A Symbolic Summation Approach to Feynman Integral Calculus

Johannes Blümlein

Deutsches Elektronen-Synchrotron, DESY, Platanenallee 6, D-15738 Zeuthen, Germany

Sebastian Klein

*Institut für Theoretische Teilchenphysik und Kosmologie, RWTH Aachen University,
D–52056 Aachen, Germany*

Carsten Schneider

Research Institute for Symbolic Computation, RISC, Johannes Kepler University Linz, Austria

Flavia Stan

Research Institute for Symbolic Computation, RISC, Johannes Kepler University Linz, Austria

Abstract

Given a Feynman parameter integral, depending on a single discrete variable N and a real parameter ε, we discuss a new algorithmic framework to compute the first coefficients of its Laurent series expansion in ε. In a first step, the integrals are expressed by hypergeometric multi-sums by means of symbolic transformations. Given this sum format, we develop new summation tools to extract the first coefficients of its series expansion whenever they are expressible in terms of indefinite nested product-sum expressions. In particular, we enhance the known multi-sum algorithms to derive recurrences for sums with complicated boundary conditions, and we present new algorithms to find formal Laurent series solutions of a given recurrence relation.

Key words: Feynman integrals, multi-summation, recurrence solving, formal Laurent series

1. Introduction

Starting with single summation over hypergeometric terms developed, e.g., in Gosper (1978); Zeilberger (1990a); Petkovšek (1992); Abramov and Petkovšek (1994); Paule (1995) symbolic summation has been intensively enhanced to multi-summation like, e.g., the holonomic approach of Zeilberger (1990b); Chyzak (2000); Schneider (2005a); Koutschan (2009). In this article we use the techniques of Fasenmyer (1945); Wilf and Zeilberger (1992) which lead to efficient algorithms developed, e.g., in Wegschaider (1997) to compute recurrence relations for hypergeometric multi-sums. Besides this, we rely on multi-summation algorithms presented in Schneider (2007) that generalize the summation techniques worked out in Petkovšek et al. (1996); the underlying algorithms are based on a refined difference field theory elaborated in Schneider (2008, 2010) that is adapted from Karr's ΠΣ-fields originally introduced in Karr (1981).

We aim at combining these summation approaches which leads to a new framework to evaluate Feynman integrals. In a nutshell, given a Feynman integral, we transform it to hypergeometric multisums, compute afterwards linear recurrences for these multi-sums, and finally decide constructively by recurrence solving whether the integrals (resp. the multisums) have series expansions whose coefficients can be represented in terms of indefinite nested sums and products. The method consists of a completely algebraic algorithm. It is therefore well-suited for implementation in computer algebra systems.

We show in a first step that Feynman parameter integrals, which contain local operator insertions, in D-dimensional Minkowski space with one time- and $(D-1)$ Euclidean space dimensions, $\varepsilon = D - 4$ and $\varepsilon \in \mathbb{R}$ with $|\varepsilon| \ll 1$, can be transformed by means of symbolic computation to hypergeometric multi-sums $\mathcal{S}(\varepsilon, N)$ with N an integer parameter. Given this representation, one can check by analytic arguments whether the integrals can be expanded in a Laurent series w.r.t. the parameter ε, and we seek summation algorithms to compute the first few coefficients of this expansion whenever they are representable in terms of indefinite nested sums and products. Due to the special input class of Feynman integrals, these solutions can be usually transformed to harmonic sums or S-sums; see Blümlein and Kurth (1999); Vermaseren (1999); Moch et al. (2002); Ablinger (2009).

In general, we present an algorithm (see Theorem 1) that decides constructively, if these first coefficients of the ε–expansion can be written in such indefinite nested product-sum expressions. Here one first computes a homogeneous recurrence by WZ-theory and Wegschaider's approach. This recurrence together with initial values gives an alternative representation for the series expansion (see Lemma 1). Moreover, we develop a recurrence solver (see Corollary 1) which computes the coefficients of the expansion in terms of indefinite nested product-sum expressions whenever this is possible. The backbone of this solver relies on algorithms from Petkovšek (1992); Abramov and Petkovšek (1994); Schneider (2001, 2005b). Since the solutions are highly nested by construction, their simplification to sum representations with minimal depth are crucial; see Schneider (2010).

From the practical point of view there is one crucial drawback of the proposed solution: looking for such recurrences is extremely expensive. For our examples arising from particle physics the proposed algorithm is not applicable considering the available computer and time resources. On that score we relax this very restrictive requirement and search for possibly inhomogeneous recurrence relations. However, the input sums have summands which present poles outside the given summation ranges. Combining Wegschaider's package `MultiSum` and the new package `FSums` presented in Stan (2010)

we determine recurrences with inhomogeneous sides consisting of well-defined sums with fewer sum quantifiers. Applying our method to these simpler sums by recursion will lead to an expansion of the right hand side of the starting recurrence. Finally, we compute the coefficients of the original input sum by our new recurrence solver mentioned above.

The outline of the article is as follows. In Section 2 we explain all computation steps that lead from Feynman integrals to hypergeometric multi-sums of the form (7) which can be expanded in a Laurent expansion (11) where the coefficients $F_i(N)$ can be represented in the form (12). In the beginning of Section 3 we face the problem that the multi-sums (7) have to be split further in the form (13) to fit the input class of our summation algorithms. We first discuss convergent sums only. The treatment of those sums which diverge in this special format or sums with several infinite summations that have difficult convergence properties will be dealt with later, cf. Remark 5. In the remaining parts of Section 3 we present the general mechanisms to compute the first coefficients $F_i(N)$ for a given hypergeometric multi-sum. In Section 4 we present an algorithmic approach to hypergeometric sums with non-standard boundary conditions. This allows us to generate the inhomogeneous sides of recurrences delivered by Wegschaider's package `MultiSum`. Finally, in Section 5 we obtain a method that is capable of computing the coefficients $F_i(N)$ in reasonable time. Conclusions are given in Section 6.

2. Multiple sum representations of Feynman integrals

We show how integrals emerging in renormalizable Quantum Field Theories, like Quantum Electrodynamics or Quantum Chromodynamics, see e.g. Blümlein (2009), can be transformed by means of symbolic computation to hypergeometric multi-sums. We study a very general class of Feynman integrals which are of relevance for many physical processes at high energy colliders, such as the Large Hadron Collider, LHC, and others.

The processes obey special-relativistic kinematics with energy-momentum vectors in Minkowski space, $\mathbb{M}^D$, see e.g., Naas and Schmid (1961), i.e., a D-dimensional linear space where the elements $a = (a_0, \vec{a}) \in \mathbb{M}^D$ decompose into the time coordinate $a_0 \in \mathbb{R}$ and the spatial coordinates $\vec{a} \in \mathbb{R}^{D-1}$ which form a $D-1$-dimensional Euclidean subspace; the bilinear form is defined by $a.b \equiv \langle a, b \rangle = a_0 b_0 - \vec{a}\vec{b} \in \mathbb{R}$ for $b = (b_0, \vec{b}) \in \mathbb{M}^D$. Below analytic continuations in $D := 4 + \varepsilon$ with $\varepsilon \in \mathbb{R}$ are considered. Here we study integrals

$$\mathcal{I}(\varepsilon, N, p) = \int \frac{d^D p_1}{(2\pi)^D} \cdots \int \frac{d^D p_k}{(2\pi)^D} \frac{\mathcal{N}(p_1, \ldots p_k; p; M^2; \Delta, N)}{(-p_1^2 + m_1^2)^{l_1} \ldots (-p_k^2 + m_k^2)^{l_k}} \prod_V \delta_V \qquad (1)$$

with $\Delta, p, p_i \in \mathbb{M}^D$ and $m_i \in \{0, M\}$ for some $M \in \mathbb{R}$ with $M > 0$. The restriction that there is only one mass M is the only one specifying the class of Feynman diagrams from arbitrary ones. The propagator powers l_i obey $l_i \in \mathbb{N}$ and for the special vector Δ in (1) one has $\Delta.\Delta = 0$. The numerator $\mathcal{N}$ is usually given in terms of finite sums where the range depends on a discrete parameter N and where the summand depends on the scalars $p.p_j, p_i.p_j, \Delta.p_i$ $(1 \leq i, j \leq k)$, on M^2 and on N. In particular, for each $N \in \mathbb{N}$, $\mathcal{N}$ is a polynomial in terms of these scalars and M^2 where the exponents of the $\Delta.p_j$ $(1 \leq j \leq k)$ in a given monomial sum up to N and the exponents of the remaining scalars and M^2 are constant. The δ_V occurring in (1) are shortcuts for Dirac delta functions in D dimensions $\delta_V \equiv \delta^{(D)} \left(\sum_{l=1}^k a_{V,l} p_l \right), a_{V,l} \in \mathbb{Q}$. I.e., if $a_{V,i} \neq 0$, we get

$$\int d^D p_i \delta^{(D)} \left(\sum_{l=1}^k a_{V,l} p_l \right) f(p_i) := \frac{f(p_i)}{|a_{V,i}|} \Big|_{p_i = u} \quad \text{with} \quad u := -\frac{1}{a_{V,i}} \sum_{l=1, l \neq i}^k a_{V,l} p_l; \qquad (2)$$

here f stands for the integrand of (1). For each such rule (2) for the remaining δ_V, one integral sign in (1) can be eliminated. As a consequence we obtain integrals of the same shape but with fewer integral signs. Such an integral may be easily linearly transformed into Euclidean integrals (Wick rotation, Feynman (1949); Wick (1950)) in the Euclidean space by replacing $a = (a_0, \vec{a}) \in \mathbb{M}^D$ with $\bar{a} = (ia_0, \vec{a})$. In this way, for $b = (b_0, \vec{b})$ the bilinear form $\langle \bar{a}, \bar{b} \rangle = -a_0 b_0 - \vec{a}.\vec{b} < 0$ obtains a definite sign; $\sqrt{-\langle \bar{a}, \bar{a} \rangle}$ is then the Euclidean norm $||\bar{a}||$. Summarizing, we obtain an Euclidean integral of the same shape as (1) with the Euclidean momenta $\bar{p}_i, \bar{p}$ (instead of p_i, p) and where the denominators can be written in the form $((\sum_{j=1}^{k} c_j^{(i)} \bar{p}_j)^2 + m_i^2)^{l_i}$ with $c_j^{(i)} \in \mathbb{Q}$ (instead of $(-p_i^2 + m_i^2)^{l_i}$); this format is due to the usage of (2).

Subsequently, we show how this Euclidean integral can be mapped to an integral on an m-dimensional unit cube. Define $D_i := (\sum_{j=1}^{k} c_j^{(i)} \bar{p}_j)^2 + m_i^2$. Then we loop over r ($r = 1, 2, \ldots, k$) as follows. For the rth iteration, fix $\bar{q} := \bar{p}_r$. W.l.o.g. assume that $c_r^{(i)} \in \{0, 1\}$ for $1 \leq i \leq k$. Now collect those denominator factors $D_i^{l_i}$ where $\bar{q}$ occurs, say $\prod_{j=1}^{n} D_{i_j}^{l_{i_j}}$ ($n \in \mathbb{N}$). Then we use the formula

$$\frac{1}{\prod_{j=1}^{n} D_{i_j}^{l_{i_j}}} = \frac{\Gamma(l)}{\prod_{j=1}^{n} \Gamma(l_{i_j})} \int_0^1 dx_1 \ldots \int_0^1 dx_n \delta\left(\sum_{j=1}^{n} x_j - 1\right) \frac{\prod_{j=1}^{n} x_j^{l_{i_j} - 1}}{(x_1 D_{i_1} + \ldots x_n D_{i_n})^l} \quad (3)$$

with $l = \sum_{j=1}^{n} l_{i_j}$; here δ is the Dirac delta function, the variables x_k are called Feynman parameters, and $\Gamma(z)$ denotes the Gamma-function. Due to the Dirac delta function, we get that $A := x_1 D_{i_1} + \ldots x_n D_{i_j} = \bar{q}^2 + a.\bar{q} + b$ where a and b are expressions free of $\bar{q}$. Hence we can write $A = (\bar{q} + a/2)^2 + R$ with $R := -a^2/4 + b$ being free of $\bar{q}$. Replacing the denominator of our integral by this formula, we can simplify A further. Namely, using the shift-invariance w.r.t. the vector $\bar{q}$, which holds in D-dimensional Euclidean space, the denominator A can be brought to the form $(\bar{q}^2 + R)$ without changing the integral. Finally, expanding the numerators and applying the $\bar{q}$-integral termwise lead to integrals of the form $\int \frac{d^D \bar{q}}{(2\pi)^D} \frac{\prod_{\lambda=1}^{m} q_\lambda.\bar{q}}{(\bar{q}^2 + R)^l}$ where the expression q_λ is free of $\bar{q}$. If m is odd, i.e., an odd number of vector multiplications w.r.t. $\bar{q}$ arise, the integral evaluates to 0 by symmetry. If m is even, one exploits the simplification

$$\int \frac{d^D \bar{q}}{(2\pi)^D} \frac{\prod_{\lambda=1}^{m/2} q_\lambda.\bar{q}}{(\bar{q}^2 + R)^l} = r(D) \int \frac{d^D k}{(2\pi)^D} \frac{(\bar{q}^2)^r}{(\bar{q}^2 + R)^l}$$

where $r(D)$ stands for a rational function in D (i.e., in ε) that can be determined by an explicit formula. To this end, the following formula is applied to the remaining integrals:

$$\int \frac{d^D \bar{q}}{(2\pi)^D} \frac{(\bar{q}^2)^r}{(\bar{q}^2 + R)^l} = \frac{1}{(16\pi^2)^{D/4}} \frac{\Gamma(r + D/2)\Gamma(l - r - D/2)}{\Gamma(D/2)\Gamma(l)(R^2)^{l-r-D/2}}.$$

Usually, these operations are carried out in terms of tensors to keep the size compact and to determine additional relations efficiently. The above procedure is repeated until all momentum integrals for the p_r ($r = 1, 2, \ldots, k$) are computed. As a result one is left with the integrals over $x_i \in [0, 1]$, equipped with a pre-factor $C(\varepsilon, N, M)$.

Step 1: From Feynman parameter integrals to Mellin–Barnes integrals and multinomial series.

Parts of these scalar integrals again can be computed trivially related to the δ-distributions,

$$\int_0^1 dx_l \delta\left(\sum_{k=1}^{n} x_k - 1\right) = \theta\left(1 - \sum_{k=1, k \neq l}^{n} x_k\right) \prod_{m=1, m \neq l}^{n} \theta(x_m),$$

where $\theta(z)$ is 1 if $z \geq 0$ and 0 otherwise. There may be more integrals, which can be computed, usually as indefinite integrals, without special effort. Mapping all Feynman-parameter integrals onto the m-dimensional unit cube (as described above) one obtains the following structure :

$$\mathcal{I}(\varepsilon, N) = C(\varepsilon, N, M) \int_0^1 dy_1 \ldots \int_0^1 dy_m \frac{\sum_{i=1}^k \prod_{l=1}^{r_i} [P_{i,l}(y)]^{\alpha_{i,l}(\varepsilon,N)}}{[Q(y)]^{\beta(\varepsilon)}} , \qquad (4)$$

with $k \in \mathbb{N}$, $r_1, \ldots, r_k \in \mathbb{N}$ and where $\beta(\varepsilon)$ is given by a rational function in ε, i.e., $\beta(\varepsilon) \in \mathbb{Q}(\varepsilon)$, and similarly $\alpha_{i,l}(\varepsilon, N) = n_{i,l}N + \overline{\alpha}_{i,l}$ for some $n_{i,l} \in \{0,1\}$ and $\overline{\alpha}_{i,l} \in \mathbb{Q}(\varepsilon)$, see also Bogner and Weinzierl (2010) in the case when no local operator insertions are present. $C(\varepsilon, N, M)$ is a factor which depends on the dimensional parameter ε, the integer parameter N and M. $P_i(y), Q(y)$ are polynomials in the remaining Feynman parameters $y = (y_1, \ldots, y_m)$ written in multi-index notation. In (4) all terms which stem from local operator insertions were geometrically resummed; see Bierenbaum et al. (2009b).

Remark. (1) After splitting the integral (4) (in particular, the k summands), the integrands fit into the input class of the multivariate Almkvist-Zeilberger algorithm. Hence, if the split integrals are properly defined, they obey homogeneous recurrence relations in N due to the existence theorems in Apagodu and Zeilberger (2006). However, so far we failed to compute these recurrences due to time and space limitations.

Remark. (2) Usually the calculation of $\mathcal{I}(\varepsilon, N)$ for fixed integer values of N is a simpler task. If sufficiently many of these values are known, one may guess these recurrences and with this input derive closed forms for $\mathcal{I}(\varepsilon, N)$ using the techniques applied in Blümlein et al. (2009). This has been illustrated for a large class of 3-loop quantities. However, at present no method is known to calculate the amount of moments needed.

The y_i-integrals finally turn into Euler integrals. Here we outline a general framework, although in practice, different algorithms are used in specific cases, cf. e.g. Ablinger et al. (2010a, 2011b). To compute the integrals (4) over the variables y_i we proceed as follows:

- decompose the denominator function using Mellin–Barnes integrals, see Paris and Kaminski (2001) and references therein,
- decompose the numerator functions, if needed, into multinomial series.

The denominator function has the structure

$$[Q(y)]^{\beta(\varepsilon)} = \left[\sum_{k=1}^n q_k(y) \right]^{\beta(\varepsilon)} ,$$

with $q_k(y) = a_1 \ldots a_m$ where $a_i \in \{1, y_i, 1 - y_i\}$ for $1 \leq i \leq m$. This function can be decomposed applying its Mellin-Barnes integral representation $(n-1)$ times,

$$\frac{1}{(A+B)^q} = \frac{1}{2\pi i} \int_{\gamma-i\infty}^{\gamma+i\infty} d\sigma \; A^\sigma \; B^{-q-\sigma} \frac{\Gamma(-\sigma)\Gamma(q+\sigma)}{\Gamma(q)} . \qquad (5)$$

Here γ denotes the real part of the contour. Often Eq. (5) has to be considered in the sense of its analytic continuation, see Whittaker and Watson (1996). The numerator factors $[P_{i,l}(y)]^{\alpha_{i,l}(\varepsilon,N)}$ obey

$$[P_{i,l}(y)]^{\alpha_{i,l}(\varepsilon,N)} = \left[\sum_{k=1}^w p_k(y) \right]^{\alpha_{i,l}(\varepsilon,N)} ,$$

where the monomials $p_k(y)$ have the same properties as $q_k(y)$. One expands

$$[P_{i,l}(y)]^{\alpha_{i,l}(\varepsilon,N)} = \sum_{k_1,\ldots,k_{w-1}\geq 0} \binom{\alpha_{i,l}(\varepsilon,N)}{k_1,\ldots,k_{w-1}} \prod_{l=1}^{w-1} p_l(y)^{k_l} p_w(y)^{\alpha_{i,l}(\varepsilon,N)-\sum_{r=1}^{w-1}k_r}.$$

Now all integrals over the variables y_j can be computed by using the formula

$$\int_0^1 dy\, y^{\alpha-1}(1-y)^{\beta-1} = B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$$

and one obtains

$$\mathcal{I}(\varepsilon,N) = \frac{1}{(2\pi i)^n} \int_{\gamma_1-i\infty}^{\gamma_1+i\infty} d\sigma_1 \ldots \int_{\gamma_n-i\infty}^{\gamma_n+i\infty} d\sigma_n$$
$$\sum_{k_1=1}^{L_1(N)} \ldots \sum_{k_v=1}^{L_v(N,k_1,\ldots,k_{v-1})} \sum_{k=1}^{l} C_k(\varepsilon,N,M) \frac{\Gamma(z_{1,k})\ldots\Gamma(z_{u,k})}{\Gamma(z_{u+1,k})\ldots\Gamma(z_{v,k})}; \tag{6}$$

$l \in \mathbb{N}$ and the summation over k_i comes from the multinomial sums, i.e., the upper bounds $L_1(N),\ldots,L_v(N,k_1,\ldots,k_{v-1})$ are integer linear in the dependent parameters or ∞. Moreover, the $z_{u,k}$ are linear functions with rational coefficients in terms of ε, the Mellin-Barnes integration variables $\sigma_1,\ldots,\sigma_n$, and the summation variables $k_1,\ldots,k_v$.

Step 2: Representation in multi–sums. The Mellin-Barnes integrals are carried out applying the residue theorem in Eq. (6). The following representation is obtained:

$$\mathcal{I}(\varepsilon,N) = \sum_{n_1=1}^{\infty} \ldots \sum_{n_r=1}^{\infty} \sum_{k_1=1}^{L_1(N)} \ldots \sum_{k_v=1}^{L_v(N,k_1,\ldots,k_{v-1})} \sum_{k=1}^{l} C_k(\varepsilon,N,M) \frac{\Gamma(t_{1,k})\ldots\Gamma(t_{v',k})}{\Gamma(t_{v'+1,k})\ldots\Gamma(t_{w',k})}. \tag{7}$$

Here the $t_{l,k}$ are linear functions with rational coefficients in terms of the $n_1,\ldots,n_r$, of the $k_1,\ldots,k_v$, and of ε. Note that the residue theorem may imply more than one infinite sum per Mellin-Barnes integral, i.e., $r \geq n$.

In general, this approach leads to a highly nested multi-sum. Fixing the loop order of the Feynman integrals and restricting to certain special situations usually enables one to find sum representations with fewer summation signs. E.g., as worked out in Bierenbaum et al. (2008), one can identify the underlying sums in terms of generalized hypergeometric functions, i.e., the number of infinite sums are reduced to one or in some cases to zero.

Step 3: Laurent series in ε. Eq. (7) can now be expanded in the parameter ε using

$$\Gamma(n+1+\bar{\varepsilon}) = \frac{\Gamma(n)\Gamma(1+\bar{\varepsilon})}{B(n,1+\bar{\varepsilon})} \tag{8}$$

with $\bar{\varepsilon} = r\varepsilon$ for some $r \in \mathbb{Q}$ and

$$B(n,1+\bar{\varepsilon}) = \frac{1}{n} \exp\left(\sum_{k=1}^{\infty} \frac{(-\bar{\varepsilon})^k}{k} S_k(n)\right) = \frac{1}{n} \sum_{k=0}^{\infty} (-\bar{\varepsilon})^k S_{\underbrace{1,\ldots,1}_{k}}(n) \tag{9}$$

and other well-known transformations for the Γ-functions. Here the harmonic sums $S_{\vec{a}}(N)$ Blümlein and Kurth (1999); Vermaseren (1999) for $N \in \mathbb{N}$ are recursively defined by

$$S_{b,\vec{a}}(N) = \sum_{k=1}^{N} \frac{(\text{sign}(b))^k}{k^{|b|}} S_{\vec{a}}(k), \quad S_{\emptyset} = 1. \tag{10}$$

6

Note that in (8) n may stand for a linear combination of parameters with coefficients in $\mathbb{Q}$. In case of non-integer weight factors r_i for the parameters in n analytic continuations of harmonic sums have to be considered Blümlein (2000, 2009); Blümlein (2010); Blümlein and Moch (2005). In case that n is not an integer one may shift to $n \rightarrow k \ n \in \mathbb{N}$, which leads to the usual definition of the harmonic sums in (9). However, the summation operators have now to be generalized and one usually ends up with *cyclotomic* harmonic sums worked out in Ablinger et al. (2011a).

Applying (8) with (9) to each factor in (7) produces for some $L > 0$ the expansion

$$\mathcal{I}(\varepsilon, N) = \sum_{l=-L}^{\infty} \varepsilon^l I_l(N); \tag{11}$$

L equals the loop order in case of infra-red finite integrals; otherwise, L may be larger.

Remark 1. In order to guarantee correctness of this construction, i.e., performing the expansion first on the summand level of (7) and afterwards applying the summation on the coefficients of the summand expansion (i.e., exchanging the differential operator D_ε and the summation quantifiers) analytic arguments have to be considered. For all our computations this construction was possible.

The general expression of the functions $I_l(N)$ in terms of nested sums are

$$I_l(N) = \sum_{n_1=1}^{\infty} ... \sum_{n_r=1}^{\infty} \sum_{k_1=1}^{L_1(N)} ... \sum_{k_v=1}^{L_v(N,k_1,...,k_{v-1})} \sum_{j=1}^{s} H_j(N; n_1, ..., n_r;, k_1, ..., k_v)$$
$$\times \prod_i S_{\vec{a}_{i,j}}(L_{i,j}(N; n_1, ..., n_r;, k_1, ..., k_v)); \tag{12}$$

$H_j(N; n_1, ..., k_v)$ denote proper hypergeometric terms [1] and $S_{\vec{a}_{i,j}}(L_{i,j}(N; n_1, ..., k_v))$ are harmonic sums with the index set $\vec{a}_{i,j}$ and $L_{i,j}$ (usually integer linear) functions of the arguments $(N; n_1, ..., k_v)$. The sum-structure in (12) is usually obtained performing the synchronization of arguments, see Vermaseren (1999), and applying the associated quasi–shuffle algebra, see Blümlein (2004).

3. First approach to the problem

In the following we limit the investigation to a sub-class of integrals of the type (1) and consider two- and simpler three-loop diagrams, which occurred in the calculation of the massive Wilson coefficients for deep-inelastic scattering; see Ablinger et al. (2011b); Blümlein et al. (2006); Bierenbaum et al. (2007, 2009a, 2008). Looking at the reduction steps of the previous section we obtain the following result. If we succeed in finding the representation (11) with (12) it follows constructively that for each $N \in \mathbb{N}$ with $N \geq \lambda$ for some $\lambda \in \mathbb{N}$ the integral $\mathcal{I}(\varepsilon, N)$ has a Laurent expansion in ε and thus it is an analytic function in ε throughout an annular region centered at 0 where the pole at $\varepsilon = 0$ has order L. In Bierenbaum et al. (2008); Ablinger et al. (2011b, 2010b) we started with

---

[1] For a precise definition of proper hypergeometric terms we refer, e.g., to Wegschaider (1997). For all our applications it suffices to know that H_j might be a product of Gamma-functions (occurring in the numerator and denominator) with linear dependence on the variables N, n_i, k_i times a rational function in these variables where the denominator factors linearly.

the sum representation of the coefficients (12) and the main task was to simplify the expressions in terms of harmonic sums. In this article, we follow a new approach that directly attacks the sum representation (7) and searches for the first coefficients of its ε-expansion (11). By splitting (7) accordingly (and pulling $C_k(\varepsilon, N, m)$) our integral can be written as a linear combination of hypergeometric multi-sums of the following form.

Assumption 1.

$$\mathcal{S}(\varepsilon, N) = \sum_{\sigma_1=p_1}^{\infty} \cdots \sum_{\sigma_s=p_s}^{\infty} \sum_{j_0=q_0}^{L_1(N)} \sum_{j_1=q_1}^{L_2(N,j_0)} \cdots \sum_{j_r=q_r}^{L_r(N,j_0,\ldots,j_{r-1})} \mathcal{F}(N, \sigma, j_0, \ldots, j_{r-1}), \varepsilon) \quad (13)$$

where

(1) $N \in \mathbb{N}$ with $N \geq \lambda$ for some given $\lambda \in \mathbb{N}$, $\varepsilon > 0$ is a real parameter;

(2) the upper summation bounds $L_l(N, j_0, \ldots, j_{l-1})$ are integer linear in $N, j_0, \ldots, j_{l-1}$, and the lower bounds are given constants $p_i, q_l \in \mathbb{N}$ for all $1 \leq i \leq s$ and $0 \leq l \leq r$;

(3) $\mathcal{F}$ is a proper hypergeometric term (see Footnote 1) with respect to the integer variable N and all summation variables $(\sigma, j) = (\sigma_1, \ldots, \sigma_s, j_0, \ldots, j_r) \in \mathbb{Z}^{s+r+1}$.

Remark 2. While splitting the sum (7) into sums of the form (13) it might happen that the infinite sums over individual monomials diverge for fixed values of ε, despite the convergence of the complete expression. We will deal with these cases in Section 5 and consider only sums which are convergent at the moment.

In other words, we assume that (13) itself is analytic in ε throughout an annular region centered at 0 and we try to find the first coefficients $F_t(N), F_{t+1}(N), \ldots, F_u(N)$ in terms of indefinite nested product-sum expressions of its expansion

$$\mathcal{S}(\varepsilon, N) = F_t(N)\varepsilon^t + F_{t+1}(N)\varepsilon^{t+1} + F_{t+2}(N)\varepsilon^{t+2} + \ldots. \quad (14)$$

with $t \in \mathbb{Z}$. In all our computations it turns out that the summand $\mathcal{F}(N, \sigma, j, \varepsilon)$ satisfies besides properties (1)–(3) the following asymptotic behavior:

(4) for all $1 \leq i \leq s$ we have

$$\mathcal{F}(N, \sigma, j, \varepsilon) = \mathcal{O}\left(\sigma_i^{-d_i} e^{-c_i \sigma_i}\right) \quad \text{as} \quad \sigma_i \to \infty \quad \text{with} \quad c_i \geq 0, \quad d_i > 0; \quad (15)$$

for simplicity we do not consider the log-parts. For later considerations in Section 4 we suppose that such constants c_i and d_i are given explicitly. E.g., using the behavior (Whittaker and Watson, 1996, Section 13.6) of $\log \Gamma(z)$ for large $|z|$ in the region where $|\arg(z)| < \pi$ and $|\arg(z + a)| < \pi$:

$$\log \Gamma(z + a) = (z + a - \frac{1}{2}) \log z - z + \mathcal{O}(1), \quad (16)$$

such constants can be easily computed. If not all $c_i > 0$ for $1 \leq i \leq s$, things get more complicated and –for simplicity– we restrict ourselves to the case that $s = 1$ and $c_1 = 0$; we refer again to Section 5 for further details how one can treat the more general case.

(5) If $s = 1$ and $c_1 = 0$, we suppose that we are given a constant $o \in \mathbb{N}$ such that

$$\mathcal{S}(\varepsilon, N) = \sum_{\sigma_1=p_1}^{\infty} \sigma_1^o \mathcal{F}(N, \sigma_1, j, \varepsilon) \quad (17)$$

converges absolutely for any small nonzero ε around 0, $N \geq B$ and any j that runs over the summation range.

Using, e.g., facts about hypergeometric functions from (Andrews et al., 1999, Thm. 2.1.1) the maximal such constant o in (17) can be determined.

Example 1. The following sum is a typical entry from the list of sum representations for a class of Feynman parameter integrals we computed:

$$
\begin{aligned}
\mathcal{U}(\varepsilon, N) &:= \sum_{\sigma_1=0}^{\infty} \sum_{j_0=0}^{N-3} \sum_{j_1=0}^{N-j_0-3} \sum_{j_2=0}^{j_0+1} \binom{j_0+1}{j_2} \frac{\binom{N-j_0-3}{j_1}\Gamma(j_1+j_2+2)\Gamma(j_1+j_2+3)}{N!} \\
&\times \frac{(-1)^N \left(\frac{\varepsilon}{2}+1\right)_{\sigma_1} (-\varepsilon)_{\sigma_1} (j_1+j_2+3)_{\sigma_1} \left(3-\frac{\varepsilon}{2}\right)_{j_1}}{(j_1+4)_{\sigma_1} \left(-\frac{\varepsilon}{2}+j_1+j_2+4\right)_{\sigma_1} \left(4-\frac{\varepsilon}{2}\right)_{j_1+j_2}} \frac{\Gamma(N-j_0-1)\Gamma(N-j_1-j_2-1)}{\Gamma(\sigma_1+1)\Gamma(j_1+4)\Gamma(N-j_0-2)};
\end{aligned}
\tag{18}
$$

we denote by $(x)_k = x(x+1)\ldots(x+k-1)$ the Pochhammer symbol defined for non-negative integers k. Then using formulas such as $(x)_k = \Gamma(x+k)/\Gamma(x)$ and $\binom{x}{k} = \Gamma(x+1)/\Gamma(x-k+1)/\Gamma(k+1)$ and applying (16) we get the asymptotic behavior $\mathcal{O}(\sigma_1^{-5})$ of the summand. Moreover, we choose the maximal $o = 3$ such that condition (17) is satisfied.

Subsequently, we will develop an algorithm that finds, whenever possible, representations for the coefficients in the expansion (14) in terms of indefinite nested sums and products [2].

Theorem 1. Let $\mathcal{S}(\varepsilon, N)$ be a sum with properties (1)–(5) from Assumption 1 which forms an analytic function in ε throughout an annular region centered at 0 with the Laurent expansion (14) for some $t \in \mathbb{Z}$ for each nonnegative N; let $u \in \mathbb{N}$. Then there is an algorithm which finds the maximal $r \in \{t-1, t, \ldots, u\}$ such that the $f_t(N), \ldots, f_r(N)$ are expressible in terms of indefinite nested product-sums; it outputs such expressions $F_t(N), \ldots, F_r(N)$ and $\lambda \in \mathbb{N}$ s.t. $f_i(k) = F_i(k)$ for all $0 \le i \le r$ and all $k \in \mathbb{N}$ with $k \ge \lambda$.

This result is based on the fact that such sums $\mathcal{S}(\varepsilon, N)$ satisfy a recurrence relation.

Example 2. Consider the single nested sum

$$
\mathcal{S}(\varepsilon, N) = \sum_{k=0}^{N-1} \frac{(-2)^k (k+2)\Gamma(4-\varepsilon)\Gamma\left(\frac{\varepsilon}{2}+3\right)\Gamma(N)\Gamma\left(-\frac{\varepsilon}{2}+k+2\right)}{\Gamma\left(2-\frac{\varepsilon}{2}\right)\Gamma(-\varepsilon+k+4)\Gamma\left(\frac{\varepsilon}{2}+k+3\right)\Gamma(N-k)}
\tag{19}
$$

over a proper hypergeometric term; note that an expansion (14) with $t = 0$ exists following the arguments from Remark 1. In the first step we compute the recurrence relation

$$
a_0(\varepsilon, N)\mathcal{S}(\varepsilon, N) + a_1(\varepsilon, N)\mathcal{S}(\varepsilon, N+1) + a_2(\varepsilon, N)\mathcal{S}(\varepsilon, N+2) = h(\varepsilon, N)
\tag{20}
$$

with

$$
\begin{aligned}
h(\varepsilon, N) &= -24N - 48 + (2N-20)\varepsilon + (2N+6)\varepsilon^2 + 2\varepsilon^3, \\
a_0(\varepsilon, N) &= 2N(N+1)(\varepsilon+2N+5), \quad a_1(\varepsilon, N) = (N+1)\left(\varepsilon^2 + 2\varepsilon N + 5\varepsilon + 4N + 12\right), \\
a_2(\varepsilon, N) &= (\varepsilon - N - 4)(\varepsilon + 2N + 3)(\varepsilon + 2N + 6)
\end{aligned}
\tag{21}
$$

---

[2] This means in particular indefinite nested sums over hypergeometric terms (like binomials, factorials, Pochhammer symbols) that may occur as polynomial expressions with the additional constraint that the summation index i_j of a sum $\sum_{i_j=1}^{i_{j+1}} f(i_j)$ may occur only as the upper index of its inner sums and products, but not inside the inner sums themselves; for a formal but lengthy definition see Schneider (2010). Typical examples are sums of the form (10) above, or of the forms (33) and (34) given below.

which holds for all $N \geq 1$. This task can be accomplished for instance by the packages Paule and Schorn (1995), Wegschaider (1997) or Schneider (2007) which are based on the creative telescoping paradigm presented in Zeilberger (1990a) or the paradigm presented in Fasenmyer (1945). Then together with the first two initial values for $N = 1, 2$,

$$\mathcal{S}(\varepsilon, 1) = 2 \qquad \text{and} \qquad \mathcal{S}(\varepsilon, 2) = 2 - \frac{6}{\varepsilon + 6} = 1 + \frac{1}{6}\varepsilon - \frac{1}{36}\varepsilon^2 + O(\varepsilon^3), \qquad (22)$$

we will be able to compute, e.g., the sum representations of the first 2 coefficients

$$F_0(N) = \frac{3(2N^2 + 4N + 1)}{2N(N+1)(N+2)} - \frac{3(-1)^N}{2N(N+1)(N+2)}, \qquad (23)$$

$$F_1(N) = \frac{10N^3 + 52N^2 + 63N + 10}{8N(N+1)(N+2)^2} - \frac{3S_1(N)}{2N(N+2)} + \frac{3S_{-1}(N)}{2N(N+2)} + \frac{(-1)^N(N-10)}{8N(N+1)(N+2)^2}; \qquad (24)$$

of the ε-expansion (14) with $t = 0$; for more details see Examples 3 and 4.

In Subsection 3.1 we will develop a recurrence solver which finds the representation of the $F_i(N)$ from (14) in terms of indefinite nested sums and products whenever this is possible. Afterwards, we combine all these methods to prove Theorem 1 in Subsection 3.2.

3.1. A recurrence solver for ε-expansions

Restricting the $\mathcal{O}$-notation to formal Laurent series $f = \sum_{i=r}^{\infty} f_i \varepsilon^i$ and $g = \sum_{i=s}^{\infty} g_i \varepsilon^i$, the notation $f = g + O(\varepsilon^t)$ for some $t \in \mathbb{Z}$ means that the order of $f - g$ is larger or equal to t, i.e., $f - g = \sum_{i=t}^{\infty} h_i \varepsilon^i$. Subsequently, $\mathbb{K}$ denotes a field with $\mathbb{Q} \subseteq \mathbb{K}$ in which the usual operations can be computed. We start with the following

Lemma 1. Let $\mu \in \mathbb{N}$, and let $a_0(\varepsilon, N), \ldots, a_d(\varepsilon, N) \in \mathbb{K}[\varepsilon, N]$ be such that $a_d(0, k) \neq 0$ for all $k \in \mathbb{N}$ with $k \geq \mu$. Let $h_t, \ldots, h_u : \mathbb{N} \to \mathbb{K}$ ($t, u \in \mathbb{Z}$ with $t \leq u$) be functions, and let $c_{i,k} \in \mathbb{K}$ with $t \leq i \leq u$ and $\mu \leq k < \mu + d$. Then there are unique functions $F_t, \ldots, F_u : \mathbb{N} \to \mathbb{K}$ (up to the first μ evaluation points) such that $F_i(k) = c_{i,k}$ for all $t \leq i \leq u$ and $\mu \leq k < d + \mu$ and such that for $T(\varepsilon, N) = \sum_{i=t}^{u} F_i(N)\varepsilon^i$ we have

$$a_0(\varepsilon, N)T(\varepsilon, N) + \cdots + a_d(\varepsilon, N)T(\varepsilon, N+d) = h_0(N) + \cdots + h_u(N)\varepsilon^u + O(\varepsilon^{u+1}) \quad (25)$$

for all $N \geq \mu$. If the $h_i(N)$ are computable, the values of the $F_i(N)$ with $N \geq \mu$ can be computed by recurrence relations.

Proof. Plugging the ansatz $T(\varepsilon, N) = \sum_{i=t}^{u} F_i(N)\varepsilon^i$ into (25) and doing coefficient comparison w.r.t. ε^t yields the constraint

$$a_0(0, N)F_t(N) + \cdots + a_d(0, N)F_t(N+d) = h_t(N). \qquad (26)$$

Since $a_d(0, N)$ is non-zero for any integer evaluation $N \geq \mu$, the function $F_0 : \mathbb{N} \to \mathbb{K}$ is uniquely determined by the initial values $F_t(\mu) = c_{t,\mu}, \ldots, F_t(\mu + d - 1) = c_{t,\mu+d-1}$ – up to the first μ evaluation points; in particular the values $F_t(k)$ for $k \geq \mu$ can be computed by the recurrence relation (26). Moving the $F_t(N)\varepsilon^t$ in (25) to the right hand side gives

$$a_0(\varepsilon, N) \sum_{i=t+1}^{u} F_i(N)\varepsilon^i + \cdots + a_d(\varepsilon, N) \sum_{i=t+1}^{u} F_i(N+d)\varepsilon^i$$
$$= -\left[a_0(\varepsilon, N)h_t(N)\varepsilon^t + \cdots + a_d(\varepsilon, N)h_t(N+d)\varepsilon^t \right] + \sum_{i=t}^{u} h_i(N)\varepsilon^i;$$

denote the coefficient of ε^i on the right side by $\tilde{h}_i$. Since the coefficient of ε^t on the left side is 0, it is also 0 on the right side and we can write

$$a_0(\varepsilon, N) \sum_{i=t+1}^{u} F_i(N)\varepsilon^i + \cdots + a_d(\varepsilon, N) \sum_{i=t+1}^{u} F_i(N+d)\varepsilon^i = \sum_{i=t+1}^{u} \tilde{h}_i(N)\varepsilon^i + O(\varepsilon^{u+1})$$

for all $N \in \mathbb{N}$ with $N \geq \mu$. Repeating this process proves the lemma. $\quad\square$

Example 3. Consider the recurrence (20) with the coefficients (21). Then by Lemma 1 there are unique functions $F_0(N)$ and $F_1(N)$ with $T(N) = F_0(N) + F_e(N)\varepsilon$ such that $T(\varepsilon, 1) = 2$, $T(\varepsilon, 2) = 1 + \frac{1}{6}\varepsilon$ and

$$a_0(\varepsilon, N)T(\varepsilon, N) + a_1(\varepsilon, N)T(\varepsilon, N+1) + a_2(\varepsilon, N)T(\varepsilon, N+2) = h(\varepsilon, N) + O(\varepsilon^2) \quad (27)$$

hold for $N \geq 1$. In particular, by setting $\varepsilon = 0$, we get

$$a_0(0, N)F_0(N) + a_1(0, N)F_0(N+1) + a_2(0, N)F_0(N+2) = -24N - 48; \qquad (28)$$

the values of $F_0(N)$ can be computed with (28) and the initial values $F_0(1) = 2$, $F_0(2) = 1$.

At this point we exploit algorithms from Petkovšek (1992); Abramov and Petkovšek (1994); Schneider (2001, 2005b) which can constructively decide if a solution with certain initial values is expressible in terms of indefinite nested products and sums. To be more precise, with the algorithms implemented in `Sigma` one can solve the following problem.

Problem RS: **R**ecurrence **S**olver for indefinite nested product-sum expressions.
Given $a_0(N), \ldots, a_d(N) \in \mathbb{K}[N]$; given $\mu \in \mathbb{N}$ such that $a_d(k) \neq 0$ for all $k \in \mathbb{N}$ with $N \geq \mu$; given an expression $h(N)$ in terms of indefinite nested product-sum expressions which can be evaluated for all $N \in \mathbb{N}$ with $N \geq \mu$; given the initial values $(c_\mu, \ldots, c_{\mu+d-1})$ which produce the sequence $(c_i)_{i \geq \mu} \in \mathbb{K}^{\mathbb{N}}$ by the defining recurrence relation

$$a_0(N)c_N + a_1(N)c_{N+1} + \cdots + a_d(N)c_{N+d} = h(N) \quad \forall N \geq \mu.$$

Find, if possible, $\lambda \in \mathbb{N}$ with $\lambda \geq \mu$ and an indefinite nested product-sum expression $g(N)$ such that $g(k) = c_k$ for all $k \geq \lambda$.

Remark. Later, we will give further details only for a special case that occurred in almost all instances of our computations related to Feynman integrals; see Theorem 3.

Example 4. With the input $F_0(1) = 2$, $F_0(2) = 1$ and (28) `Sigma` computes the solution (23). Plugging this partial solution $T(\varepsilon, N) = F_0(N) + \ldots$ into (27) and doing coefficient comparison leads to

$$\sum_{i=0}^{2} a_i(0, N)F_1(N+i) = \frac{-10N^4 - 98N^3 - 344N^2 - 511N - 267}{(N+2)(N+3)(N+4)} - \frac{3(-1)^N(3N+7)}{(N+2)(N+3)(N+4)}.$$

Then together with $F_1(1) = 0$, $F_1(2) = 1/6$, `Sigma` finds (24). Since also (2) satisfies (27) with the same initial values (22), the first two coefficients of the expansion of (2) are equal to $F_0(N)$ and $F_1(N)$ by Lemma 1.

This iterative procedure can be summarized as follows.

Algorithm FLSR (**F**ormal **L**aurent **S**eries solutions of linear **R**ecurrences)
Input: $\mu \in \mathbb{N}$; $a_0(\varepsilon, N), \ldots, a_d(\varepsilon, N) \in \mathbb{K}[\varepsilon, N]$ such that $a_d(0, k) \neq 0$ for all $k \in \mathbb{N}$ with $k \geq \mu$; indefinite nested product-sum expressions $h_t(N), \ldots, h_u(N)$ ($t, u \in \mathbb{Z}$ with $t \leq u$) which can be

evaluated for all $N \in \mathbb{N}$ with $N \geq \mu$; $c_{i,j} \in \mathbb{K}$ with $t \leq i \leq u$ and $\mu \leq j < \mu + d$.

Output $(r, \lambda, \tilde{T}(N))$: The maximal number $r \in \{t - 1, 0, \ldots, u\}$ s.t. for the unique solution $T(N) = \sum_{i=t}^{u} F_i(N)\varepsilon^i$ with $F_i(k) = c_{i,k}$ for all $\mu \leq k < \mu + d$ and with the relation (25) the following holds: there are indefinite nested product-sum expressions that are equal to $F_t(N), \ldots, F_r(N)$ for all $N \geq \lambda$ for some $\lambda \geq \mu$; if $r \geq 0$, return such an expression $\tilde{T}(N)$ for $T(N)$ together with λ.

   (1) (Preprocessing) By Lemma 1 we can compute as many initial values $c_{i,k} := F_i(k)$ for $k \geq \mu$ as needed for the steps given below (at most $\lambda - \mu$ extra values are needed).
   (2) Set $r := t$, $\lambda := \mu$, and $\tilde{T}(N) := 0$.
   (3) Note that $(F_r(N))_{N \geq \mu}$ is defined by the initial values $F_r(N)$ ($\lambda \leq N < d + \lambda$) and the recurrence
$$a_0(0, N)F_r(N) + \cdots + a_d(0, N)F_r(N + d) = h_r(N) \qquad (29)$$
   for all $N \in \mathbb{N}$ with $N \geq \lambda$; see the proofs of Lemma 1 or Theorem 2. By solving problem RS decide constructively if there is a $\lambda' \geq \lambda$ such that $F_r(N)$ can be computed in terms of an indefinite nested product-sum expression $\tilde{F}_r(N)$ for all $N \in \mathbb{N}$ with $N \geq \lambda'$.
   (4) If this fails, RETURN $(r - 1, \lambda, \tilde{T}(N))$. Otherwise, set $\tilde{T}(N) := \tilde{T}(N) + \tilde{F}_r(N)\varepsilon^r$.
   (5) If $r = u$, RETURN $(r, \lambda, \tilde{T}(N))$.
   (6) Collect the coefficients (product-sum expressions) w.r.t. ε^i for all i $(r + 1 \leq i \leq u)$:
$$h_i'(N) := \mathrm{coeff}\Big(-\Big[a_0(\varepsilon, N)F_r(N) + \cdots + a_d(\varepsilon, N)F_r(N + d)\Big] + \sum_{i=r+1}^{u} h_i(N)\varepsilon^i, \varepsilon^i\Big).$$
   (7) Set $h_i := h_i'$ for all $r + 1 \leq i \leq u$, set $r := r + 1$ and GOTO Step 3.

Theorem 2. The algorithm terminates and fulfills the input–output specification.

Proof. We show that entering the rth iteration of the loop $(r \geq t)$ we have for all $N \geq \lambda$ that

$$a_0(\varepsilon, N)\sum_{i=r}^{u} F_i(N)\varepsilon^i + \cdots + a_d(\varepsilon, N)\sum_{i=r}^{u} F_i(N + d)\varepsilon^i = \sum_{i=r}^{u} h_i(N)\varepsilon^i + O(\varepsilon^{u+1}) \quad (30)$$

where the $h_r(N), \ldots, h_u(N)$ are given explicitly in terms of indefinite nested product-sum expressions. Moreover, we show that the obtained expression $\tilde{T}(N) = \sum_{i=t}^{r-1} \tilde{F}_i(N)\varepsilon^i$ equals the values $\sum_{i=t}^{r-1} F_i(N)\varepsilon^i$ for each $N \geq \lambda$. For $r = t$ this holds by assumption. Now suppose that these properties hold when entering the rth iteration of the loop $(r \geq t)$. Then coefficient comparison in (30) w.r.t. ε^r yields the constraint (29) for all $N \geq \lambda$ as claimed in Step 3 of the algorithm. Solving problem RS decides constructively if there is a $\lambda' \geq 0$ such that $F_r(N)$ can be computed by an expression in terms of indefinite nested product-sum expressions, say $\tilde{F}_r(N)$, for all N with $N \geq \lambda'$. If this fails, $F_r(N)$ cannot be represented with such an expression and the output $(r - 1, \lambda, \tilde{T}(N))$ with $\tilde{T}(N) = \sum_{i=t}^{r-1} \tilde{F}_i(N)$ is correct. Otherwise, the indefinite nested product-sum expressions $\tilde{F}_i(N)$ for $t \leq i \leq r$ give the values $F_i(N)$ for all $N \in \mathbb{N}$ with $N \geq \lambda'$. Now move the term $F_r(N)\varepsilon^r$ in (30) to the right hand side and replace it with $\tilde{F}_r(N)\varepsilon^r$. This gives

$$a_0(\varepsilon, N)\sum_{i=r+1}^{u} F_i(N)\varepsilon^i + \cdots + a_d(\varepsilon, N)\sum_{i=r+1}^{u} F_i(N + d)\varepsilon^i = -\sum_{i=0}^{d} a_i(\varepsilon, N)\tilde{F}_r(N + i)$$
$$+ \sum_{i=r}^{u} h_i(N)\varepsilon^i + O(\varepsilon^{u+1}) =: \tilde{h}_{r+1}(N)\varepsilon^{r+1} + \cdots + \tilde{h}_u(N)\varepsilon^u + O(\varepsilon^{u+1})$$

for all $N \geq \lambda'$ where $\tilde{h}_{r+1}(N), \ldots, \tilde{h}_u(N)$ are given in terms of indefinite nested product-sum expressions that can be evaluated for all $N \in \mathbb{N}$ with $N \geq \lambda'$. By redefining the $h_i(N)$ as in Step 7 of the algorithm we obtain the relation (30) for the case $r + 1$. $\square$

Algorithm FLSR has been implemented within the summation package Sigma. E.g., the expansion for the sum (19) with $s = 0$, $t = 1$ and $\texttt{start} = 1$ is computed by

$$\texttt{GenerateExpansion}[a_0(\varepsilon, N)S[N] + a_1(\varepsilon, N)S[N + 1] + a_2(\varepsilon, N)S[N + 2],$$
$$\{-24N - 48, 2N - 20\}, S[N], \{\varepsilon, s, t\}, \{\texttt{start}, \{\{2, 1\}, \{0, 1/6\}\}\}];$$

here the $a_i(\varepsilon, N)$ stand for the polynomials (21), $\{-24N - 48, 2N - 20\}$ is the list of the first coefficients on the right hand side of (20), and $\texttt{start}$ tells the procedure that the list of initial values $\{\{2, 1\}, \{0, 1/6\}\}$ from (22) corresponds to $N = 1, 2$.

   As demonstrated already in Example 4 the following application is immediate.

Corollary 1. For each nonnegative N, let $\mathcal{S}(\varepsilon, N)$ be an analytic function in ε throughout an annular region centered at 0 with the Laurent expansion $\mathcal{S}(\varepsilon, N) = \sum_{i=t}^{\infty} f_i(N)\varepsilon^i$ for some $t \in \mathbb{Z}$, and suppose that $\mathcal{S}(\varepsilon, N)$ satisfies the recurrence (25) with coefficients and inhomogeneous part as stated in Algorithm FLSR for some $\mu \in \mathbb{N}$; define $c_{i,k} := F_i(k)$ for $t \leq i \leq u$ and $\mu \leq k < \mu + d$. Let $(r, \lambda, \sum_{i=t}^{r} F_i(N)\varepsilon^i)$ be the output of Algorithm FLSR. Then $f_i(k) = F_i(k)$ for all $t \leq i \leq r$ and all $k \in \mathbb{N}$ with $k \geq \lambda$.

For further considerations we restrict to the following special case. We observed –to our surprise– in almost all examples arising from Feynman integrals that the operator

$$\sum_{i=0}^{d} a_i(0, N)S_N^i = c(N)(S_N - b_d(N))(S_N - b_{d-1}(N))\ldots(S_N - b_1(N)) \qquad (31)$$

with the shift operator S_N factorizes completely for some $b_1, \ldots, b_d, c \in \mathbb{K}(N)$; the rational functions can be computed by Petkovšek's algorithm Petkovšek (1992). In this particular instance we can construct immediately the complete solution space of

$$a_0(0, N)F(N) + \cdots + a_d(0, N)F(N + d) = X(N) \qquad (32)$$

for a generic sequence $X(N)$. Namely, choose $\mu_i \in \mathbb{N}$ such that the numerator and denominator polynomial of $b_i(j)$ have no zeros for all evaluations $j \in \mathbb{N}$ with $j \geq \mu_i$, and take $\lambda := \max_{1 \leq i \leq d} \mu_i + 1$. Now define for $1 \leq i \leq d$ the hypergeometric terms $h_i(N) = \prod_{j=\lambda}^{N} b_i(j - 1)$. Then by Abramov and Petkovšek (1994) one gets the d linearly independent solutions

$$H_1(N) := h_1(N), \ldots, H_d(N) := h_1(N) \sum_{i_1=\lambda}^{N-1} \frac{h_2(i_1)}{h_1(i_1 + 1)} \cdots \sum_{i_{d-1}=\lambda}^{i_{d-2}-1} \frac{h_d(i_{d-1})}{h_d(i_{d-1} + 1)} \qquad (33)$$

of the homogeneous version of (32), and the particular solution

$$P(N) := \frac{h_1(N)}{c(N)} \sum_{i_1=\lambda}^{N-1} \frac{h_2(i_1)}{h_1(i_1 + 1)} \cdots \sum_{i_{d-1}=\lambda}^{i_{d-2}-1} \frac{h_d(i_{d-1})}{h_{d-1}(i_{d-1} + 1)} \sum_{i_d=\lambda}^{i_{d-1}-1} \frac{X(i_d)}{h_d(i_d + 1)} \qquad (34)$$

of (32) itself. In other words, the solution space of (32) is explicitly given by

$$\{c_1 H_1(N) + \cdots + c_d H_d(N) + P(N) | c_1 \ldots, c_d \in \mathbb{K}\}; \qquad (35)$$

here the nesting depth (counting the nested sums) of H_i is $i - 1$ and of P is d.

   Given this explicit solution space (35) we end up with the following result.

Theorem 3. Let $h_t(N), h_{t+1}(N), \ldots$ with $t \in \mathbb{Z}$ be functions that are computable in terms of indefinite nested product-sum expressions where the nesting depth of the

13

summation quantifiers of $h_i(N)$ is d_i; let $a_i(\varepsilon, N) \in \mathbb{K}[\varepsilon, N]$ be such that the operator factors as in (31) for some $c, b_i \in \mathbb{K}(N), c \neq 0$. If $\mathcal{S}(\varepsilon, N) = \sum_{i=t}^{\infty} F_i(N)\varepsilon^i$ is a solution of

$$a_0(\varepsilon, N)\mathcal{S}(\varepsilon, N) + \cdots + a_d(\varepsilon, N)\mathcal{S}(\varepsilon, N + d) = h_t(N)\varepsilon^t + h_{t+1}(N)\varepsilon^{t+1} + \ldots, \quad (36)$$

for some functions $F_i(N)$, then the values of $F_i(N)$ can be computed by indefinite nested product-sum expressions $\tilde{F}_i(N)$. The depth of the $\tilde{F}_i(N)$ is $\leq \max_{t \leq j \leq i}(d_j + (i - j + 1)d))$.

Proof. Choose $\mu \in \mathbb{N}$ with $\mu \geq d$ such that $a_d(k) \neq 0$ for all integers $k \geq \mu$ and such that the sequences $h_i(k)$ can be computed for indefinite nested product-sum expressions for each $k \geq \mu$. Consider the rth iteration of the loop of Algorithm FLSR. Since $F_r(N)$ is a solution of (32) with $X(N) = h_r(N)$ for all $N \geq \gamma$, $F_r(N)$ is a linear combination of (35). Taking the first d initial values $F_r(\mu), \ldots, F_r(\mu + d - 1)$ the c_i are uniquely determined. Induction on $r \in \mathbb{N}$ proves the theorem. The bound on the depth is immediate. $\quad\square$

If the operator (29) factorizes as stated in (31), Alg. FLSR can be simplified as follows.

Simplification 1. The factorization (31) needs to be computed only once and the solutions $F_i(N)$ can be obtained in terms of indefinite nested product-sum expressions by simply plugging in the results of the previous steps. E.g., for our running example, we get the generic solution

$$\frac{c_1}{N(N+2)} + c_2 \frac{\sum_{i_1=1}^{N} \frac{-(-1)^{i_1}(2i_1+1)}{i_1(i_1+1)}}{2N(N+2)} - \frac{\sum_{i_1=1}^{N} \frac{(-1)^{i_1}(2i_1+1)}{i_1(i_1+1)} \sum_{i_2=1}^{i_1} \frac{(-1)^{i_2} i_2^2 X(i_2-2)}{(2i_2-1)(2i_2+1)}}{2N(N+2)} \quad (37)$$

of the recurrence $a_0(0, N)F(N) + a_1(0, N)F(N+1) + a_2(0, N)F(N+2) = X(N)$ where the coefficients are defined as in (21). In this way, one gets the solution $F_0(N)$ in terms of a double sum by setting $c_1 = c_2 = 0$ and $X(i_2) = -24i_2 + 48$ in (37), i.e.,

$$F_0(N) = \frac{-1}{2N(N+2)} \sum_{i_1=1}^{N} \frac{(-1)^{i_1}(1 + 2i_1)}{i_1(1+i_1)} \sum_{i_2=1}^{i_1} \frac{-(-1)^{i_2} 24 i_2^3}{(-1 + 2i_2)(1 + 2i_2)}. \quad (38)$$

One step further, one gets the solution $F_1(N)$ in terms of a quadruple sum by setting $c_1 = c_2 = 0$ and plugging the double sum expression

$$X(i_2) = 2i_2 - 20 - \text{coeff}(a_0(\varepsilon, i_2)F_0(i_2) + a_1(\varepsilon, i_2)F_0(i_2 + 1) + a_2(\varepsilon, i_2)F_0(i_2 + 2), \varepsilon)$$

into (37). Similarly, one obtains a sum expressions of $F_2(N)$ with nesting depth 6.

Minimizing the nesting depth. Given such highly nested sum expressions, the summation package Sigma finds alternative sum representations with minimal nesting depth. The underlying algorithms are based on a refined difference field theory worked out in Schneider (2008, 2010) that is adapted from Karr's $\Pi\Sigma$-fields originally introduced in Karr (1981). E.g., with this machinery, we simplify the double sum (38) to (23), and we reduce the quadruple sum expression for $F_1(N)$ to expressions in terms of single sums (24).

Simplification 2: The solutions (33) of the homogeneous version of the recurrence (32) can be pre-simplified to expressions with minimal nesting depth by the algorithms mentioned above. Moreover, using the algorithmic theory described in Kauers and Schneider (2006) the algorithms in Schneider (2008) can be carried over to the sum expressions like (34) involving an unspecified sequence $X(i_d)$. With this machinery, (37) simplifies to

$$\frac{c_1}{N(N+2)} + \frac{c_2(-1)^{N+1}}{2N(N+1)(N+2)} - \frac{\sum_{i_1=1}^{N} \frac{i_1 X(i_1-2)}{(2i_1-1)(2i_1+1)}}{2N(N+2)} - \frac{(-1)^N \sum_{i_1=1}^{N} \frac{(-1)^{i_1} i_1^2 X(i_1-2)}{(2i_1-1)(2i_1+1)}}{2N(N+1)(N+2)}.$$

14

Performing this extra simplification, the blow up of the nesting depth for the solutions $F_0(N), F_1(N), F_2(N), \dots$ reduces considerably: instead of nesting depth $2, 4, 6, \dots$ we get the nesting depths $1, 2, 3, \dots$. In particular, given these representations the simplification to expressions with optimal nesting depth in Step 2 also speeds up.

For simplicity we assumed that the $a_i(\varepsilon, N)$ are polynomials in ε. However, all arguments can be carried over immediately to the situation where the $a_i(\varepsilon, N)$ are formal power series with the first coefficients given explicitly. Moreover, our algorithm is applicable for more general sequences $a_i(N)$ and $h_i(N)$ whenever there are algorithms available that solve problem RS. E.g., if the coefficients $a_i(N)$ itself are expressible in terms of indefinite nested product-sum expression, problem RS can be solved by Abramov et al. (2011), and hence Algorithm FLSR is executable.

3.2. An effective method for multi-sums

For a multi-sum $\mathcal{S}(\varepsilon, N)$ with the properties (1)–(5) from Assumption 1 and with the assumption that it has a series expansion (14) for all $N \geq \lambda$ for some $\lambda \in \mathbb{N}$, the ideas of the previous section can be carried over as follows.

Step 1: Finding a recurrence. By WZ-theory (Wilf and Zeilberger, 1992, Cor. 3.3) and ideas given in (Wegschaider, 1997, Theorem 3.6) it is guaranteed that there is a recurrence of the form
$$a_0(\varepsilon, N)\mathcal{S}(\varepsilon, N) + \cdots + a_d(\varepsilon, N)\mathcal{S}(\varepsilon, N + d) = 0 \tag{39}$$
with coefficients $a_i(\varepsilon, N) \in \mathbb{K}[\varepsilon, N]$ for the multi-sum $\mathcal{S}(\varepsilon, N)$ in N that can be computed, e.g., by Wegschaider's algorithm; for infinite sums similar arguments have to be applied as in Step 2.2 of Section 4. Given such a recurrence, let $\mu \in \mathbb{N}$ with $\mu \geq \lambda$ such that $a_d(0, N) \neq 0$ for all $N \in \mathbb{N}$ with $N \geq \mu$.

Step 2: Determining initial values. If the sum (13) contains no infinite sums, i.e., $s = 0$, the initial values $F_i(k)$ in $\mathcal{S}(\varepsilon, k) = \sum_{i=t}^{\infty} F_i(k)\varepsilon^i$ for $k = \mu, \mu + 1, \dots$ can be computed immediately and can be expressed usually in terms of rational numbers. However, if infinite sums occur, it is not so obvious to which values these infinite sums evaluate for our general input class– by assumption we only know that the $F_i(k)$ for a specific integer $k \geq \mu$ are real numbers. At this point we emphasize that our approach works regardless of whether we express these sums in terms of well known constants or we just keep the symbolic form in terms of infinite sums. In a nutshell, if we do not know how to represent these values in a better way, we keep the sum representation. However, whenever possible it is desirable to rewrite these sums in terms of known values or special functions. Examples are harmonic sums which are known as limits for the external index $N \to \infty$, see Blümlein and Kurth (1999); Vermaseren (1999), to yield Euler-Zagier and multiple zeta values, cf. Blümlein et al. (2010) and references therein, and generalized harmonic sums, see Moch et al. (2002) which give special values of S-sums. In massive 2-loop computations and for the simpler 3-loop topologies these are the only known classes, whereas extensions are known in case of more massive lines, cf. e.g. Broadhurst (1999).

Step 3: Recurrence solving. Given such a recurrence (39) together with the initial values of $\mathcal{S}(\varepsilon, N)$ (hopefully in a nice closed form) we can activate Algorithm FLSR. Then by Corollary 1, we have a procedure that decides if the first coefficients of the expansion are expressible in terms of indefinite nested product-sum expressions.

Summarizing, we obtain Theorem 1 stated already in the beginning of this section.

As mentioned already in the introduction, the proposed algorithm (see steps 1,2,3 from above) is not feasible for our examples arising form particle physics: forcing Wegschaider's implementation to find a homogeneous recurrence is extremely expensive and usually fails due to the insufficient computational resources. Subsequently, we relax this restriction and search for recurrence relations which are not necessarily homogeneous.

4. Finding recurrence relations for multi-sums

Given a multi-sum $\mathcal{S}(N)$ of the form (13) we present a general method to compute a linear recurrence of $\mathcal{S}(N)$. Here the challenge is to deal with infinite sums and summands which are not well defined outside the summation range. We proceed as follows.

Step 1: Finding a summand recurrence. The sum (13) fits the input class of the algorithm Wegschaider (1997), an extension of multivariate WZ-summation due to Wilf and Zeilberger (1992). This allows us to compute a recurrence for the hypergeometric summand of (13). Before giving further details, we recall that an expression $\mathcal{F}(N, \sigma, j, \varepsilon)$ is called hypergeometric in N, σ, j, if there are rational functions $r_{\nu,\mu,\eta}(N, \sigma, j, \varepsilon) \in \mathbb{K}(N, \sigma, j, \varepsilon)$ such that $\frac{\mathcal{F}(N,\sigma,j,\varepsilon)}{\mathcal{F}(N+\nu,\sigma+\mu,j+\eta,\varepsilon)} = r_{\nu,\mu,\eta}(N, \sigma, j, \varepsilon)$ at the points $(\nu, \mu, \eta) \in \mathbb{Z}^{r+s+2}$ where this ratio is defined. Then the Mathematica package `MultiSum` described in Wegschaider (1997) solves the following problem by coefficient comparison and solving the underlying system of linear equations.

Given a hypergeometric term $\mathcal{F}(N, \sigma, j, \varepsilon)$, a finite structure set $\mathbb{S} \subset \mathbb{N}^{s+r+2}$ (w.l.o.g. we restrict to positive shifts) and degree bounds $B \in \mathbb{N}$, $\beta \in \mathbb{N}^s$, $b \in \mathbb{N}^{r+1}$.
Find, if possible, a recurrence of the form

$$\sum_{(u,v,w)\in\mathbb{S}} c_{u,v,w}(N, \sigma, j, \varepsilon) \mathcal{F}(N+u, \sigma+v, j+w, \varepsilon) = 0 \tag{40}$$

with polynomial coefficients $c_{u,v,w} \in \mathbb{K}[N, \sigma, j, \varepsilon]$, not all zero, where the degrees of the variables N, j_i and σ_i are bounded by B, β_i and b_i, respectively.

Remark 3. (1) In general, choosing $\mathbb{S}$ large enough, there always exists a summand recurrence (40) for proper hypergeometric summands $\mathcal{F}$ (see Footnote 1) due to Wilf and Zeilberger (1992). In all our computations we found such a recurrence by setting the degree bounds to 1, i.e., $B = \beta_i = b_i = 1$.
(2) To determine a small structure set $\mathbb{S} \subseteq \mathbb{N}^{s+r+2}$ which provides a solution w.r.t. our fixed degree bounds, A. Riese and B. Zimmermann enhanced the package `MultiSum` by a method based on modular computations. In this way one can loop through possible choices inexpensively until one succeeds to find such a recurrence (40).

Next, the algorithm successively divides the polynomial recurrence operator (40) by all forward-shift difference operators

$$\Delta_{\sigma_i}\mathcal{F}(N, \sigma, j, \varepsilon) := \mathcal{F}(N, \sigma_1, \ldots, \sigma_i+1, \ldots, \sigma_s, j, \varepsilon) - \mathcal{F}(N, \sigma, j, \varepsilon)$$

for $1 \le i \le s$, as well as by similar Δ-operators defined for the variables from j_i which have finite summation bounds.

At last we obtain an operator free of shifts in the summation variables (σ, j) called the principal part of the recurrence (40) which equals the sum of all delta parts in the

16

summation variables from (σ, j), i.e.,

$$\sum_{m \in \mathbb{S}'} a_m(\varepsilon, N) \mathcal{F}(N + m, \sigma, j, \varepsilon) = \sum_{l=0}^{r} \Delta_{j_l} \left(\sum_{(m,n) \in \mathbb{S}'_l} d_{m,n}(N, \sigma, j, \varepsilon) \mathcal{F}(N + m, \sigma, j + n, \varepsilon) \right)$$
$$+ \sum_{i=1}^{s} \Delta_{\sigma_i} \left(\sum_{(m,k,n) \in \mathbb{S}_i} b_{m,k,n}(N, \sigma, j, \varepsilon) \mathcal{F}(N + m, \sigma + k, j + n, \varepsilon) \right) \tag{41}$$

where the coefficients a_m, usually not all zero (see Remark 4.2), $b_{m,k,n}$ and $d_{m,n}$ are polynomials and the sets $\mathbb{S}' \subset \mathbb{N}$, $\mathbb{S}_i \subset \mathbb{N}^{s+r+2}$ and $\mathbb{S}'_l \subset \mathbb{N}^{r+2}$ are finite. Recurrences of the form (41) satisfied by the hypergeometric summand are called certificate recurrences and have polynomial coefficients $a_m(\varepsilon, N)$ free of the summation variables from (σ, j), while the coefficients of the delta-parts are polynomials involving all variables.

Remark 4. (1) In principle, the degrees of the polynomials $b_{m,k,n}$ and $d_{m,n}$ arising in (41) can be chosen arbitrarily large w.r.t. σ_i and j_i. However, in Step 2 we will sum (41) over the input range and hence we have to guarantee that the resulting sums over (41) are well defined. As a consequence, the degrees of the $d_{m,n}$ and $b_{m,k,n}$ w.r.t. the variables σ_i have to be chosen carefully if in (15) one of the constants c_i is zero. As mentioned earlier, for such situations we restrict ourselves to the case $s = 1$. In this case, the degree in the $b_{m,k,n}$ should be smaller than the constant d_1 from (15) and the degree in the $d_{m,n}$ should be not bigger than the constant o from (17). To control this total bound $b := \min(d_1 - 1, o)$, we exploit the following observation (Wegschaider, 1997, p. 43): While transforming (40) to (41) by dividing through the operators (4), one only has to perform a simple sequence of additions of the occurring coefficients in (40), and thus the degrees w.r.t. the variables do not increase. Summarizing, if we choose β_1 in our ansatz such that $\beta_1 < b$, the degrees in the $b_{m,k,n}$ and $d_{m,n}$ w.r.t. the variable σ_1 are smaller than b.
(2) In general, it might happen that the principal part is 0, i.e., we get a trivial remainder within the operator divisions. In (Wegschaider, 1997, Thm. 3.2) this situation was resolved at the cost of increasing the degrees w.r.t. some of the variables. If within this construction the degree w.r.t. σ_1 increases too much, manual adjustment is needed (e.g., force the structure set to be different or change the degree bounds manually). However, this exotic case never occurred within our computations.

Example 5. For the sum

$$S(\varepsilon, N) := \sum_{j_0=0}^{N-3} \sum_{j_1=0}^{N-3-j_0} \underbrace{(-1)^{j_1}(j_1 + 1) \binom{N - 2 - j_0}{j_1 + 1} \frac{\Gamma(j_0 + j_1 + 1)\left(1 - \frac{\varepsilon}{2}\right)_{j_0}\left(3 - \frac{\varepsilon}{2}\right)_{j_1}}{(4 - \varepsilon)_{j_0+j_1}\left(\frac{\varepsilon}{2} + 4\right)_{j_0+j_1}}}_{=:\mathcal{F}(N, j_0, j_1)} \tag{42}$$

with the discrete parameter $N \geq 3$ and $\varepsilon > 0$ the package `MultiSum` computes the summand recurrence

$$(\varepsilon - 2N)N\mathcal{F}(N, j_0, j_1) - (\varepsilon - N - 3)(\varepsilon + 2N + 2)\mathcal{F}(N + 1, j_0, j_1)$$
$$= \Delta_{j_0}[(\varepsilon^2 + j_0\varepsilon + \varepsilon - 2j_1 - 2j_0N - 4j_1N - 12N - 6)\mathcal{F}(N + 1, j_0, j_1)]$$
$$+ \Delta_{j_1}[(\varepsilon - 2N)(j_0 + j_1 - N + 1)\mathcal{F}(N, j_0, j_1)]$$
$$+ (-2N^2 + \varepsilon N + 2j_0N + 4j_1N + 4N - 2\varepsilon - \varepsilon j_0 + 2j_1)\mathcal{F}(N + 1, j_0, j_1)]. \tag{43}$$

Step 2: A recurrence for the sum. Taking as input the certificate recurrences (41) we algorithmically find the inhomogeneous part of the recurrence satisfied by the sum (13) which will contain special instances of the original multi-sum of lower nesting depth.

The recurrence for the multi-sum (13) is obtained by summing the certificate recurrence (41) over all variables from (σ, j) in the given summation range $\mathcal{R} \subseteq \mathbb{Z}^{s+r+1}$. Since it can be easily checked whether the summand $\mathcal{F}$ satisfies the (41), the certificate recurrence also provides an algorithmic proof of the recurrence for the multi-sum $\mathcal{S}(N, \varepsilon)$. In particular, since we set up the degrees of the coefficients in (41) w.r.t. the variables accordingly, see Remark 4, it follows that the resulting sums are analytically well defined.

To pass from the certificate recurrence to a homogeneous or inhomogeneous recurrences for the sum, special emphasis has to be put on the Δ-operators. In particular, the finite summation bounds appearing in (13) lead to an inhomogeneous right hand side after summing over the summand recurrence (41). A method to set up the inhomogeneous recurrences for the summation problems (13) was introduced in (Stan, 2010, Chapter 3). We summarize the steps of this approach implemented in the package FSums.

In this context, we use tuples to denote multi-dimensional intervals. The range represented by the tuple interval $[i, k]$ is the Cartesian product of the intervals defined by the components $i, k \in \mathbb{Z}^n$. More precisely, $[i, k] := [i_1, k_1] \times [i_2, k_2] \times \cdots \times [i_n, k_n]$ where $[i_j, k_j] = \{i_j, i_j + 1, \ldots, k_j\}$. Often when working with nested sums, summation ranges for inner sums will depend on the value of a variable for an outer sum. Intervals whose endpoints are defined by tuples are not enough to represent the summation ranges for these sums. We will use a variant of the cartesian product notation to denote such a summation range. Namely, to refer to a variable associated to a range, we will specify it as a subscript at the corresponding interval and use $\ltimes$ signs instead of the $\times$ symbols. For example, the range for the sum (18) can be written as $[0, \infty) \times [0, N-3]_{j_0} \ltimes [0, N - j_0 - 3] \ltimes [0, j_0 + 1]$. We also introduce this notation for the initial range of the sum (13) as

$$\mathcal{R} := \mathcal{R}_\sigma \times \mathcal{R}_j \tag{44}$$

where $\mathcal{R}_\sigma := [p, \infty)$ and $\mathcal{R}_j = [q_0, L_1(N)] \ltimes \cdots \ltimes [q_r, L_r(N, j_0, \ldots, j_{r-1})]$, are the infinite and the finite range, respectively.

Step 2.1: Refining the input sum. As indicated earlier, we consider the summands from (13) as well-defined only inside the initial input range $\mathcal{R} \subseteq D_\mathcal{F}$ where $D_\mathcal{F}$ denotes the set of well-defined values for the proper hypergeometric function $\mathcal{F}$. Because of this restriction we need to determine a possible smaller summation range over which we are allowed to sum the certificate recurrences (41).

Example 6. We illustrate this phenomenon by our concrete example (42). Let us start by summing over the initial summation range $\mathcal{R} = [0, N-3]_{j_0} \ltimes [0, N-3-j_0]$ over the delta parts on the right hand side of the recurrence (43) which is of the form (41). For this we denote the polynomial coefficients inside the delta parts Δ_{j_0} and Δ_{j_1} with $e(N, j_0, j_1, \varepsilon)$ and $d_1(N, j_0, j_1, \varepsilon)$, $d_2(N, j_0, j_1, \varepsilon)$, respectively. By summing over the first term inside the Δ_{j_1}-part and using the telescoping property, we have

$$\sum_{j_0=0}^{N-3} \sum_{j_1=0}^{N-3-j_0} \Delta_{j_1}[d_1(N, j_0, j_1, \varepsilon)\mathcal{F}(N, j_0, j_1)] = \sum_{j_0=0}^{N-3} \left. (d_1(N, j_0, j_1, \varepsilon)\mathcal{F}(N, j_0, j_1)) \right|_{j_1=0}^{j_1=N-2-j_0}$$

$$= \sum_{j_0=0}^{N-3} d_1(N, j_0, N-2-j_0, \varepsilon)\mathcal{F}(N, j_0, N-2-j_0) - \sum_{j_0=0}^{N-3} d_1(N, j_0, 0, \varepsilon)\mathcal{F}(N, j_0, 0)$$

18

where we use the short-hand notation $\sum_{k=0}^{l} \mathcal{F}(k,l)\big|_{l=A}^{l=B} := \sum_{k=0}^{B} \mathcal{F}(k,B) - \sum_{k=0}^{A} \mathcal{F}(k,A)$. We observe that, after telescoping, the upper bound $N - 2 - j_0$ for j_1 translates into a term outside the original summation range. To work under the assumption that our summand $\mathcal{F}(N, j_0, j_1)$ is well-defined only inside its range $\mathcal{R}$, we need to adjust the range over which we sum the certificate recurrence or shift this relation with respect to the free parameter N. As discussed in (Stan, 2010, Chapter 3), the approach based on computing a smaller admissible summation range is more efficient since it leads to fewer new sums in the inhomogeneous parts of the recurrences.

In the case of our example $\mathcal{S}(\varepsilon, N)$, we consider the new range $\mathcal{R}' = [0, N-4]_{j_0} \ltimes [0, N - j_0 - 4]$. As a consequence we compute separately a single sum which was called in (Stan, 2010, Chapter 3) a sore spot,

$$\mathcal{S}(\varepsilon, N) = \sum_{j_0=0}^{N-4} \sum_{j_1=0}^{N-4-j_0} \mathcal{F}(N, j_0, j_1) + \sum_{j_0=0}^{N-3} \mathcal{F}(N, j_0, N - j_0 - 3). \tag{45}$$

In general, the package `FSums` contains an algorithm that determines the inevitable summation range and computes the necessary sore spots for sums of the form (13); these extra sums with lower nesting depth have to be considered separately (see also the DIVIDE step in our method described in Section 5). Subsequently, we denote the sum over the restricted range $\mathcal{R}'$ by $\mathcal{S}'(\varepsilon, N)$.

Step 2.2: Determining the inhomogeneous part of the recurrence. Summing a certificate recurrence of the form (41) over the restricted range $\mathcal{R}'$ determined in the previous step leads to a recurrence for the new sum $\mathcal{S}'(\varepsilon, N)$. The inhomogeneous part contains special instances of this sum of lower nesting depth. Next, we introduce the types of sums appearing on the right hand side.

Step 2.2.1: The finite summation bounds. Shift compensating sums are the first side-effect of nonstandard summation bounds. They appear when we sum over the left hand side of the recurrence over a given definite range, because our upper summation bounds depend on the other summation parameters.

Example 7. Subsequently, we will illustrate these aspects with our running example (42). As deduced from Step 2.1, we continue from now on with the new sum

$$\mathcal{S}'(\varepsilon, N) = \sum_{j_0=0}^{N-4} \sum_{j_1=0}^{N-4-j_0} \mathcal{F}(N, j_0, j_1). \tag{46}$$

When we sum the certificate recurrence (43) over the restricted range $\mathcal{R}'$, we obtain

$$\sum_{j_0=0}^{N-4} \sum_{j_1=0}^{N-4-j_0} \mathcal{F}(N+1, j_0, j_1) = \mathcal{S}'(\varepsilon, N+1) - \sum_{j=0}^{N-3} \mathcal{F}(N+1, j, N-3-j). \tag{47}$$

Compensating sums of this form appear only in the case of upper summation bounds depending on the free variable N. After summing over the left hand side of the recurrence, we will move the resulting compensating sums, with a change of sign, to the inhomogeneous part.

Example 8. Including the new shifted sum as the first term of the output, the following procedure of `FSum` delivers the right hand side of (47)

19

In[1]:= **ShiftCompensatingSums**$[F[N, j_0, j_1], \{\{j_0, 0, N-4\}, \{j_1, 0, N-4-j_0\}\}, N, 1]$

Out[1]= $\mathrm{SUM}[N+1] + \mathrm{FSum}[-F[1+N, j_0, -3-j_0+N], \{\{j_0, 0, -3+N\}\}]$.

Note that we use the structure `FSum` to store sums with nonstandard boundary conditions of the form (13). This data type contains two components, the summand and a list structure for the summation range. The nested range is stored in the order given in (13), starting with the infinite sums and ending with the sums with finite summation bounds in the order of their dependence.

When summing over the Δ-parts we generate two types of sums on the right side of the recurrence, the Δ-boundary sums and the so-called telescoping compensating sums.

Example 9. When summing over the Δ_{j_0}-part of the recurrence (43), we get

$$\sum_{j_0=0}^{N-3} \sum_{j_1=0}^{N-3-j_0} \Delta_{j_0} [e(N, j_0, j_1, \varepsilon) \mathcal{F}(N+1, j_0, j_1)]$$

$$= \sum_{j_0=1}^{N-2} \sum_{j_1=0}^{N-2-j_0} e(N, j_0, j_1, \epsilon) F(N+1, j_0, j_1) - \sum_{j_0=0}^{N-3} \sum_{j_1=0}^{N-3-j_0} e(N, j_0, j_1, \epsilon) F(N+1, j_0, j_1).$$

Now one sees that exactly the sum with the summation index j_0 cancels and one obtains

$$\sum_{j_1=0}^{N-3-j_0} (e(N, j_0, j_1, \varepsilon) \mathcal{F}(N+1, j_0, j_1)) \Big|_{j_0=0}^{j_0=N-2} + \sum_{j_0=1}^{N-2} e(N, j_0, N-2-j_0, \varepsilon) \mathcal{F}(N+1, j_0, N-2-j_0).$$

Because of the structure of the summation bounds for the nested sums (13) we can use again our procedure `ShiftCompensatingSums` to generate the shift compensating sums and to read off the telescoping compensating sums. This connection becomes clearer when we consider the more involved sum (18) (with its restricted range $N-4$ instead of its original range $N-3$) and apply, e.g., the Δ_{j_0}-operator:

$$\sum_{\sigma_0=0}^{\infty} \sum_{j_0=0}^{N-4} \sum_{j_1=0}^{N-j_0-4} \sum_{j_2=0}^{j_0} \Delta_{j_0} [\mathcal{F}(N, \sigma_0, j_0, j_1, j_2)] = \sum_{\sigma_0=0}^{\infty} \sum_{j_1=0}^{N-j_0-4} \sum_{j_2=0}^{j_0} \mathcal{F}(N, \sigma_0, j_0, j_1, j_2) \Big|_{j_0=0}^{j_0=N-3}$$

$$+ \sum_{\sigma_0=0}^{\infty} \sum_{j_0=1}^{N-3} \sum_{j_2=0}^{j_0-1} \mathcal{F}(N, \sigma_0, j_0, N-j_0-3, j_2) - \sum_{\sigma_0=0}^{\infty} \sum_{j_0=1}^{N-3} \sum_{j_1=0}^{N-j_0-4} \mathcal{F}(N, \sigma_0, j_0, j_1, j_0);$$

note that the first element on the right side of this identity produces the Δ-boundary sums while the last two are due to telescoping compensation. More precisely, with

In[2]:= **ShiftCompensatingSums**$[F[N, \sigma_0, j_0-1, j_1, j_2], \{\{\sigma_0, 0, \infty\}, \{j_1, 0, N-j_0-4\},$
$\{j_2, 0, j_0\}\}/.j_0 \to (j_0-1), j_0, 1]$

Out[2]= $\{\mathrm{FSum}[F[N, \sigma_0, j_0, j_1, j_2], \{\{\sigma_0, 0, \infty\}, \{j_1, 0, N-4-j_0\}, \{j_2, 0, j_0\}\}], \mathrm{FSum}[F[N, \sigma_0, j_0, N-3-j_0, j_2], \{\{\sigma_0, 0, \infty\}, \{j_2, 0, j_0-1\}\}], \mathrm{FSum}[-F[N, \sigma_0, j_0, j_1, j_0], \{\{\sigma_0, 0, \infty\}, \{j_1, 0, N-4-j_0\}\}]\}$

we obtain exactly this result: the delta boundary sums are obtained by evaluating the first entry of the output for $j_0 = 0$ and $j_0 = N - 3$ and the compensating sums result by adding the shifted sum $[1, N-3]_{j_0}$ to the range of the other terms in the output. A detailed description of these computations can be found in (Stan, 2010, Alg. 4).

Step 2.2.2: The infinite summation bounds. To sum over the delta parts in (41) coming from the summation variables σ_i, e.g., $\Delta_{\sigma_i} b_{m,k,n}(N, \sigma, j, \varepsilon) \mathcal{F}(N+m, \sigma+k, j+n, \varepsilon)$ we

have to ensure that $\lim_{\sigma_i \to \infty} b_{m,k,n}(N, \sigma, j, \varepsilon)\mathcal{F}(N+m, \sigma+k, j+n, \varepsilon)$ exists. Looking at the asymptotic conditions (15) of the input sum (13), there will be no problem if $c_i > 0$. However, if the constant c_i is zero, we need to verify that the degrees of the polynomial coefficients $b_{m,k,n}$ appearing in the respective Δ_{σ_i}-part are smaller than the bound β_i. As worked out in Remark 4 this property is guaranteed by our ansatz.

The above sections introduced the types of sums, i.e., shift and telescoping compensating sums as well as delta boundary sums, which will appear on the right hand side of the inhomogeneous recurrences satisfied by summation problems of the form (13) after summing over corresponding certificate recurrences (41). A procedure to generate these inhomogeneous recurrences is implemented in the package FSums. E.g., the recurrence satisfied by the sum $\mathcal{S}'(\varepsilon, N)$, which we denote by SUM[N], is returned by

In[3]:= **finalRecS = InhomogenRec[certRecS, {{j_0, 0, $-4 + N$}, {j_1, 0, $-4 - j_0 + N$}}, N]**

Out[3]= $(\varepsilon - 2N)N\text{SUM}[N] + (3 - \varepsilon + N)(2 + \varepsilon + 2N)\text{SUM}[1 + N] ==$

$$\text{FSum}[(1 + j_0 - N)(-\varepsilon + 2N)F[N, j_0, 0], \{\{j_0, 0, -4 + N\}\}]+$$
$$\text{FSum}[-2(\varepsilon - 2N)F[N, j_0, -3 - j_0 + N], \{\{j_0, 0, -4 + N\}\}]+$$
$$\text{FSum}[(\varepsilon - 2N)(2 + j_0 - N)F[1 + N, j_0, 0], \{\{j_0, 0, -4 + N\}\}]+$$
$$\text{FSum}[(6 - \varepsilon - \varepsilon^2 + 2j_1 + 12N + 4j_1 N)F[1 + N, 0, j_1], \{\{j_1, 0, -4 + N\}\}]+$$
$$\text{FSum}[(3 - \varepsilon + N)(2 + \varepsilon + 2N)F[1 + N, j_0, -3 - j_0 + N], \{\{j_0, 0, -3 + N\}\}]+$$
$$\text{FSum}[(\varepsilon + \varepsilon^2 + 2j_0 + \varepsilon j_0 - 2N + 2j_0 N - 4N^2)F[1 + N, j_0, -3 - j_0 + N], \{\{j_0, 1, -3 + N\}\}]+$$
$$\text{FSum}[-((6 + 2\varepsilon + 2j_0 + \varepsilon j_0 + 6N - \varepsilon N + 2j_0 N - 2N^2)F[1 + N, j_0, -3 - j_0 + N]), \{\{j_0, 0, -4 + N\}\}];$$

here certRecS stands for the certificate recurrence (43).

5. An efficient approach to find ε-expansions for multi-sums

Let $\mathcal{S}(\varepsilon, N)$ be a multi-sum of the form (13) with the properties (1)–(5) from Assumption 1 and assume that $\mathcal{S}(\varepsilon, N)$ has a series expansion (14) for all $N \geq \lambda$ for some $\lambda \in \mathbb{N}$. Combining the methods of the previous sections we obtain the following general method to compute the first coefficients, say $F_t(N), \ldots, F_u(N)$ of (14).

Divide and conquer strategy
(1) BASE CASE: If $\mathcal{S}(\varepsilon, N)$ has no summation quantifiers, compute the expansion by formulas such as (8) and (9).
(2) DIVIDE: As worked out in Section 4, compute a recurrence relation

$$a_0(\varepsilon, N)\mathcal{S}(\varepsilon, N) + \cdots + a_d(\varepsilon, N)\mathcal{S}(\varepsilon, N + d) = h(\varepsilon, N) \tag{48}$$

with polynomial coefficients $a_i(\varepsilon, N) \in \mathbb{K}[\varepsilon, N]$, $a_m(\varepsilon, N) \neq 0$ and the right side $h(\varepsilon, N)$ containing a linear combination of hypergeometric multi-sums each with less than $s + r + 1$ summation quantifiers. Note: In some cases, the sum has to be refined and some "sore spots" (again with fewer summation quantifiers) have to be treated separately by calling our method again; see Step 2.1 in Section 4.
(3) CONQUER: Apply the strategy recursively to the simpler sums in $h(\varepsilon, N)$. This results in an expansion of the form

$$h(\varepsilon, N) = h_t(N)\varepsilon^t + h_{t+1}(N)\varepsilon^{t+1} + \cdots + h_u(N)\varepsilon^u + O(\varepsilon^{u+1}); \tag{49}$$

if the method fails to find the $h_t(N), \ldots, h_u(N)$ in terms of indefinite nested product-sum expressions, STOP.

(4) COMBINE: Given (48) with [3] (49), compute, if possible, the $F_t(N), \ldots, F_u(N)$ of (14) in terms of nested product-sum expressions by executing Algorithm FLSR.

We illustrate our method with the double sum (42); internally we transform all the objects in terms of $\Gamma(x)$-functions in order to apply expansion formulas such as (8) and (9). First, we compute the summand recurrence given in (43). While computing a recurrence for the sum itself, it turns out that we have to refine the summation range, i.e., our computation splits into two problems as given in (45). We continue with the refined double sum (46) and obtain the inhomogeneous recurrence `finalRecS` given in `Out[3]`. Now we apply recursively our method and compute successively expansions for each of the single sums on the right hand side; see also Example 2. Adding all the expansions termwise gives the recurrence

$$(\varepsilon - 2N)N\mathcal{S}'(\varepsilon, N) - (\varepsilon - N - 3)(\varepsilon + 2N + 2)\mathcal{S}'(\varepsilon, N+1) =$$
$$\frac{18(2N^6 - 3N^5 - 8N^4 + 13N^3 - 4N + 8)}{(N-2)(N-1)N(N+1)(N+2)} - \frac{36(2N^4 + N^3 - 9N^2 - 2N + 4)(-1)^N}{(N-2)(N-1)N(N+1)(N+2)}$$
$$+ \varepsilon\left[\frac{3(N^8 - 6N^7 - 32N^6 + 20N^5 + 151N^4 + 14N^3 - 200N^2 - 28N + 56)}{(N-2)(N-1)N(N+1)^2(N+2)^2}\right.$$
$$+ \frac{6(2N^6 + N^5 - 14N^4 + 9N^3 + 40N^2 - 22N - 28)(-1)^N}{(N-2)(N-1)N(N+1)^2(N+2)^2} + \left.\frac{36S_1(N)}{N+1}\right]$$
$$+ \varepsilon^2\left[\frac{9S_1(N)^2}{N+1} - \frac{6(N-5)S_1(N)}{(N+1)^2} - \frac{N^6(5N^3 + 48N^2 + 246N + 568)}{4(N-1)(N-2)(N+1)^3(N+2)^3}\right.$$
$$\left(\frac{9\left(N^4 - N^3 - 4N^2 + 4N + 8\right)}{(N-2)(N-1)N(N+1)(N+2)} - \frac{18\left(2N^4 + N^3 - 9N^2 - 2N + 4\right)(-1)^N}{(N-2)(N-1)N(N+1)(N+2)}\right)S_2(N)$$
$$+ + \left.\frac{363N^6 + 3720N^5 + 3672N^4 - 5280N^3 - 10712N^2 - 4592N - 128}{4N(N-1)(N-2)(N+1)^3(N+2)^3}\right] + O(\varepsilon^3).$$

Together with its first initial value $\mathcal{S}'(\varepsilon, 4) = \frac{27}{16} - \frac{1}{128}\varepsilon - \frac{11}{1024}\varepsilon^2$ Algorithm FLSR computes the series expansion of $\mathcal{S}'(\varepsilon, N)$. Finally, we compute the expansion of the extra sum $\sum_{j_0=0}^{N-3}\mathcal{F}(N, j_0, N-3-j_0)$ with our method, and adding this result to our previous computation leads to the final result

$$\mathcal{S}(\varepsilon, N) = \frac{81(N^2 - 3N + 2)}{4N^2} + \varepsilon\left[\frac{3(N^4 - 13N^3 - 28N^2 - 32N + 24)}{8N^3(N+2)} + \frac{9(N+3)S_1(N)}{N(N+1)(N+2)}\right]$$
$$+ \varepsilon^2\left[\frac{9(N+3)S_1(N)^2}{4N(N+1)(N+2)} - \frac{3(5N^3 + 36N^2 + 37N - 18)S_1(N)}{4N(N+1)^2(N+2)^2} + \frac{9(N^2 + 3N + 4)S_2(N)}{4N^2(N+1)(N+2)}\right.$$
$$- \left.\frac{5N^6 + 17N^5 + 162N^4 + 208N^3 + 592N^2 + 240N - 288}{32N^4(N+2)^2}\right] + O(\varepsilon^3).$$

Similarly, we compute, e.g., the first two coefficients of the expansion of the sum (18):

$$\mathcal{U}(\varepsilon, N) = \frac{3(-1)^N\left(N^2 + 2N - 1\right)S_1(N)}{N(N+1)} - 9(-1)^N + \frac{6\,(-1)^N S_2(N)}{N} +$$
$$\varepsilon\left[\zeta(2)\left(-\frac{3(-1)^N(4N+3)}{2N} - \frac{3(-1)^N(3N+2)\,S_{-1}(N)}{N} + \frac{9}{2N}\right) + \frac{3(-1)^N S_1(N)^2}{2\,(N+1)}\right.$$
$$+ \frac{(-1)^N\left(2N^4 + 34N^3 + 101N^2 + 89N + 2\right)S_1(N)}{2N(N+1)^2\,(N+2)} + \frac{3(-1)^N\left(4N^2 + 14N + 13\right)}{(N+1)(N+2)}$$
$$+ \frac{(-1)^N\left(-30N^2 - 38N + 1\right)S_2(N)}{2N(N+1)} - \frac{9(-1)^N(2N+1)S_3(N)}{N} + \frac{9\,S_{-2}(N)}{N} + 9(-1)^N S_{2,1}(N)$$

---

[3] Cf. Step 2 of Section 3.2 to see how we deal with the initial values.

$$- \frac{6(-1)^N(3N+2)S_{-2}(N)S_{-1}(N)}{N} + \frac{6(-1)^N(3N+2)\ S_{-2,-1}(N)}{N}\Bigg] + O(\varepsilon^2)$$

where $\zeta(2) = \sum_{i=1}^{\infty} \frac{1}{i^2} = \pi^2/6$.

Remark 5. In the following we give further comments on our proposed method and provide strategies for using it in the context of the evaluation of Feynman integrals.

1. A heuristic. The conquer step turns our procedure into a method and not into an algorithm. Knowing that there is an expansion of $\mathcal{S}(\varepsilon, N)$ in terms of indefinite nested sums and products and plugging this solution into the left hand side of (48) shows that also the right hand side of (48) can be written in terms of indefinite nested product-sum expressions. But in our method the right hand side is split into various sub-sums and it is not guaranteed that each sum on its own is expressible in terms of indefinite nested product-sum expressions – only the combination has this particular form. However, for our input class arising from Feynman-integrals this method always worked.

2. A hybrid version for speed–ups. As it turned out, the bottleneck in our computations is the task to compute a recurrence of the form (48) with the `MultiSum`-package. To be more precise, in several cases we succeeded in finding a structure set S with the corresponding degree bounds for the polynomial coefficients, but we failed to determine the summand recurrence (40) explicitly, since the underlying linear system was too large to solve. For such situations, we dropped, e.g., the outermost summation quantifier, say $\sum_{\sigma_1=p_1}^{\infty}$ and searched for a recurrence in σ_1; in particular the variable N was put in the base field $\mathbb{K}$. In this simpler form, we succeeded in finding a recurrence. Next, we computed the initial values (in terms of N) by using another round of our method. With this input, Algorithm FLSR found an expansion with coefficients in terms of $F_t(N,\sigma), F_{t+1}(N,\sigma), \dots, F_u(N,\sigma)$. To this end, we applied the infinite sum

$$\sum_{\sigma_1=p_1}^{\infty} F_i(\sigma, N) \tag{50}$$

to the coefficients $F_i(N,\sigma)$ and simplified these expressions further by the techniques described in Ablinger et al. (2011b). In various situations, it turned out that this hybrid technique was preferable to computing a pure recurrence in N or just simplifying the expressions (12) by using the methods given in Ablinger et al. (2011b).

3. Asymptotic expansions for infinite expressions. As mentioned in Remark 2 we obtained also sums of the form (13) which could be defined only by considering a truncated version of the infinite sums. For such cases we computed the coefficients $F_i(\sigma, N)$ as above and considered –instead of (50)– the expressions $\sum_{\sigma=0}^{a} F_i(\sigma, N)$ for large values a. To be more precise, we computed asymptotic expansions for all these sums and combined them to one asymptotic expansion in a. In this final form all the expressions canceled which were not defined when performing $a \to \infty$ and we ended up with the correct $F_i(N)$.

4. Dealing with several infinite sums. In all our computations only a single infinite sum arose. In principle, our method works also in the case when there are several such sums. However, in order to set up the recurrence in Section 4, we need additional properties such as (17) for the multivariate case. If such properties are not available, we propose two strategies: 4.1 Drop some (or all) of the infinite sums and proceed as explained in point 2 of our remark. 4.2 Set up the recurrence with formal sums and expand the sums on the right hand side: here one can either use the strategies as described in Step 4 of Section 2 (in particular, if asymptotic expansions have to be computed), or one can proceed with

the method of this section whenever the sum is analytically well defined.

6. Conclusion

We presented a general framework that enables one to compute the first coefficients $F_i(N)$ of the Laurent expansion of a given Feynman parameter integral, whenever the $F_i(N)$ are expressible in terms of indefinite nested product-sum expressions. Namely, starting from such integrals, we described a symbolic approach to obtain a multi-sum representation over hypergeometric terms. Given this representation, we developed symbolic summation tools to extract these coefficients from its sum representation. In order to tackle this problem, Wegschaider's `MultiSum` package has been enhanced with Stan's package `FSum` that handles sums which do not satisfy finite support conditions. Moreover, given a recurrence relation of the form (36) together with initial values, we used Schneider's recurrence solver that decides constructively, if the first coefficients of the formal Laurent series solution are expressible in terms of indefinite nested product-sum expressions.

In order to fit the input class of hypergeometric multi-sum packages, we split the sums at the price of possible divergencies. We overcame this situation by combining our new methods with other tools described, e.g., in Ablinger et al. (2011b); see Remark 5. Further analysis of the introduced method should lead to a uniform approach that can handle in one stroke also solutions in terms of asymptotic expansions.

The described summation tools assisted in the task to compute two- and simple three-loop diagrams, which occurred in the calculation of the massive Wilson coefficients for deep-inelastic scattering; see Ablinger et al. (2011b); Blümlein et al. (2006); Bierenbaum et al. (2007, 2009a, 2008). We are curious to see whether these new summation technologies find their application also in other fields of research.

References

Ablinger, J., February 2009. A computer algebra toolbox for harmonic sums related to particle physics. Master's thesis, RISC, Johannes Kepler University.

Ablinger, J., Bierenbaum, I., Blümlein, J., Hasselhuhn, A., Klein, S., Schneider, C., Wissbrock, F., 2010a. Heavy Flavor DIS Wilson coefficients in the asymptotic regime . Nucl. Phys. B (Proc. Suppl.) 205–206, 242–249, [arXiv:1007.0375 [hep-ph]].

Ablinger, J., Bluemlein, J., Schneider, C., 2011a. Harmonic sums and polylogarithms generated by cyclotomic polynomials. J. Math. Phys. 52(10), pp. 1-52, [arXiv:1105.6063v1 [math-ph]].

Ablinger, J., Blümlein, J., Klein, S., Schneider, C., 2010b. Modern Summation Methods and the Computation of 2- and 3-loop Feynman Diagrams . Nucl. Phys. B (Proc. Suppl.) 205-206, 110–115, arXiv::1006.4797 [math-ph], DESY-10-090, SFB-CPP-10-53, TTK-10-37.

Ablinger, J., Blümlein, J., Klein, S., Schneider, C., Wissbrock, F., 2011b. The $O(\alpha_s^3)$ Massive Operator Matrix Elements of $O(n_f)$ for the Structure Function $F_2(x, Q^2)$ and Transversity. Nucl. Phys. **B** 844, 26–54, arXiv:1008.3347 [hep-ph].

Abramov, S., Bronstein, M., Petkovšek, M., Schneider, C., 2011. In preparation.

Abramov, S., Petkovšek, M., 1994. D'Alembertian solutions of linear differential and difference equations. In: von zur Gathen, J. (Ed.), Proc. ISSAC'94. ACM Press, pp. 169–174.

Andrews, G. E., Askey, R., Roy, R., 1999. Special functions. Vol. 71 of Encyclopedia of Mathematics and its Applications. Cambridge University Press, Cambridge.

Apagodu, M., Zeilberger, D., 2006. Multi-variable Zeilberger and Almkvist–Zeilberger algorithms and the sharpening of Wilf–Zeilberger theory. Advances in Applied Math. 37, 139–152.

24

Bierenbaum, I., Blümlein, J., Klein, S., 2007. Two-loop massive operator matrix elements and unpolarized heavy flavor production at asymptotic values $Q^2 \gg m^2$. Nucl. Phys. B 780, 40–75, [arXiv:hep-ph/0703285].

Bierenbaum, I., Blümlein, J., Klein, S., 2009a. The gluonic operator matrix elements at $o(\alpha_s^2)$ for dis heavy flavor production. Phys. Lett. B 672, 401–406, [arXiv:0901.0669 [hep-ph]].

Bierenbaum, I., Blümlein, J., Klein, S., 2009b. Mellin Moments of the $O(\alpha_s^3)$ Heavy Flavor Contributions to unpolarized Deep-Inelastic Scattering at $Q^2 \gg m^2$ and Anomalous Dimensions. Nucl. Phys. B820, 417–482, [arXiv:0904.3563 [hep-ph]].

Bierenbaum, I., Blümlein, J., Klein, S., Schneider, C., 2008. Two-loop massive operator matrix elements for unpolarized heavy flavor production to $O(\epsilon)$. Nucl. Phys. B 803 (31-2), 1–41, [arXiv:0803.0273 [hep-ph]].

Blümlein, J., 2000. Analytic continuation of mellin transforms up to two-loop order. Comput. Phys. Commun. 133, 76–104, [arXiv:hep-ph/0003100].

Blümlein, J., 2004. Algebraic relations between harmonic sums and associated quantities. Comput. Phys. Commun. 159 (1), 19–54, [arXiv:hep-ph/0311046].

Blümlein, J., 2009. Structural relations of harmonic sums and Mellin transforms up to weight w = 5. Comput. Phys. Commun. 180, 2218–2249, [arXiv:0901.3106 [hep-ph]].

Blümlein, J., 2010. Structural relations of harmonic sums and Mellin transforms at weight $w = 6$. In: Motives, quantum field theory, and pseudodifferential operators. Vol. 12 of Clay Math. Proc. Amer. Math. Soc., pp. 167–187.

Blümlein, J., Broadhurst, D. J., Vermaseren, J. A. M., 2010. The multiple zeta value data mine. Comput. Phys. Commun. 181, 582–625, [arXiv:0907.2557 [math-ph]].

Blümlein, J., Freitas, A. D., van Neerven, W. L., Klein, S., 2006. The longitudinal heavy quark structure function $F_L^{Q\overline{Q}}$ in the region $Q^2 \gg m^2$ at $O(\alpha_s^3)$. Nucl. Phys. B **755** (2006) 272–285 755, 272–285, [arXiv:hep-ph/0608024].

Blümlein, J., Kauers, M., Klein, S., Schneider, C., 2009. Determining the closed forms of the $O(a_s^3)$ anomalous dimensions and Wilson coefficients from Mellin moments by means of computer algebra. Comput. Phys. Commun. 180, 2143–2165, [arXiv:0902.4091 [hep-ph]].

Blümlein, J., Kurth, S., 1999. Harmonic sums and Mellin transforms up to two-loop order. Phys. Rev. D60, 014018, [arXiv:hep-ph/9810241].

Blümlein, J., Moch, S. O., 2005. Analytic continuation of the harmonic sums for the 3-loop anomalous dimensions 614, 53–61, [arXiv:hep-ph/0503188].

Bogner, C., Weinzierl, S., 2010. Feynman graph polynomials. Int. J. Mod. Phys. A 25, 2585–2618, [arXiv:1002.3458 [hep-ph]].

Broadhurst, D. J., 1999. Massive 3-loop Feynman diagrams reducible to SC* primitives of algebras of the sixth root of unity. Eur. Phys. J. 8, 311–333, [arXiv:hep-th/9803091].

Chyzak, F., 2000. An extension of Zeilberger's fast algorithm to general holonomic functions. Discrete Math. 217 (1-3), 115–134, fPSAC 1997.

Fasenmyer, M. C., November 1945. Some generalized hypergeometric polynomials. Ph.D. thesis, University of Michigan.

Feynman, R., 1949. Space-time approach to quantum electrodynamics. Phys. Rev. D76, 769–789.

Gosper, Jr., R. W., 1978. Decision procedure for indefinite hypergeometric summation. Proc. Nat. Acad. Sci. U.S.A. 75 (1), 40–42.

Karr, M., 1981. Summation in finite terms. J. ACM 28 (2), 305–350.

Kauers, M., Schneider, C., 2006. Indefinite summation with unspecified summands. Discrete Math. 306 (17), 2021–2140.

Koutschan, C., September 2009. Advanced applications of the holonomic systems approach. Ph.D. thesis, RISC, Johannes Kepler University.

Moch, S., Uwer, P., Weinzierl, S., 2002. Nested sums, expansion of transcendental functions, and multiscale multiloop integrals. J. Math. Phys. 43, 3363–3386.

Naas, J., Schmid, H. (Eds.), 1961. Mathematisches Wörterbuch. Vol. II. Teubnerberlag Leipzig

and Akademie Verlag, Berlin.

Paris, R., Kaminski, D., 2001. Asymptotics and Mellin-Barnes Integrals. Vol. 85 of Encyclopedia of Mathematics and its Applications. Cambridge University Press, Cambridge.

Paule, P., 1995. Greatest factorial factorization and symbolic summation. J. Symbolic Comput. 20 (3), 235–268.

Paule, P., Schorn, M., 1995. A Mathematica Version of Zeilberger's Algorithm for Proving Binomial Coefficient Identities. J. Symbolic Comput. 20, 673–698.

Petkovšek, M., 1992. Hypergeometric solutions of linear recurrences with polynomial coefficients. J. Symb. Comp. 14, 243–264.

Petkovšek, M., Wilf, H. S., Zeilberger, D., 1996. $A = B$. A. K. Peters, Wellesley, MA.

Schneider, C., May 2001. Symbolic Summation in Difference Fields. Ph.D. thesis, RISC, J. Kepler University Linz.

Schneider, C., 2005a. A new Sigma approach to multi-summation. Advances in Applied Math. 34 (4), 740–767.

Schneider, C., 2005b. Solving parameterized linear difference equations in terms of indefinite nested sums and products. J. Differ. Equations Appl. 11 (9), 799–821.

Schneider, C., 2007. Symbolic summation assists combinatorics. Sem. Lothar. Combin. 56, 1–36.

Schneider, C., 2008. A refined difference field theory for symbolic summation. J. Symbolic Comput. 43 (9), 611–644.

Schneider, C., 2010. A symbolic summation approach to find optimal nested sum representations. In: Motives, quantum field theory, and pseudodifferential operators. Vol. 12 of Clay Math. Proc. Amer. Math. Soc., pp. 285–308.

Stan, F., 2010. Algorithms for special functions: Computer algebra and analytical aspects. Ph.D. thesis, RISC, Johannes Kepler University Linz.

Vermaseren, J., 1999. Harmonic sums, Mellin transforms and integrals. Int. J. Mod. Phys. A14, 2037–2976.

Wegschaider, K., May 1997. Computer generated proofs of binomial multi-sum identities. Master's thesis, RISC, Johannes Kepler University.

Whittaker, E., Watson, G., 1996. A Course of Modern Analysis. Cambridge Mathematical Library Series. Cambridge University Press, Cambridge.

Wick, G., 1950. The evaluation of the collision matrix. Phys. Rev. 80, 268–272.

Wilf, H. S., Zeilberger, D., 1992. An algorithmic proof theory for hypergeometric (ordinary and "q") multisum/integral identities. Invent. Math. 108 (3), 575–633.

Zeilberger, D., 1990a. A fast algorithm for proving terminating hypergeometric identities. Discrete Mathematics 80 (2), 207–211.

Zeilberger, D., 1990b. A holonomic systems approach to special functions identities. J. Comput. Appl. Math. 32, 321–368.