A toolbox to solve coupled systems of differential and difference equations[*]

Jakob Ablinger and Carsten Schneider[†]

Research Institute for Symbolic Computation (RISC)
Johannes Kepler University, Altenbergerstraße 69, A-4040 Linz, Austria
E-mail: Jakob_Ablinger@risc.jku.at, Carsten.Schneider@risc.jku.at

Johannes Blümlein and Abilio de Freitas

Deutsches Elektronen–Synchrotron, DESY,
Platanenallee 6, D–15738 Zeuthen, Germany
E-mail: johannes.bluemlein@desy.de, abilio.de.freitas@desy.de

We present algorithms to solve coupled systems of linear differential equations, arising in the calculation of massive Feynman diagrams with local operator insertions at 3-loop order, which do *not* request special choices of bases. Here we assume that the desired solution has a power series representation and we seek for the coefficients in closed form. In particular, if the coefficients depend on a small parameter ε (the dimensional parameter), we assume that the coefficients themselves can be expanded in formal Laurent series w.r.t. ε and we try to compute the first terms in closed form. More precisely, we have a decision algorithm which solves the following problem: if the terms can be represented by an indefinite nested hypergeometric sum expression (covering as special cases the harmonic sums, cyclotomic sums, generalized harmonic sums or nested binomial sums), then we can calculate them. If the algorithm fails, we obtain a proof that the terms cannot be represented by the class of indefinite nested hypergeometric sum expressions. Internally, this problem is reduced by holonomic closure properties to solving a coupled system of linear difference equations. The underlying method in this setting relies on decoupling algorithms, difference ring algorithms and recurrence solving. We demonstrate by a concrete example how this algorithm can be applied with the new Mathematica package `SolveCoupledSystem` which is based on the packages `Sigma`, `HarmonicSums` and `OreSys`. In all applications the representation in *x*-space is obtained as an iterated integral representation over general alphabets, generalizing Poincaré iterated integrals.

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

[†]Speaker.

1. Introduction

A massive three loop Feynman diagram $I(N)$ with a local operator insertion can be written in terms of multiple integrals or multiple sums, which depend on a discrete variable $N \in \mathbb{N}$ and the dimensional parameter $\varepsilon = D - 4$, where $D \in \mathbb{R}$ denotes the space-time dimension, see e.g. see [1, 2]. Then one is interested in the first coefficients of its formal Laurent series w.r.t. ε (in short ε-expansion)

$$I(N) = \varepsilon^o I_o(N) + \varepsilon^{o+1} I_{o+1}(N) + \varepsilon^{o+2} I_{o+2}(N) + \dots \tag{1.1}$$

with order $o \in \mathbb{Z}$. In this article we present tools to decide algorithmically if the coefficients $I_i(N)$ up to a certain order can be written in terms of indefinite nested hypergeometric sums (in short, nested hypergeometric sums) which can be defined as follows. Let $f(N)$ be an expression that evaluates at non-negative integers (from a certain point on) to elements of a field $\mathbb{K}$ containing the rational numbers $\mathbb{Q}$. Then $f(N)$ is called a nested hypergeometric sum expression w.r.t. N if it is composed by elements from the rational function field $\mathbb{K}(N)$, the three operations $(+, -, \cdot)$, hypergeometric expressions of the form $\prod_{k=l}^{N} h(k)$ with $l \in \mathbb{N}$ and $h(k)$ being a rational function in k and being free of N, and sums of the form $\sum_{k=l}^{N} h(k)$ with $l \in \mathbb{N}$ and with $h(k)$ being a nested hypergeometric sum expression w.r.t. k and being free of N. This class of special functions covers as special cases harmonic sums [3],

$$S_{a_1,\dots,a_k}(N) = \sum_{N \geq i_1 \geq i_2 \geq \dots \geq i_k \geq 1} \frac{\text{sign}(a_1)^{i_1}}{i_1^{|a_1|}} \cdots \frac{\text{sign}(a_k)^{i_k}}{i_k^{|a_k|}} \tag{1.2}$$

for non-negative integers N and nonzero integers a_i $(1 \leq i \leq k)$ and more generally, generalized harmonic sums [4, 5], cyclotomic harmonic sums [6] or nested binomial sums [7, 8].[1]

In order to calculate the coefficients in (1.1), the summation package `Sigma` enhanced by the package `EvaluateMultiSums` [10, 11], the integration package `MultiIntegrate` [12], and the package `HarmonicSums` [13] have been applied successfully in many applications. However, in the course of recent calculations, these tools turned out to be not sufficient and we extended them significantly by using the integration by parts (IBP) identities [14]. Namely, encoding $I(N)$ by a generating function (formal power series)

$$\hat{I}(x) = \sum_{N=0}^{\infty} I(N) x^N,$$

we can activate the powerful `C++` program `Reduze 2` [15] based on Laporta's algorithm [16] to reduce $\hat{I}(x)$ to a linear combination of master integrals. In many calculations, see e.g., [17, 18, 19] these remaining integrals are now suitable for symbolic summation and integration. However, some of the master integrals are rather hard too handle or are not in the proper form for symbolic summation and integration. Here we utilize the fact that `Reduze 2` can produce recursively defined coupled systems of linear differential equations in terms of these master integrals. Following the tactics in [20] the main task is to extract the required information from these coupled systems and to reassemble information of the coefficients $I_i(N)$ in (1.1) for further processing. In this article

---

[1]For surveys on these quantities see e.g. [9].

we are interested in computing the $I_i(N)$ in closed form. If a first-order coupled system has a specific form, one could use, e.g., the methodology described in [21]. In the following we introduce a very general and efficient approach relying on decoupling algorithms [22, 23, 24], recurrence solvers [25, 26] and difference ring algorithms [27, 28, 29]: we obtain a complete algorithm that extracts the first coefficients of the Laurent series and computes simultaneously the representation of the coefficients in terms of nested hypergeometric sum expressions, whenever this is possible. The first ideas of this new algorithm have been introduced in [30] and the main features are worked out in [19] by concrete examples coming from massive 3-loop ladder and *V*-diagrams. In the following we will complement these achievements by precise input-output specifications and further details of the algorithms. Moreover, we will illustrate how the differential equation algorithm can be executed within the new package `SolveCoupledSystem` that relies on the packages Sigma and `OreSys`. In addition, the package `HarmonicSums` is used to gain significant speed-ups.

We will use the following notations. Let $\mathbb{K}$ be a computable field containing the rational numbers as sub-field (e.g., $\mathbb{K} = \mathbb{Q}$). In the following $\mathbb{K}[N]$ (or $\mathbb{K}[\varepsilon, N]$) denotes the ring of polynomials in the variable N (or in the variables ε and N). Moreover, $\mathbb{K}(N)$ (or $\mathbb{K}(\varepsilon, N)$) denotes the field of rational functions in the variable N (or in the variables ε and N). We denote by $\mathbb{K}((\varepsilon))$ the field of formal Laurent series, i.e., elements are of the form $\sum_{k=o}^{\infty} f_k \varepsilon^k$ with $f_k \in \mathbb{K}$ and $o \in \mathbb{Z}$. Furthermore, we denote by $\mathbb{K}((\varepsilon))[[x]]$ the ring of power series whose elements are of the form $\sum_{i=0}^{\infty} f_i x^i$ with $f_i \in \mathbb{K}((\varepsilon))$. Furthermore, $\mathbb{K}^{\mathbb{N}}$ (or $\mathbb{K}((\varepsilon))^{\mathbb{N}}$) denotes the ring of sequences with entries form $\mathbb{K}$ (or from $\mathbb{K}((\varepsilon))$).

2. The algorithmic machinery

First, we will address the problem how one can decide algorithmically, if a sequence can be calculated by a nested hypergeometric sum expression provided that the sequence is described by a linear recurrence (linear difference equation) in terms of nested hypergeometric sum expressions (see Theorem 1). Given this technology, we can extract the first coefficients of a Laurent series expansion in terms of nested hypergeometric sum expressions provided that the Laurent series is a solution of a recurrence of certain kind (see Theorem 2). Using uncoupling algorithms, this result can be generalized further to coupled systems (see Theorem 3). Finally, we can carry over this result to coupled systems of linear differential equations (see Theorem 4).

2.1 Finding nested hypergeometric solutions of linear recurrences

The main engine relies on the following algorithmic result [11, 29, 25, 26].

Theorem 1. Suppose that a sequence $\langle I(N) \rangle_{N \geq 0} \in \mathbb{K}^{\mathbb{N}}$ is a solution of the difference equation

$$a_0(N)I(N) + a_1(N)I(N+1) + \cdots + a_d(N)I(N+d) = r(N) \tag{2.1}$$

with $N \geq 0$ for given rational functions $a_0(N), \ldots, a_d(N) \in \mathbb{K}(N)$, not all zero, and a nested hypergeometric sum expression $r(N)$. Then one can determine an $m \in \mathbb{N}$ with the following property. If one is given the values $I(N)$ for all $0 \leq N \leq m$, then one can decide algorithmically if there exists a nested hypergeometric sum expression that calculates the values $I(N)$ for all $N \geq 0$ (or at least from a certain point on).

Proof. For this result we refer to Section 4.3 in [11] and Section 2.4 [29]. It has been implemented within $\mathtt{Sigma}$ as follows. The recurrence operator is factorized as much as possible into linear factors. Then each linear factor leads to one additional linearly independent solution of the homogeneous version of the recurrence by introducing one extra indefinite summation quantifier and introducing one hypergeometric expression [25]. In this way one finds a basis of all solutions of the homogeneous recurrence that can be expressed in terms of nested hypergeometric sum expressions. If the recurrence factorizes completely, the particular solution can be obtained straightforwardly. However, if the recurrence does not fully factorize, one has to activate algorithms from [26] in order to calculate a particular solution in terms of nested hypergeometric sum expressions or to prove that such a representation is not possible. If there is not such a solution, $I(N)$ cannot be represented by a nested hypergeometric sum expression.

In the process of this calculation one can determine a $\mu \in \mathbb{N}$ such that the solutions in terms of nested hypergeometric sum expressions can be evaluated and are a solution of the recurrence. Now compute the finite set R of all non-negative integer roots of $a_d(N) \in \mathbb{K}[N]$. If $R = \{\}$, set $\mu' = \mu$, else set $\mu' := \max(1 + \max(R), \mu)$. Thus for all $\lambda \in \mathbb{N}$ with $\lambda \geq \mu'$ we have that $a_d(\lambda) \neq 0$. Then using d initial values, namely $I(\mu'), \ldots, I(\mu' + d - 1)$, one can check if the found solutions can be combined to an expression which produces the same initial values. If this is possible, this expression agrees with $I(N)$ for all $N \geq \mu'$: since the leading coefficient of (2.1) does not evaluate to zero, there is exactly one sequence which has these d initial values and which is a solution of the recurrence. Note that this construction is always successful if one computes d linearly independent solutions of the input recurrence (the particular solution is then just a by-product). Otherwise, if this construction fails, it follows that $I(N)$ cannot be expressed by a nested hypergeometric sum expression (i.e., that at least one solution of the homogeneous recurrence is of different nature). Summarizing, if we can compute the first $m := \mu' + d - 1$ initial values of $I(N)$, we can execute the decision procedure described above.

Example. Consider the sequence $I(N)$ that is determined by the linear recurrence

$\mathsf{In[1]{:}=}$ $\mathbf{rec} = -2(\mathbf{N}+1)(\mathbf{N}+2)^2\mathbf{I[N]} - (\mathbf{N}+2)\left(-6\mathbf{N}^2 - 28\mathbf{N} - 32\right)\mathbf{I[N+1]}$
$+ \left(-6\mathbf{N}^3 - 50\mathbf{N}^2 - 136\mathbf{N} - 120\right)\mathbf{I[N+2]} - (-\mathbf{N}-2)(\mathbf{N}+4)(2\mathbf{N}+8)\mathbf{I[N+3]} == -\dfrac{4(\mathbf{N}+2)}{3(\mathbf{N}+3)};$

(loaded into Mathematica) and the initial values $I(1) = 5$, $I(2) = \frac{130}{27}$, $I(3) = \frac{169}{36}$. Loading in the summation package Sigma into Mathematica, one can solve the recurrence in terms of nested hypergeometric sum expressions as follows.

$\mathsf{In[2]{:}=}$ $\ll \mathbf{Sigma}.\mathbf{m}$

Sigma - A summation package by Carsten Schneider © RISC-Linz

$\mathsf{In[3]{:}=}$ $\mathbf{recSol} = \mathbf{SolveRecurrence}[\mathbf{rec}, \mathbf{I}_{1,-3}[\mathbf{N}], \mathbf{IndefiniteSummation} \rightarrow \mathbf{False}]$

$\mathsf{Out[3]{=}}$ $\left\{\left\{0, \dfrac{1}{-N-1}\right\}, \left\{0, -\dfrac{\sum_{i=1}^{N} 1}{N+1}\right\}, \left\{0, -\dfrac{\sum_{i=1}^{N}\sum_{j=1}^{i}\frac{1}{j(1+j)}}{N+1}\right\}, \left\{1, -\dfrac{\sum_{i=1}^{N}\sum_{j=1}^{i}\frac{\sum_{k=1}^{j}\frac{2}{3}}{j(1+j)}}{N+1}\right)\right\}\right\}$

Here the first three entries are linearly independent solutions of the homogeneous version of the recurrence and the last entry is a particular solution of the recurrence itself.

Within $\mathtt{Sigma}$ a strong toolbox has been developed to simplify these solutions by flattening the sums optimally and by finding denominators with minimal degrees within the setting of difference rings [10, 11]. In particular, the simplified expressions are built by sums that are algebraically inde-

4

pendent [28]. These features can be activated by dropping `IndefiniteSummation→False` in ln[3]. For the class of harmonic sums, generalized harmonic sums, cyclotomic harmonic sums and binomial nested sums these features are also available within the package

ln[4]:= << **HarmonicSums.m**

> HarmonicSums by Jakob Ablinger © RISC-Linz

For such sums the rather involved difference ring theory can be avoided, and one can calculate very efficiently the simplified representation as follows:

ln[5]:= **recSol = ReduceToBasic[TransformToSSums[recSol], Dynamic → Automatic]**

Out[5]= $\{\{0, \frac{1}{-N-1}\}, \{0, -\frac{N}{N+1}\}, \{0, \frac{1}{(N+1)^2} + \frac{S_1(N)}{N+1}\}, \{1, \frac{2(N^2+N-1)}{3(N+1)^2} - \frac{2(N+2)}{3(N+1)}S_1(N)\}\}$

Remark. Internally, only the so-called basis-sums remain that cannot be eliminated by relations induced by the underlying quasi-shuffle algebra. For harmonic sums these ideas are worked out in [31] and have been extended for cyclotomic sums, generalized harmonic sums and nested binomial sums [13, 5, 6, 8]. We remark further that the basis sums produce sequences which are algebraically independent [32].

Summarizing, the solution set of Out[5] is given by the set

$$\{c_1 \frac{-1}{N+1} + c_2 \frac{-N}{N+1} + c_3 \left(\frac{1}{(N+1)^2} + \frac{S_1(N)}{N+1}\right) + \frac{2(N^2+N-1)}{3(N+1)^2} - \frac{2(N+2)}{3(N+1)}S_1(N) | c_1, c_2, c_3 \in \mathbb{K}\}$$

of nested hypergeometric sum expressions. The initial values $I(1), I(2), I(3)$ can be fulfilled with $c_1 = -\frac{49}{9}, c_2 = -\frac{41}{9}, c_3 = -\frac{2}{3}$ which yields

$$I(N) = \frac{59N^2 + 120N + 49}{9(N+1)^2} - \frac{2(N+3)S_1(N)}{3(N+1)}. \tag{2.2}$$

Hence we have shown that $I(N)$ can be calculated for $N \geq 0$ by a nested hypergeometric expression.

2.2 Finding Laurent series solutions of linear difference equations

During the calculation of a Feynman integral $I(N)$ one often obtains linear recurrences for $I(N)$ depending on a dimensional parameter ε with $D = 4 + \varepsilon \in \mathbb{R}$. In lucky situations one finds a nested hypergeometric sum representation of $I(N)$ using the recurrence solver of Subsection 2.1 with $\varepsilon \in \mathbb{K}$. However, in most cases one fails to find any solution of the given recurrence in terms of nested hypergeometric sum expressions, but one finds an ε-expansion whose coefficients can be represented by nested hypergeometric expressions using the following algorithmic machinery [1].

Theorem 2. Suppose that the sequence $\langle I(N) \rangle_{N \geq 0} \in \mathbb{K}((\varepsilon))^{\mathbb{N}}$ with (1.1) is a solution of the difference equation

$$a_0(\varepsilon, N)I(N) + a_1(\varepsilon, N)I(N+1) + \cdots + a_d(\varepsilon, N)I(N+d) = r(N) \tag{2.3}$$

for explicitly given $a_0(\varepsilon, N), \ldots, a_d(\varepsilon, N) \in \mathbb{K}(\varepsilon, N)$ and for a sequence $\langle r(N) \rangle_{N \geq 0} \in \mathbb{K}((\varepsilon))^{\mathbb{N}}$ with

$$r(x) = \varepsilon^o r_o(N) + \varepsilon^{o+1} r_{o+1}(N) + \varepsilon^{o+2} r_{o+2}(N) + \ldots \tag{2.4}$$

Here we assume[2] that the $a_i(\varepsilon, N)|_{\varepsilon \to 0}$ do not introduce poles for $0 \leq i \leq d$ and that not all $a_i(0, N)$ are zero. Then for any $u \in \mathbb{Z}$ one can determine an $m \in \mathbb{N}$ with the following property.

---

[2]This assumption can be always guaranteed by multiplying an appropriate factor ε^s with $s \in \mathbb{Z}$ on both sides of (1.1).

If one is given the values $I_j(N)$ for all $o \leq j \leq u$ and $0 \leq N \leq m$ and one is given for all $o \leq j \leq u$ nested hypergeometric sum expressions that calculate the values $r_j(N)$ for all $N \geq 0$ (or at least from a certain point on), then one can decide algorithmically if for all $o \leq j \leq u$ there exist nested hypergeometric sum expressions that calculate the values $I_j(N)$ for all $N \geq 0$ (or at least from a certain point on).

Proof. Let $u \in \mathbb{Z}$ and suppose that the $r_o(N), \dots, r_u(N)$ can be represented in terms of nested hypergeometric sum expressions. We make the Ansatz (1.1) with unknown coefficients $I_j(N)$ and plug them into (2.3). Then the left and right hand sides are both Laurent series which are equal if and only if the coefficients agree. In particular, the lowest term must agree, i.e., we obtain the following constraint

$$a_0(0,N)I_o(N) + a_1(0,N)I_o(N+1) + \dots + a_d(0,N)I_o(N+d) = r_o(N) \tag{2.5}$$

which is a linear recurrence of order $d' = \max\{0 \leq i \leq d | a_i(0,N) \neq 0\} \geq 0$. Activating Theorem 1 to this recurrence with appropriately chosen initial values (given by Theorem 1) one can decide algorithmically if $I_o(N)$ is expressible by a nested hypergeometric sum expression. If such a representation is not possible, the theorem is proven. Otherwise, we take this representation and make the Ansatz (1.1) with the known coefficient $I_o(N)$ and the unknown coefficients $I_j(N)$ ($j > o$) and plug them into (2.3). Then by construction the coefficients of ε^o on the left and right hand sides agree and the term of ε^o can be eliminated by subtracting it on both sides. Now one repeats this process for the next lowest term. In this way one can decide algorithmically if all $I_o(N), \dots, I_u(N)$ can be represented by nested hypergeometric sum expressions. In the process of this construction we choose $m \in \mathbb{Z}$ such that the used initial values are covered by $I_j(0), \dots, I_j(m)$ with $o \leq j \leq u$.

Example. Take $\langle I(N) \rangle_{N \geq 0}$ with the ε-expansion (1.1) of order $o = -3$ where the first two coefficients are determined by the initial values

$$I(1) = \frac{5}{\varepsilon^3} - \frac{163}{12\varepsilon^2} + O(\varepsilon^{-1}), \quad I(2) = \frac{130}{27\varepsilon^3} - \frac{695}{54\varepsilon^2} + O(\varepsilon^{-1}), \quad I(3) = \frac{169}{36\varepsilon^3} - \frac{395}{32\varepsilon^2} + O(\varepsilon^{-1}) \tag{2.6}$$

and the linear recurrence

In[6]:= $\mathbf{recEp} = -2(N+1)(N+2)(2+\varepsilon+N)\mathbf{I[N]} - (N+2)(-32-7\varepsilon+2\varepsilon^2-28N-5\varepsilon N-6N^2)\mathbf{I[N+1]} - (120+3\varepsilon-14\varepsilon^2-\varepsilon^3+$

$136N+13\varepsilon N-4\varepsilon^2N+50N^2+4\varepsilon N^2+6N^3)\mathbf{I[N+2]} + (2-\varepsilon+N)(4+\varepsilon+N)(8+\varepsilon+2N)\mathbf{I[N+3]}$

$== \frac{1}{\varepsilon^3}\frac{-4(N+2)}{3(N+3)} + \frac{1}{\varepsilon^2}\left[-\frac{2(2N+7)S_1}{3(N+3)} - \frac{2(4N^4+35N^3+101N^2+105N+25)}{3(N+1)(N+2)(N+3)^2}\right] + O(\varepsilon^{-1}).$

We seek to calculate a nested hypergeometric sum representation for I_{-3}, and I_{-2}, i.e., we set $u = -2$. Note that the expansion on the right hand side of In[6] is sufficiently high expanded. The recurrence (2.5) in our concrete instance is precisely In[1] with $I_{-3}(N) = I(N)$. In addition, the initial values agree with (2.6). Hence the found nested hypergeometric sum expression (2.2) represents $I_{-3}(N)$. Continuing this process we can calculate the coefficient $I_{-2}(N)$. Within `Sigma` this calculation can be carried out automatically with the function call

In[7]:= $\mathbf{GenerateExpansion}[\mathbf{recEp}[[1]], \{\mathbf{Coefficient}[\mathbf{recEp}[[2]], \varepsilon^{-3}], \mathbf{Coefficient}[\mathbf{recEp}[[2]], \varepsilon^{-2}]\}, \mathbf{I[N]}, \{\varepsilon, -3, -2\},$

$$\{\{5, \frac{130}{27}, \frac{169}{36}\}, \{-\frac{163}{12}, -\frac{695}{54}, -\frac{395}{32}\}\}, \mathbf{MinInitialValue} \rightarrow \mathbf{1}]$$

Out[7]= $\{\frac{59N^2+120N+49}{9(N+1)^2} - \frac{2(N+3)S_1(N)}{3(N+1)}, -\frac{2(20N^3+58N^2+57N+22)}{3(N+1)^3} + \frac{2(N+2)(2N-1)S_1}{3(N+1)^2} - \frac{S_1^2}{N+1} - \frac{S_2}{N+1}\}$

Remark. For further details and speed-ups we refer to [1].

2.3 Finding Laurent series solutions of coupled systems of linear difference equations

We generalize Theorem 2 to solve coupled systems as follows [19].

Theorem 3. Suppose that the sequences $\langle I_1(N)\rangle_{N\geq0}\rangle, \ldots, \langle I_n(N)\rangle_{N\geq0}\rangle \in (\mathbb{K}((\varepsilon)))^{\mathbb{N}}$ with

$$I_i(N) = \varepsilon^o I_{i,o}(N) + \varepsilon^{o+1} I_{i,o+1}(N) + \varepsilon^{o+2} I_{i,o+2}(N) + \ldots$$

are solutions of the coupled system of difference equations

$$A_0 \begin{pmatrix} I_1(N) \\ \vdots \\ I_n(N) \end{pmatrix} + A_1 \begin{pmatrix} I_1(N+1) \\ \vdots \\ I_n(N+1) \end{pmatrix} \cdots + A_d \begin{pmatrix} I_1(N+d) \\ \vdots \\ I_n(N+d) \end{pmatrix} = \begin{pmatrix} r_1(N) \\ \vdots \\ r_n(N) \end{pmatrix} \tag{2.7}$$

for explicitly given $n \times n$ matrices $A_0, \ldots, A_d$ with entries from $\mathbb{K}(\varepsilon, N)$ and for some sequences $\langle r_1(N)\rangle_{N\geq0}, \ldots, \langle r_n(N)\rangle_{N\geq0} \in \mathbb{K}((\varepsilon))^{\mathbb{N}}$ with

$$r_i(N) = \varepsilon^o r_{o,i}(N) + \varepsilon^{o+1} r_{o+1,i}(N) + \varepsilon^{o+2} r_{o+2,i}(N) + \ldots \tag{2.8}$$

Then for any $u_1, \ldots, u_n \in \mathbb{Z}$ one can determine $m_1, \ldots, m_n \in \mathbb{N}$ $v_1, \ldots, v_n \in \mathbb{Z}$ and $w_1, \ldots, w_n \in \mathbb{Z}$ with the following property.
If one is given the values $I_{i,j}(N)$ for all $1 \leq i \leq n$, $o \leq j \leq v_i$ and $0 \leq N \leq m_i$ and one is given for all $1 \leq i \leq n$ and $o \leq j \leq w_i$ nested hypergeometric sum expressions that calculate the values $r_{i,j}(N)$ for all $N \geq 0$ (or at least from a certain point on), then one can decide algorithmically if for all $1 \leq i \leq n$ and $o \leq j \leq u_i$ there are nested hypergeometric sum expressions that calculate the values $I_{i,j}(N)$ for all $N \geq 0$ (or at least from a certain point on).

Proof. The algorithmic steps can be summarized as follows. First, we transform the coupled system (2.7) to a first order system as explained in [19]. Then we can apply any decoupling algorithm from [22] to uncouple the system, e.g., w.r.t. $I_1(N)$. In our implementation we chose Zürcher's algorithm [23] implemented in the package `OreSys` [24]. In the generic case one obtains one linear recurrence in $I_1(N)$ which is of the form (2.4). We can assume that the evaluations $a_i(0,N)$ are possible and that not all $a_i(0,N)$ are zero (see Footnote 2). In addition, the decoupling algorithm expresses the remaining sequences $I_2(N), \ldots, I_n(N)$ by a linear combination of the shifted versions of $I_1(N)$ and shifted versions of the (2.8).
A subtle point is to which order ε^{w_i} the expressions in (2.8) should be expanded. To extract this knowledge, we decouple the system by considering $r_1(N), \ldots, r_n(N)$ first as unspecified sequences. Then analysing the corresponding output gives an upper bound for the w_i; details on these aspects can be found in [19]. Since the description of the $I_2(N), \ldots, I_n(N)$ is given in terms of a linear combination of the shifted versions of $I_1(N)$, it might be necessary to expand $I_1(N)$ higher than u_1. E.g., if $\varepsilon^l I_1(N+i)$ with $l < 0$ is one of the components. Analysing these combinations yields the required order v_1. Now we are ready to apply Theorem 2 to calculate the values $I_{1,i}(N)$ with $o \leq i \leq v_1$ by means of a linear recurrence. In this process we determine that the first m_1 initial values are needed[3]. If one fails to get the representation of $I_1(N)$ in terms of nested hypergeometric sum expressions, the theorem is proven. Otherwise, by the properly chosen v_1, the $w_1, \ldots, w_n$ and

---

[3]In all our examples m_1 agreed with the recurrence order plus some extra points induced by the physical problem.

m_1, this yields also a nested hypergeometric sum representation of the $I_2(N),\ldots,I_n(N)$.

In the degenerated case, the decoupling algorithm provides several scalar linear recurrences, say in the $I_1(N),\ldots,I_l(N)$, and the remaining $I_{l+1}(N),\ldots,I_n(N)$ are expressed by a linear combination of the shifted versions of the $I_1(N),\ldots,I_l(N)$ and the $r_i(N)$. Applying Theorem 2 with the corresponding w_i,v_i,m_i (as described for the generic case) l times leads to the desired result.

Example. Consider the sequences $I_1(N),I_2(N),I_3(N)$ which are solutions of the coupled system (2.7) with $d=1$ and $n=3$ where

$$A_0 = \begin{pmatrix} N+1 & 0 & 0 \\ \varepsilon(3\varepsilon+2) & -2(3\varepsilon+1) & -2(-1+\varepsilon-2N) \\ -\varepsilon(3\varepsilon+2) & 2(3+3\varepsilon+2N) & 2(\varepsilon+1) \end{pmatrix}, \qquad A_1 = \begin{pmatrix} -2-\varepsilon-N & 2 & 0 \\ -2\varepsilon(3\varepsilon+2) & 2(5\varepsilon+2) & 4(-1+\varepsilon-N) \\ 0 & -2(4+\varepsilon+2N) & 0 \end{pmatrix}$$

and

$$r_1(N) = \left(-\frac{4(N+3)}{3(N+2)}\varepsilon^{-3} + \left(\frac{2}{3}\frac{6N^3+29N^2+45N+21}{(N+1)(N+2)^2} - \frac{2(2N+3)S_1(N)}{3(N+2)}\right)\varepsilon^{-2} + O(\varepsilon^{-1}),\right.$$

$$r_2(N) = -\frac{8}{3}\varepsilon^{-3} + \left(\frac{4(3N+1)}{3(N+1)} - \frac{8S_1(N)}{3}\right)\varepsilon^{-2} + O(\varepsilon^{-1}), \tag{2.9}$$

$$r_3(N) = \frac{8}{3}\varepsilon^{-3} + \left(\frac{-4(3N+1)}{3(N+1)} + \frac{8S_1(N)}{3}\right)\varepsilon^{-2} + O(\varepsilon^{-1})).$$

In particular, we are given the initial values (2.6) for $I_1(N) := I(N)$ We want to derive the ε-expansions for the $I_i(N)$ up to the order $u_i = -2$ for $1 \le i \le 3$. Uncoupling the system (first with generic right hand sides) shows that $w_1 = w_2 = w_3 = -2$, i.e., the ε-expansions in (2.9) are sufficiently high expanded. In particular, we obtain the linear recurrence In[6] with $I(N) = I_1(N)$ and can express $I_2(N)$ and $I_3(N)$ by the shifted versions of $I_1(N)$ and $r_1(N),r_2(N),r_3(N)$. Summarizing, the coefficients $I_{1,-3}(N)$ and $I_{1,-2}$ are computed in Out[7]. This yields the needed information to calculate the ε-expansions of $I_2(N)$ and $I_3(N)$ up to ε^{-2}.

The full machinery can be summarized after loading in the packages

In[8]:= << **OreSys.m**

> OreSys by Stefan Gerhold (optimized by C. Schneider) © RISC-Linz

In[9]:= << **SolveCoupledSystem.m**

> SolveCoupledSystem by Carsten Schneider © RISC-Linz

First, we execute the following command from the package `SolveCoupledSystem`:

In[10]:= **AnalyzeCoupledRecSystem**$[\{(A_0.\{I_1[N],I_2[N],I_3[N]\} + A_1.\{I_1[N],I_2[N],I_3[N]\}\},\{I_1[N],I_2[N],I_3[N]\},\varepsilon,-2,-2,-2]$

Out[10]= $\{\{\{I_1[N],3,-2\}\},\{-2,-2,-2\},\{\}\}$

This means that one can solve the system by providing the three consecutive initial values of $I_1(N)$ up to order $v_1 = -2$ (the starting point depends usually on the physical problem) and that one needs the ε-expansions of $r_1(N),r_2(N),r_3(N)$ up to the orders $m_1 = m_2 = m_3 = -2$. Providing the required information, we execute

In[11]:= **coupledSys** $= A_0.\{I_1[N],I_2[N],I_3[N]\} + A_1.\{I_1[N],I_2[N],I_3[N]\} - \{r_1[n],r_2[n],r_3[n]\};$

In[12]:= **SolveCoupledRecSystem**[**coupledSys**,

$\{I_1[N],I_2[N],I_3[N]\},\varepsilon,-3,\{-2,-2,-2\},\{I_1[N],1,\{\frac{5}{\varepsilon^3} - \frac{163}{12\varepsilon^2},\frac{130}{27\varepsilon^3} - \frac{695}{54\varepsilon^2},\frac{169}{36\varepsilon^3} - \frac{395}{32\varepsilon^2}\}\}]$

Out[12]= $\left\{\frac{1}{\varepsilon^3}\left(\frac{4(3N^2+6N+4)}{3(N+1)^2} + \frac{4S_1}{3(N+1)}\right) + \frac{1}{\varepsilon^2}\left(-\frac{2(20N^3+58N^2+57N+22)}{3(N+1)^3} + \frac{2(N+2)(2N-1)S_1}{3(N+1)^2} - \frac{S_1^2}{N+1} - \frac{S_2}{N+1}\right),\right.$

$\left.\frac{4}{3\varepsilon^3} - \frac{2}{\varepsilon^2},\frac{8}{3\varepsilon^3} + \frac{1}{\varepsilon^2}\left(-\frac{4(4N^2+7N+2)}{3(N+1)^2} + \frac{4(N+2)S_1}{3(N+1)}\right)\right\}$

and obtain the ε-expansions of $I_1(N),I_2(N)$ and $I_3(N)$ up to the orders $-2,-2$ and -2, respectively.

2.4 Finding power series solutions of coupled systems of linear differential equations

Finally, we are ready to present our differential equation solver for coupled systems.

Theorem 4. Suppose that the power series $\hat{I}_1, \dots, \hat{I}_n \in \mathbb{K}((\varepsilon))[[x]]$ with

$$\hat{I}_i(x) = \sum_{N=0}^{\infty} \left(I_{i,o}(N)\,\varepsilon^o + I_{i,o+1}(N)\,\varepsilon^{o+1} + I_{i,o+2}(N)\,\varepsilon^{o+2} + I_{i,o+3}(N)\,\varepsilon^{o+3} + \dots \right) x^N \qquad (2.10)$$

for some common $o \in \mathbb{Z}$ are solutions of the coupled system of differential equations[4]

$$A_0 \begin{pmatrix} \hat{I}_1(x) \\ \vdots \\ \hat{I}_n(x) \end{pmatrix} + A_1 D_x \begin{pmatrix} \hat{I}_1(x) \\ \vdots \\ \hat{I}_n(x) \end{pmatrix} \cdots + A_\delta D_x^\delta \begin{pmatrix} \hat{I}_1(x) \\ \vdots \\ \hat{I}_n(x) \end{pmatrix} = \begin{pmatrix} \hat{r}_1(x) \\ \vdots \\ \hat{r}_n(x) \end{pmatrix} \qquad (2.11)$$

for explicitly given $n \times n$ matrices $A_0, \dots, A_\delta$ with entries from $\mathbb{K}(x)$ and for some $\hat{r}_1(x), \dots, \hat{r}_u(x) \in \mathbb{K}((\varepsilon))[[x]]$ with

$$\hat{r}_i(x) = \sum_{N=0}^{\infty} \left(r_{i,o}(N)\,\varepsilon^o + r_{i,o+1}(N)\,\varepsilon^{o+1} + r_{i,o+2}(N)\,\varepsilon^{o+2} + r_{i,o+3}(N)\,\varepsilon^{o+3} + \dots \right) x^N. \qquad (2.12)$$

Then for any $u_1, \dots, u_n \in \mathbb{Z}$ one can determine $m_1, \dots, m_n \in \mathbb{N}$ and $w_1, \dots, w_n \in \mathbb{Z}$ with the property as stated in Theorem 3.

Proof. This result follows straightforwardly by holonomic closure properties. Namely, take the n partial linear differential equations in the $\hat{I}(x)$ and clear denominators by multiplying them with an appropriate polynomial from $\mathbb{K}[\varepsilon, x]$. Take one of the terms of the n equations, say $a(\varepsilon, x) D^k \hat{I}_i(x)$ with $0 \le k \le \delta$, $1 \le i \le n$ and $a(\varepsilon, x) \in \mathbb{K}[\varepsilon, x]$. With the Ansatz (2.10) we get

$$a(\varepsilon,x) D^k \hat{I}_i(x) = a(\varepsilon,x) D^k \sum_{N=0}^{\infty} \left(I_{i,o}(N)\,\varepsilon^o + I_{i,o+1}(N)\,\varepsilon^{o+1} + I_{i,o+2}(N)\,\varepsilon^{o+2} + \dots \right) x^N$$

$$= a(\varepsilon,x) \sum_{N=0}^{\infty} \left(I_{i,o}(N)\,\varepsilon^o + I_{i,o+1}(N)\,\varepsilon^{o+1} + I_{i,o+2}(N)\,\varepsilon^{o+2} + \dots \right) \prod_{j=0}^{k-1}(N-j) x^{N-k}.$$

Now we plug all these terms and (2.12) into the n equations. Doing coefficient comparison w.r.t. x^N leads to a coupled system of linear difference equations in the $I_1(N), \dots, I_n(N)$ and the $r_1(N), \dots, r_n(N)$. Performing an appropriate shift in N yields a coupled system of the form (2.7) with the right hand sides $r'_i(N)$ which depend linearly on the $r_i(N)$ and their shifted versions from (2.12). Note that the coefficients in $r'_i(N)$ can be expressed in terms of nested hypergeometric sum expressions since the coefficients in $r_i(N)$ can be expressed in terms of nested hypergeometric sum expressions. Thus we can apply Theorem 3 and obtain the claimed result.

Example. Consider the integrals $\hat{I}_1(x), \hat{I}_2(x), \hat{I}_3(x)$ given by $\hat{I}_1(x) = J_6(1,1;x)$, $\hat{I}_2(x) = J_6(2,1;x)$ and $\hat{I}_3(x) = J_6(1,2;x)$ where

$$J_6(\nu_2, \nu_4; x) = \int \frac{d^D k_1}{(2\pi)^D} \frac{d^D k_2}{(2\pi)^D} \frac{d^D k_3}{(2\pi)^D} \frac{1}{P_2^{\nu_2} P_4^{\nu_4} P_5 P_7 P_8 P_{10}} \qquad (2.13)$$

---

[4]Here D_x stands for the derivative operator.

with the propagators $P_2 = (k_1 - p)^2 - m^2$, $P_4 = (k_2 - p)^2 - m^2$, $P_5 = k_3^2 - m^2$, $P_7 = (k_3 - k_2)^2$, $P_8 = (k_1 - k_2)^2$ and $P_{10} = 1 - x\Delta.k_1$. Note that $\hat{I}_i(x)$ can be given in the power series representation (2.10) and the main task is to determine the coefficients $I_{i,j}(N)$. The IBP algorithm of `Reduze 2` [15] delivers e.g. the coupled system (2.11) with $n = 3$ and $\delta = 1$ where A_1 is the 3×3 identity matrix and

$$A_0 = - \begin{pmatrix} \dfrac{1+\varepsilon-x}{(x-1)x} & \dfrac{-2}{(x-1)x} & 0 \\[2mm] \dfrac{-\varepsilon(3\varepsilon+2)(x-2)}{4(x-1)x} & \dfrac{-2-5\varepsilon+x+3\varepsilon x}{2(x-1)x} & \dfrac{(-2\varepsilon-x+\varepsilon x)}{2(x-1)x} \\[2mm] \dfrac{\varepsilon(3\varepsilon+2)}{4(x-1)} & \dfrac{2+\varepsilon-3x-3\varepsilon x}{2(x-1)x} & -\dfrac{\varepsilon+1}{2(x-1)} \end{pmatrix} .$$

Furthermore, the $r_i(x)$ are given by a linear combination of master integrals where each one has a power series representation of the form (2.10). In particular, one can use symbolic summation tools [10, 11] to calculate the coefficients of the coefficients (up to a certain modest order in ε) in terms of nested hypergeometric sum expressions. For other situations multiple integration methods [12] are the appropriate tool. Thus the $\hat{r}_i(x)$ have a power series representation of the form (2.12) where the $r_{i,j}(N)$ can be given (up to a certain modest ε-order) explicitly in terms of nested hypergeometric sum expressions.

Now we activate our machinery. By holonomic closure properties we obtain precisely the coupled difference system from Example 2.3. Taking the data from Out[10] we calculate the required $r_{i,j}(N)$ by means of symbolic summation. Furthermore, we calculate the initial values (2.6) by exploiting the α-parametrization of the integrals; for further details on this method we refer to [17]. Note that in other situations we also used our summation tools, provided a reasonable sum representation has been derived. Finally, we activate the function call In[12] to get the final result.

Remark. The integrals $\hat{I}_1(x) = J_6(1,1;x)$, $\hat{I}_2(x) = J_6(2,1;x)$ and $\hat{I}_3(x) = J_6(1,2;x)$ with (2.13) themselves are master integrals produced by `Reduze 2` [15] in order to calculate the diagram[5]

                                                                                    (2.14)

in [19]. There we calculated the ε-expansion up to order 3 (and not just to order -2) in terms of 40 harmonic sums up to weight 7. The total calculation time was 229 seconds. The most complicated coupled system for diagram (2.14) had dimension $n = 4$. Interestingly enough, the right hand sides of (2.7) can be given in terms of generalized harmonic sums only, but the solution is given in terms of nested binomial sums. This strongly indicates that straightforward tactics, like transforming the system to a particular shape and reading off the solutions, are not sufficient for such systems.

Note that the obtained results in N-space presented in [19] and also in [18, 17] can be transformed to x-space by using iterated integral representations over general alphabets, generalizing in parts Poincaré iterated integrals [8].

3. Conclusion

We worked out a solver for coupled systems of linear differential equations where in one stroke

---

[5]The graph has been produced by `Axodraw` [33].

1. the coefficients of the formal power series solution are expanded in its ε-expansion;

2. the coefficients of the ε-expansion are given up to the desired order in terms nested hypergeometric expressions.

More precisely, we presented a complete algorithm which either provides such a solution or proves that such a representation is not possible. These algorithms have been implemented in the new Mathematica package `SolveCoupledSystem` which is based on the packages `Sigma` and `HarmonicSums`.

In concrete calculations IBP-techniques provide a recursively defined system of coupled equations. In [19] a new method has been worked out in order to treat these recursive systems fully automatically by using the algorithms presented in this article.

Acknowledgement. We would like to thank A. Behring and A. von Manteuffel for discussions.

References

[1] J. Blümlein, S. Klein, C. Schneider and F. Stan, J. Symbolic Comput. 47 (2012) 1267 [arXiv:1011.2656 [cs.SC]].

[2] S. Weinzierl, in: Computer Algebra in Quantum Field Theory: Integration, Summation and Special Functions, Texts & Monographs in Symbolic Computation eds. C. Schneider and J. Blümlein (Springer, Wien, 2013) 381 [arXiv:13016918 [hep-ph]].

[3] J. Blümlein and S. Kurth, Phys. Rev. D **60** (1999) 014018 [arXiv:hep-ph/9810241].
J.A.M. Vermaseren, Int. J. Mod. Phys. A **14** (1999) 2037 [arXiv:hep-ph/9806280].

[4] S. Moch, P. Uwer and S. Weinzierl, J. Math. Phys. **43** (2002) 3363 [hep-ph/0110083].

[5] J. Ablinger, J. Blümlein and C. Schneider, J. Math. Phys. **54** (2013) 082301 [arXiv:1302.0378 [math-ph]].

[6] J. Ablinger, J. Blümlein and C. Schneider, J. Math. Phys. **52** (2011) 102301 [arXiv:1105.6063 [math-ph]].

[7] J. Fleischer, A. V. Kotikov and O. L. Veretin, Nucl. Phys. B **547** (1999) 343 [hep-ph/9808242];
A. I. Davydychev and M. Y. Kalmykov, Nucl. Phys. B **699** (2004) 3 [arXiv:0303162 [hep-th]];
S. Weinzierl, J. Math. Phys. **45** (2004) 2656 [arXiv:0402131 [hep-ph]].

[8] J. Ablinger, J. Blümlein, C. G. Raab and C. Schneider, J. Math. Phys. **55** (2014) 112301 [arXiv:1407.1822 [hep-th]].

[9] J. Ablinger, J. Blümlein and C. Schneider, J. Phys. Conf. Ser. **523** (2014) 012060 [arXiv:1310.5645 [math-ph]];
J. Ablinger and J. Blümlein, in: Computer Algebra in Quantum Field Theory: Integration, Summation and Special Functions, C. Schneider, J. Blümlein, Eds., p. 1, (Springer, Wien, 2013) arXiv:1304.7071 [math-ph],

[10] C. Schneider, Sém. Lothar. Combin. **56** (2007) 1, article B56b;
C. Schneider, J. Phys. Conf. Ser. **523** (2014) 012037 [arXiv:1310.0160 [cs.SC]].

[11] C. Schneider, Simplifying Multiple Sums in Difference Fields, in: Computer Algebra in Quantum Field Theory: Integration, Summation and Special Functions Texts and Monographs in Symbolic Computation eds.
C. Schneider and J. Blümlein (Springer, Wien, 2013) 325 [arXiv:1304.4134 [cs.SC]];

[12] M. Apagodu and D. Zeilberger, Adv. Appl. Math. (Special Regev Issue), **37** (2006) 139;
J. Ablinger, Computer Algebra Algorithms for Special Functions in Particle Physics, Ph.D. Thesis, J. Kepler University Linz, 2012, arXiv:1305.0687 [math-ph].

[13] J. Ablinger, A Computer Algebra Toolbox for Harmonic Sums Related to Particle Physics, Diploma Thesis, J. Kepler University Linz, 2009, arXiv:1011.1176 [math-ph];
J. Ablinger, PoS LL **2014** (2014) 019.

[14] K. G. Chetyrkin, A. L. Kataev and F. V. Tkachov, Nucl. Phys. B **174** (1980) 345.

[15] A. von Manteuffel and C. Studerus, arXiv:1201.4330 [hep-ph];
C. Studerus, Comput. Phys. Commun. **181** (2010) 1293 [arXiv:0912.2546 [physics.comp-ph]].

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

[17] J. Ablinger, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, M. Round and C. Schneider, Nucl. Phys. B **885** (2014) 280 [arXiv:1405.4259 [hep-ph]].

[18] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel and C. Schneider, Nucl. Phys. B **890** (2014) 48 [arXiv:1409.1135 [hep-ph]];
J. Ablinger, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, M. Round, C. Schneider and F. Wißbrock, Nucl. Phys. B **882** (2014) 263 [arXiv:1402.0359 [hep-ph]];
J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, M. Round, C. Schneider and F. Wißbrock , Nucl. Phys. B **886** (2014) 733 [arXiv:1406.4654 [hep-ph]];
A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel and C. Schneider, Nucl. Phys. B **897** (2015) 612 [arXiv:1504.08217 [hep-ph]];
A. Behring, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel and C. Schneider, Phys. Rev. D **92** (2015) 11, 114005 [arXiv:1508.01449 [hep-ph]].

[19] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, C. Schneider "Calculating Three Loop Ladder and V-Topologies for Massive Operator Matrix Elements by Computer Algebra". Comput. Phys. Comm., in press. [arXiv:1509.08324 [hep-ph]].

[20] A. V. Kotikov, Phys. Lett. B **254** (1991) 158;
M. Caffo, H. Czyz, S. Laporta and E. Remiddi, Acta Phys. Polon. B **29** (1998) 2627 [hep-th/9807119]; Nuovo Cim. A **111** (1998) 365 [hep-th/9805118];
T. Gehrmann and E. Remiddi, Nucl. Phys. B **580** (2000) 485 [hep-ph/9912329].

[21] J.M. Henn, Phys. Rev. Lett. **110** (2013) 251601 [arXiv:1304.1806 [hep-th]].

[22] A. Danilevskiĭ, Mat. Sbornik **2** (1937) 169;
M.A. Barkatou, Appl. Algebra Engrg. Comm. Comput. **4**(3) (1993) 185;
M. Bronstein and M. Petkovšek, Theor. Comput. Sci., **157** (1) (1996) 3;
S.A. Abramov and E.V. Zima, Proc. Int. Conf. on Computational Modelling and Computing in Physics, Dubna, RU, Sept. 16-26 (1996) 16;
A. Bostan, F. Chyzak, E. de Panafieu, ISSAC'13 June 26-29, 2013, Boston, arXiv:1301.5414 [cs.SC] and references therein.

[23] B. Zürcher, Rationale Normalformen von pseudo-linearen Abbildungen, Master's thesis, Mathematik, ETH Zürich (1994).

[24] S. Gerhold, Uncoupling systems of linear Ore operator equations, Master's thesis, RISC, J. Kepler University, Linz, 2002.

[25] M. Petkovšek, J. Symbolic Comput. **14** (1992) 243;
S.A. Abramov and M. Petkovšek, in: J. von zur Gathen (ed.) Proc. ISSAC'94, 169 ACM Press (1994) pp. 169.
P. A. Hendriks and M. F. Singer, J. Symbolic Comput., **27**(3) (1999) 239.

[26] M. Bronstein J. Symbolic Comput., **29**(6) (2000) 841. C. Schneider, Symbolic Summation in Difference Fields Ph.D. Thesis RISC, Johannes Kepler University, Linz technical report 01-17 (2001).
C. Schneider, An. Univ. Timisoara Ser. Mat.-Inform. **42** (2004) 163;
C. Schneider, J. Differ. Equations Appl. **11** (2005) 799;
C. Schneider Appl. Algebra Engrg. Comm. Comput. **16**(2005) 1.

[27] M. Karr, J. ACM **28** (1981) 305;
C. Schneider, J. Symbolic Comput. **43** (2008) 611,[arXiv:0808.2543v1];

C. Schneider, in: Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2014, 15th International Symposium, F. Winkler, V. Negru, T. Ida, T. Jebelean, D. Petcu, S. Watt, D. Zaharie (ed.), (2015) pp. 26; IEEE Computer Society, arXiv:1412.2782v1 [cs.SC];
C. Schneider, in: Computer Algebra and Polynomials, Applications of Algebra and Number Theory, J. Gutierrez, J. Schicho, M. Weimann (ed.), Lecture Notes in Computer Science (LNCS) 8942 (2015), 157[arXiv:13077887 [cs.SC]];
C. Schneider, Ann. Comb. **14** (2010) 533[arXiv:0808.2596].

[28] C. Schneider, in: Motives, Quantum Field Theory, and Pseudodifferential Operators, Clay Mathematics Proceedings Vol. **12** ed. A. Carey, D. Ellwood, S. Paycha and S. Rosenberg,(Amer. Math. Soc) (2010), 285 [arXiv:0904.2323].

[29] C. Schneider J. Symb. Comput. **72** (2016) 82,doi:10.1016/j.jsc.2015.02.002 [arXiv:1408.2776 [cs.SC]].

[30] C. Schneider, A. De Freitas and J. Blümlein, PoS LL **2014** (2014) 017 [arXiv:1407.2537 [cs.SC]].

[31] J. Blümlein, Comput. Phys. Commun. **159** (2004) 19 [hep-ph/0311046].

[32] J. Ablinger, C. Schneider. Algebraic independence of (cyclotomic) harmonic sums, 2015. arXiv:1510.03692 [cs.SC].

[33] J.A.M. Vermaseren, Comput. Phys. Commun. **83** (1994) 45.