1$, then $F(z) = 0$ is a smooth curve and the singular ideal is $\langle 1 \rangle$. From the analysis of the singular ideal we easily find that the polynomial tangent fields are generated by

$$(-F_y, F_x), \quad (F, 0), \quad (0, F). \tag{21}$$

On the other hand, if $F = 0$ is singular, then it is more difficult to find the polynomial tangent vector fields. For example, let $F = y^2 - x^3$. There is a singular point at $(0, 0)$. From the weighted Euler's homogeneous theorem, the polynomial tangent fields are generated by

$$(-F_y, F_x), \quad (F, 0), \quad (0, F), \quad (\tfrac{1}{3}x, \tfrac{1}{2}y), \tag{22}$$

where the last generator is from the weights of variables around the singular point. In general cases, when there is more than one singular point, it is not easy to derive polynomial tangent fields. We can then treat (20) as a *syzygy* equation for a_i and b and solve it by CAG methods.

Back to the IBP problem, it is convenient to convert the integrand to Baikov form. For example, for a two-loop integral with $n \geq 5$ external legs,

$$I^{(2)}_{n\geq 5} = \frac{2^{D-6}}{\pi^5 \Gamma(D-5) J} \int \prod_{i=1}^{11} dz_i \, F(z)^{\frac{D-7}{2}} \frac{P(z)}{z_1 \cdots z_k}, \tag{23}$$

while for a two-loop amplitude with $n \leq 4$ legs,

$$I^{(2)}_{4} = \frac{2^{D-5}}{\pi^4 \Gamma(D-4) J} \int \prod_{i=1}^{9} dz_i \, F(z)^{\frac{D-6}{2}} \frac{P(z)}{z_1 \cdots z_k}. \tag{24}$$

The advantage of this form is that the propagators D_i have all become linear monomials z_i. The trick is now to apply unitarity, $D_i^{-1} \to \delta(D_i)$. We consider a c-fold cut ($0 \leq c \leq k$) and let $\mathcal{S}_{\rm cut}$, $\mathcal{S}_{\rm uncut}$ and $\mathcal{S}_{\rm IBP}$ denote the sets of indices labelling cut propagators, uncut propagators, and IBPs, respectively. $\mathcal{S}_{\rm cut}$ thus contains c elements. Moreover, we let m denote the total number of z variables, and set $\mathcal{S}_{\rm uncut} = \{r_1, \ldots, r_{k-c}\}$ and $\mathcal{S}_{\rm IBP} = \{r_{k-c+1}, \ldots, r_{m-c}\}$. Then, by cutting the propagators, $z_i^{-1} \to \delta(z_i), i \in \mathcal{S}_{\rm cut}$, the integrals (23) and (24) reduce to

$$I^{(2)}_{\rm cut} = \int \frac{dz_{r_1} \cdots dz_{r_{m-c}} P(z)}{z_{r_1} \cdots z_{r_{k-c}}} F(z)^{\frac{D-h}{2}} \Big|_{z_i = 0, \forall i \in \mathcal{S}_{\rm cut}}, \tag{25}$$

where h is a constant which depends on the number of external legs: $h = 6$ for $n = 4$ and $h = 7$ for $n \geq 5$.

We make the following IBP ansatz:

$$0 = \int d\Bigg(\sum_{i=1}^{m-c} \frac{(-1)^{i+1} a_{r_i} F(z)^{\frac{D-h}{2}}}{z_{r_1} \cdots z_{r_{k-c}}} dz_{r_1} \wedge \cdots \widehat{dz_{r_i}} \cdots \wedge dz_{r_{m-c}} \Bigg). \tag{26}$$

By requiring that the resulting IBP have no dimensional shift or doubled poles, we obtain the syzygy equations

$$bF + \sum_{i=1}^{m-c} a_{r_i} \frac{\partial F}{\partial z_{r_i}} = 0, \tag{27}$$

$$a_{r_i} + b_{r_i} z_{r_i} = 0, \quad i = 1, \ldots, k-c, \tag{28}$$

where a_{r_i}, b and b_{r_i} must be polynomials in z_j. Note that the last $(k-c)$ equations in (28) are trivial since they are solved by $a_{r_i} = -b_{r_i} z_{r_i}$. We therefore only have one syzygy equation to solve,

$$bF - \sum_{i=1}^{k-c} b_{r_i} z_{r_i} \frac{\partial F}{\partial z_{r_i}} + \sum_{j=k-c+1}^{m-c} a_{r_j} \frac{\partial F}{\partial z_{r_j}} = 0 \tag{29}$$

for $m - c + 1$ polynomials b_{r_i}, a_{r_i} and b. The geometric meaning of this equation is this: we are looking for *polynomial tangent vectors* for the reducible hypersurface defined by

$$F(z) \cdot \prod_{i=1}^{k-c} z_i = 0 \tag{30}$$

We solve Eq. (29) by the CAG software Macaulay2 [20] or Singular [21].

After the IBPs from different cuts are obtained, we can reconstruct the complete IBP relations. We realize this algorithm [18] in a Mathematica package with the communication with Singular [21]. It analytically reduces all double box integrals with numerator rank ≤ 4 to the eight master integrals in about 39 seconds in the fully massless case, and to the 19 master integrals in about 211 seconds in the one-massive-particle case (on a laptop with 2.5 GHz Intel Core i7 and 16 GB RAM).

A more advanced method [23] is to consider (27) and (28) separately. Eqs. (28) can be solved by hand,

since these are trivial equations. Its solutions a_i form a *free module* M_2. Eqs. (27) define the polynomial tangent vector of an irreducible hypersurface. Its solutions a_i form a module M_1, which can by found easily by Singular. Then the simultaneous solution for (27) and (28) is

$$M = M_1 \cap M_2 \tag{31}$$

which can found by the standard algorithm of intersecting 2 modules. This advanced method, in general, is much more efficient than the previous way.

Conclusion

In this article, we have introduced applications of computational algebraic geometry in the study of high-energy physics, especially at the precision frontier of particle collider. With LHC Run II, the demand for scattering-amplitude results require new methods and mathematical tools. As the mathematical field dealing with large numbers of complex variables, algebraic geometry is the natural choice for the computation of scattering amplitudes.

One issue that often appears in applied CAG is that important algorithms (e.g. Gröbner-basis or syzygy computation) are highly sensitive to the number of symbolic variables appearing in the problem. Consequently these algorithms perform far better with exact numbers (e.g. integers/rationals) as coefficients than with symbolic coefficients. Since we frequently need symbolic results in high-energy physics, we are looking forward to CAG algorithms optimized for symbolic coefficients.

References

[1] G. Ossola, C.G. Papadopoulos, R. Pittau, Nucl. Phys. B **763**, 147 (2007) [hep-ph/0609007].

[2] G. Ossola, C.G. Papadopoulos, R. Pittau, JHEP **0803**, 042 (2008) [arXiv:0711.3596].

[3] Y. Zhang, JHEP **1209**, 042 (2012) [arXiv:1205.5707].

[4] P. Mastrolia, E. Mirabella, G. Ossola, T. Peraro, Phys. Lett. B **718**, 173 (2012) [arXiv:1205.7087].

[5] S. Badger, H. Frellesvig, Y. Zhang, JHEP **1312**, 045 (2013) [arXiv:1310.1051].

[6] S. Badger, G. Mogull, A. Ochirov, D. O'Connell, JHEP **1510**, 064 (2015) [arXiv:1507.08797].

[7] T. Gehrmann, J.M. Henn, N.A. Lo Presti, Phys. Rev. Lett. **116** (2016) 062001 [arXiv:1511.05409].

[8] D.A. Kosower, K.J. Larsen, Phys. Rev. D **85**, 045017 (2012) [arXiv:1108.1180].

[9] H. Johansson, D.A. Kosower, K.J. Larsen, PoS LL **2012** (2012) 066 [arXiv:1212.2132].

[10] S. Caron-Huot, K.J. Larsen, JHEP **1210**, 026 (2012) [arXiv:1205.0801].

[11] M. Søgaard, Y. Zhang, JHEP **1312**, 008 (2013) [arXiv:1310.6006].

[12] M. Søgaard, Y. Zhang, JHEP **1407**, 112 (2014) [arXiv:1403.2463].

[13] M. Søgaard, Y. Zhang, JHEP **1412**, 006 (2014) [arXiv:1406.5044].

[14] F.V. Tkachov, Phys. Lett. B **100**, 65 (1981).

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

[16] S. Laporta, Int. J. Mod. Phys. A **15** (2000) 5087 [hep-ph/0102033].

[17] J. Gluza, K. Kajda, D.A. Kosower, Phys. Rev. D **83**, 045012 (2011) [arXiv:1009.0472].

[18] K.J. Larsen, Y. Zhang, Phys. Rev. D **93**, 041701 (2016) [arXiv:1511.01071].

[19] M. Søgaard, Y. Zhang, arXiv:1509.08897.

[20] D.R. Grayson, M.E. Stillman, Macaulay2, a software system for research in algebraic geometry, `http://www.math.uiuc.edu/Macaulay2`

[21] W. Decker, G.-M. Greuel, G. Pfister, H. Schönemann, SINGULAR 4-0-2, a computer algebra system for polynomial computations, `http://www.singular.uni-kl.de` (2015).

[22] P. Griffiths, J. Harris, Principles of Algebraic Geometry Classics Library, Wiley, 2011.

[23] A. Georgoudis, K. Larsen, Y. Zhang, to appear