RISC Report Series 24-01 arXiv:2401.16314 [cs.SC]

CREATIVE TELESCOPING FOR HYPERGEOMETRIC DOUBLE SUMS

PETER PAULE AND CARSTEN SCHNEIDER

Dedicated to the memory of our friend Marko Petkovšek

ABSTRACT. We present efficient methods for calculating linear recurrences of hypergeometric double sums and, more generally, of multiple sums. In particular, we supplement this approach with the algorithmic theory of contiguous relations, which guarantees the applicability of our method for many input sums. In addition, we elaborate new techniques to optimize the underlying key task of our method to compute rational solutions of parameterized linear recurrences.

1. Introduction

We are interested in the following summation problem. **Given** a summand term $F(n, s_1, \dots, s_e)$ which is hypergeometric[1] in the s_i and in n. **Find** a recurrence

$$p_\gamma(n)S(n+\gamma) + \dots + p_0(n)S(n) = 0 \quad (n \geq 0) \tag{1}$$

which is satisfied by the hypergeometric multi-sum

$$S(n) := \sum_{s_1} \cdots \sum_{s_e} F(n, s_1, \dots, s_e).$$

In particular, we want to find a recurrence (1) which is P-finite, i.e., the coefficients $p_i(n)$ are polynomials in n. Moreover, we suppose that all summations are taken over finite summand supports. This means, all sums are understood to extend over all integers, positive and negative, but only finitely many terms contribute. For example, in $\sum_s \binom{n}{s}$, n a non-negative integer, the summand vanishes if $s < 0$ or $s > n$. With this restriction *homogeneous* sum recurrences are guaranteed.

In principle, one could apply the WZ method which is based on ideas of Sister Celine Fasenmyer and which is described in [PWZ96]. However, it turns out that all available implementations of this approach or of variations of it (e.g., Wegschaider's algorithm [Weg97]) meet in many applications serious problems of computational complexity. As a consequence we will follow a different approach which can be viewed as a simplified variant of Chyzak's algorithm [Chy00] within the holonomic system framework [Zei90]. A full account of computer algebra details and a comparison to [Chy00] is given in [Sch05]. For further enhancements of this holonomic summation approach in the setting of difference fields and rings [Kar81, Sch16] we refer to [Sch05, ABRS12, BRS18]. All these new features implemented within the summation package `Sigma` [Sch07] supported us to solve nontrivial problems coming, e.g., from combinatorics [APS05], number theory [SZ21] or elementary particle physics [BBF$^+$14].

In this article we will bring in new facets that explain the success of the presented summation method of double and multiple sums. On one side we will use insight from the summation theory of contiguous relations [Pau21] to show the existence of so-called hook-type recurrences which are the basic requirement of our summation approach. Further, we will illustrate in detail how these hook-type recurrences can be utilized to produce without any cost a scalar parameterized recurrence. As a consequence, the entire calculation effort is concentrated in finding a non-trivial rational solution of this derived parameterized recurrence. To gain substantial speed ups of our method we present new techniques to compute, e.g., optimal denominator predictions [Abr89b, Abr95, CPS08] and to discover parts of the the numerator contribution using the Gosper-Petkovšek

This work was supported by the Austrian Science Fund (FWF) grant P33530.

[1] $F(s)$ is hypergeometric in s iff $F(s+1)/F(s) = g(s)$ for some fixed rational function $g(s)$.

representation [Pet92, PWZ96, CPS08]. All these theoretic and algorithmic contributions will be illustrated by concrete multi-sum examples.

The outline of the article is as follows. We start with the base case of our method in Section 2: the calculation of (hook-type) recurrences of univariate hypergeometric sums. Further algorithmic and theoretic aspects concerning the existence of such recurrences are elaborated in Section 3. Based on this setup, we present our double sum method in Section 4 and supplement it with further examples in Section 5. Furthermore, we explain how this method can be extended to the multi-sum case in Section 6. In Section 7 we focus on the problem to speed up the key problem of our method. In particular, we focus on various significant improvements to solve parameterized recurrences efficiently. We conclude the article in Section 8.

2. Summation Methods for Single Sums

Here the basic task is as follows.
Given a positive integer γ and a summand term $f(n,r)$ which is hypergeometric in n and r, **compute** a P-finite recurrence (1) which is satisfied by the sum $S(n) := \sum_r f(n,r)$.

In the case that $f(n,r)$ satisfies some mild side conditions this problem can be solved by applying Zeilberger's algorithm [PWZ96]. More precisely, one can try to solve the *creative telescoping problem*: **Find** polynomials $p_i(n)$, free of r, and $g(n,r)$ such that

$$p_\gamma(n)f(n+\gamma,r)+p_{\gamma-1}(n)f(n+\gamma-1,r)+\cdots+p_0(n)f(n,r)=\Delta_r g(n,r); \tag{2}$$

Δ_r denotes the (forward) difference operator defined as usual by $\Delta_r g(r) = g(r+1) - g(r)$. One can show that if such a $g(n,r)$ exists, it must be a rational function multiple of $f(n,r)$. Finally, note that given a solution for (2), recurrence (1) is obtained from (2) by summation over all r.

Example 1. Within the computer algebra system Mathematica one may use the Paule-Schorn implementation [PS95] to carry out this summation paradigm. For instance, one can compute for the univariate hypergeometric sum

$$f_1(n,s) := \sum_{k=0}^{s} \binom{n}{k}^2 \binom{n+s-k}{n}, \tag{3}$$

the recurrence

$$-((1+s)^2 f_1(n,r,s)) + (5+6s+2s^2+n+n^2) f_1(n,r,s+1) - (2+s)^2 f_1(n,r,s+2) = 0 \tag{4}$$

as follows. After loading the package

In[1]:= **<< RISC‘fastZeil‘**

Fast Zeilberger Package
written by Peter Paule, Markus Schorn, and Axel Riese
© RISC-JKU

into Mathematica and defining its summand $f(n,k,s)$ with

In[2]:= $\mathbf{summand = Binomial[n,k]^2 Binomial[n+s-k,n];}$

one can solve the creative telescoping problem (here k and s takes over the role of r and n in (2)) with the following command:

In[3]:= $\mathbf{Zb[summand, \{k, 0, s\}, s]}$

If ‘s’ is a natural number and ‘n’ is no negative integer, then:

Out[3]= $\{-(1+s)^2\mathtt{SUM}[s] + (5+n+n^2+6s+2s^2)\mathtt{SUM}[1+s] - (2+s)^2\mathtt{SUM}[2+s] == 0\}$

Alternatively, one may use the `Sigma` package [Sch07]

In[4]:= `<<Sigma.m`

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

by inserting the input sum

In[5]:= $\mathbf{f1 = SigmaSum[SigmaBinomial[n,k]^2 SigmaBinomial[n+s-k,n], k, 0, s]}$

Out[5]= $\sum_{k=0}^{s} \binom{n}{k}^2 \binom{-k+n+s}{n}$

and executing the following function call:

In[6]:= **GenerateRecurrence[f1, s]**

Out[6]= $\{(1+s)^2\text{SUM}[s] + (-5-n-n^2-6s-2s^2)\text{SUM}[(+s] + (2+s)^2\text{SUM}[2+s] == 0\}$

□

It can be that for a fixed order γ there exists only the trivial solution, i.e., where all the $p_i(n)$ in (2) are 0. In this case one has to increase the order γ incrementally until a non-trivial solution is computed. Its existence is guaranteed by the theory explained in [PWZ96]; see also Section 3 below.

2.1. **A slight but important variation.** Many identities involve summands in more than one independent variable. For instance, instead of the summand $f(n,r)$ consider the summand $f(m,n,r)$, now hypergeometric in m, n and r. For the following it is important to note that completely analogous to (2) one can compute hook-type recurrences like

$$p_\gamma(m,n)f(m+1,n,r) + p_{\gamma-1}(m,n)f(m,n+\gamma-1,r) + \cdots + p_0(m,n)f(m,n,r) = \Delta_r g(m,n,r) \quad (5)$$

if they exist. This task can be accomplished by a variation of [PWZ96]. Moreover, the question whether relations like (5) do exist, will be considered in Section 3. Summing (5) over all r (again assuming finite summand support) yields

$$p_\gamma(m,n)S(m+1,n) + p_{\gamma-1}(m,n)S(m,n+\gamma-1) + \cdots + p_0(m,n)S(m,n) = 0 \quad (m \geq 0) \quad (6)$$

with $S(m,n) = \sum_r f(m,n,r)$ and where the $p_i(m,n)$ are polynomials in m and n.

Example 2. The calculation of such hook-type recurrences can be accomplished, e.g., with the Paule-Schorn implementation; see [Pau21]. For instance, given the summand $f(n,s,k)$ of (3) defined in In[2] (here n, s and k take over the role of m, n and r in (6)) one can compute the rational functions $\rho_i(n,s,k) \in \mathbb{Q}(n,s,k)$ with $f(n,s,k) = \rho_0(n,s,k)f_1(n,s,k)$,

$$f(n,s+1,k) = \rho_1(n,s,k)f_1(n,s,k) \quad \text{and} \quad f(n+1,s,k) = \rho_2(n,s,k)f_1(n,s,k)$$

by executing

In[7]:= **$\{\rho_0, \rho_1, \rho_2\}$ = FunctionExpand[{summand, (summand/.s → s + 1), (summand/.n → n + 1)}/summand]**

Out[7]= $\{1, \frac{1-k+n+s}{1-k+s}, \frac{(1+n)(1-k+n+s))}{(1-k+n)^2}\}$

Then we can extract the hook-type recurrence with the Paule-Schorn implementation by executing the function call

In[8]:= **Gosper[summand, {k, 0, s}, Parameterized → $\{\rho_0, \rho_1, \rho_2\}$]**

Out[8]= $\{\text{Sum}[(1+n^2+2s-2ns+2s^2)F_0[k] - 2(1+s)^2F_1[k] + (1+n)^2F_2[k], \{k,0,s\}] == \frac{2(n-s)^2(1+n-s)^2\text{Binomial}[1+n,s]^2}{(1+n)^2}\}$

More precisely, the output yields

$$\sum_{k=0}^{s} \left[(1+n^2+2s-2ns+2s^2)F_0[k] - 2(1+s)^2F_1[k] + (1+n)^2F_2[k]\right] = \frac{2(n-s)^2(1+n-s)^2}{(1+n)^2}\binom{n+1}{s}^2$$

with $F_0[k] = \rho_0(n,s,k)f_1(n,s,k) = f_1(n,s,k)$, $F_1[k] = \rho_1(n,s,k)f_1(n,s,k) = f_1(n,s+1,k)$ and $F_2[k] = \rho_2(n,s,k)f_1(n,s,k) = f_1(n+1,s,k)$. Then splitting the sum into parts and taking care of the summation ranges produces

$$(1+2s+2s^2-2sn+n^2)f_1(n,s) - 2(1+s)^2 f_1(n,s+1) + (1+n)^2 f_1(n+1,s) = 0; \quad (7)$$

this example playing an important role in Example 11 below will be explored further in the next Section 3.

Alternatively, one may use the summation package `Sigma` by taking the input sum In[5] and executing the command

In[9]:= **GenerateRecurrence[f1, OneShiftIn → n]**

Out[9]= $\{(1+n^2+2s-2ns+2s^2)\mathrm{SUM}[s]-2(1+s)^2\mathrm{SUM}[1+s]+(1+n)^2\mathrm{SUM}[1+n,s]==0\}$

□

3. Existence of Recurrences

To discuss, in particular, to guarantee the existence of "hook-type" recurrences of the form as in (5) we make use of the approach described in [Pau21]. This approach is based on a *parameterized* version of Gosper's algorithm containing Zeilberger's creative telescoping as a special instance. In [Pau21] this idea is used to derive contiguous relations from telescoping contiguous relations, thus covering the existence of both the Zeilberger-type recurrences as in (2) and the hook-type recurrences as in (5).

As a concrete illustrating example we choose the hook-type recurrence (7) for the sum (3) which will play an important role in Example 11. As in [Pau21] we use the notation

$$ {}_pF_q\left(\begin{matrix} a_1,\dots,a_p \\ b_1,\dots,b_q \end{matrix};z\right)_k := \frac{(a_1)_k\cdots(a_p)_k}{(b_1)_k\cdots(b_q)_k}\frac{z^k}{k!}, \tag{8} $$

where $(x)_k$ is the shifted factorial

$$ (x)_k = x(x+1)\cdots(x+k-1) \text{ if } k\geq 1 \text{ and } (x)_0=1. $$

Remark 1. The motivation for the notation (8) and for considering recurrences for such summands where integer shifts in more than one parameter are allowed goes back to Gauß who was the first to compile a table of fifteen classical contiguous relations; e.g., [Gau13, 7.2],

$$ (b-a)\,{}_2F_1\left(\begin{matrix} a,b \\ c \end{matrix};z\right) + a\,{}_2F_1\left(\begin{matrix} a+1,b \\ c \end{matrix};z\right) - b\,{}_2F_1\left(\begin{matrix} a,b+1 \\ c \end{matrix};z\right) = 0, \tag{9} $$

where

$$ {}_2F_1\left(\begin{matrix} a,b \\ c \end{matrix};z\right) = \sum_{k=0}^{\infty}\frac{(a)_k(b)_k}{(c)_k k!}z^k, $$

and where the variables a,b,c, and z range over $\mathbb{C}$ with $|z|<1$ as a condition for convergence.

In [Pau21, Def. 3] the existence and derivation of contiguous relations such as (9) is algorithmically explained as limiting cases of *telescoping contiguous relations*. For example, relation (9) is obtained by taking the limit $n\to\infty$ after summing both sides of

$$ \begin{aligned} &c_0\cdot{}_2F_1\left(\begin{matrix} a,b \\ c \end{matrix};z\right)_k + c_1\cdot{}_2F_1\left(\begin{matrix} a+1,b \\ c \end{matrix};z\right)_k + c_2\cdot{}_2F_1\left(\begin{matrix} a,b+1 \\ c \end{matrix};z\right)_k \\ &\quad = \Delta_k\,C(k)\,{}_2F_1\left(\begin{matrix} a,b \\ c \end{matrix};z\right)_k,\ k\geq 0, \end{aligned} \tag{10} $$

over k from 0 to n. Theorem 1 in [Pau21] predicts the existence of the c_j, $0\leq j\leq 2$, as rational functions in $\mathbb{C}(a,b,c,z)$, not all zero, and of a polynomial $C(x)\in\mathbb{C}[x]$ with $C(0)=0$ and $\deg C(x)\leq 1$. Moreover, as exemplified in [Pau21, Ex.1, Sec. 6], these constituents can be computed via parameterized creative telescoping:

$$ c_0=b-a, c_1=a, c_2=-b, \text{ and } C(x)=0. $$

In view of Zeilberger's creative telescoping paradigm, telescoping contiguous relations in which shifts in only one variable occur can be called of Zeilberger-type. An example is

$$ \begin{aligned} &c_0\cdot{}_2F_1\left(\begin{matrix} a,b \\ c \end{matrix};z\right)_k + c_1\cdot{}_2F_1\left(\begin{matrix} a+1,b \\ c \end{matrix};z\right)_k + c_2\cdot{}_2F_1\left(\begin{matrix} a+2,b \\ c \end{matrix};z\right)_k \\ &\quad = \Delta_k\,C(k)\,{}_2F_1\left(\begin{matrix} a,b \\ c \end{matrix};z\right)_k; \end{aligned} \tag{11} $$

here Theorem 1 in [Pau21] predicts $C(0) = 0$ and $\deg C(x) \leq 2$. Indeed, by parameterized telescoping one computes [Pau21, eqs. (78) and (79)],

$$(c_0, c_1, c_2) = (a(a-c+1), a((a-b+1)z - 2a - 2 + c), a(a+1)(1-z)) \text{ and } C(x) = x(x+c-1).$$

We remark that (11), after summation over k from 0 to n, in the limit $n \to \infty$ turns into

$$(a+1-c)\,{}_2F_1\left(\begin{matrix}a,b\\c\end{matrix};z\right) + ((a+1-b)z - 2(a+1)+c)\ {}_2F_1\left(\begin{matrix}a+1,b\\c\end{matrix};z\right)$$
$$+ (1-z)(a+1)\,{}_2F_1\left(\begin{matrix}a+2,b\\c\end{matrix};z\right) = 0, \tag{12}$$

which is the first entry (with a replaced by $a+1$) in the list of fifteen fundamental contiguous relations stated by Gauß [Gau13, 7.2]. In the context of the present article taking such limits is irrelevant. Nevertheless, we will exploit the theory for parameterized telescoping relations to guarantee the existence of hook-type recurrences and also for computing them algorithmically.

Back to our illustrating example, the hook-type relation (7) satisfied by $f_1(n,s)$ as in (3). It is easily verified that $f_1(n,s)$ can be rewritten as

$$f_1(n,s) = \binom{n+s}{n} F_1(n,s) \tag{13}$$

with

$$F_1(n,s) := \sum_{k=0}^{s} {}_3F_2\left(\begin{matrix}-n,-n,-s\\1,-n-s\end{matrix};1\right)_k = {}_3F_2\left(\begin{matrix}-n,-n,-s\\1,-n-s\end{matrix};1\right). \tag{14}$$

The latter equality follows from the fact that the hypergeometric ${}_3F_2$-series terminates at $k = s$ owing to the factor $(-s)_k$ in the kth summand and $0 \leq s \leq n$.

Using (14) the hook-type relation (7) rewrites into

$$(1+2s+2s^2-2sn+n^2)F_1(n,s) + (1+n)^2\frac{\binom{n+s+1}{n+1}}{\binom{n+s}{n}}F_1(n+1,s) - 2(1+s)^2\frac{\binom{n+s+1}{n}}{\binom{n+s}{n}}F_1(n,s+1)$$
$$= (1+2s+2s^2-2sn+n^2)F_1(n,s) + (1+n)(1+n+s)F_1(n+1,s)$$
$$- 2(1+s)(1+n+s)F_1(n,s+1) = 0. \tag{15}$$

The shift-structure of the hook-type recurrence (15) leads to conjecture the existence of a telescoping contiguous relation with left hand side

$$c_0 \cdot {}_3F_2\left(\begin{matrix}a,b,c\\d,e\end{matrix};1\right)_k + c_1 \cdot {}_3F_2\left(\begin{matrix}a-1,b-1,c\\d,e-1\end{matrix};1\right)_k + c_2 \cdot {}_3F_2\left(\begin{matrix}a,b,c-1\\d,e-1\end{matrix};1\right)_k. \tag{16}$$

Namely, setting $a = -n, b = -n, c = -s, d = 1$, and $e = -n-s$ the ${}_3F_2(\dots)_k$ expressions in (16) from left to right turn into the summands of $F_1(n,s)$, $F_1(n+1,s)$, and $F_1(n,s+1)$.

Indeed, the respective telescoping contiguous relation is predicted as a special instance of the following general theorem where $\mathbb{K}$ is a suitable field containing $\mathbb{Q}$.

Theorem 1 (Theorem 1A in [Pau21])**.** *Let $a_1, \dots, a_{q+1}$ and $b_1, \dots, b_q$ be complex parameters. For $0 \leq l \leq q$ let*

$$(\alpha_1^{(l)}, \dots, \alpha_{q+1}^{(l)}, \beta_1^{(l)}, \dots, \beta_q^{(l)})$$

be pairwise different tuples with non-negative integer entries. Then there exist $c_0, \dots, c_q$ in $\mathbb{K}$, not all 0, and a polynomial $C(x) \in \mathbb{K}[x]$ such that for all $k \geq 0$,

$$\sum_{l=0}^{q} c_l \cdot {}_{q+1}F_q\left(\begin{matrix}a_1+\alpha_1^{(l)}, \dots, a_{q+1}+\alpha_{q+1}^{(l)}\\b_1-\beta_1^{(l)}, \dots, b_q-\beta_q^{(l)}\end{matrix};1\right)_k = \Delta_k\, C(k)\,{}_{q+1}F_q\left(\begin{matrix}a_1, \dots, a_{q+1}\\b_1, \dots, b_q\end{matrix};1\right)_k. \tag{17}$$

Moreover, $C(0) = 0$, and if $C(x) \neq 0$, for the polynomial degree of $C(x)$ one has

$$\deg C(x) \leq 1 + M \quad \textit{where} \quad M := \max_{0 \leq l \leq q}\{\alpha_1^{(l)} + \dots + \alpha_{q+1}^{(l)} + \beta_1^{(l)} + \dots + \beta_q^{(l)}\}. \tag{18}$$

For our case we have

$$(a_1, a_2, a_3) = (a-1, b-1, c-1) \text{ and } (b_1, b_2) = (d, e),$$

and with

$$(\alpha_1^{(0)}, \alpha_2^{(0)}, \alpha_3^{(0)}, \beta_1^{(0)}, \beta_2^{(0)}) = (1,1,1,0,0),$$
$$(\alpha_1^{(1)}, \alpha_2^{(1)}, \alpha_3^{(1)}, \beta_1^{(1)}, \beta_2^{(1)}) = (0,0,1,0,1),$$
$$(\alpha_1^{(2)}, \alpha_2^{(2)}, \alpha_3^{(2)}, \beta_1^{(2)}, \beta_2^{(2)}) = (1,1,0,0,1)$$

the theorem gives

$$\begin{aligned} &c_0 \cdot {}_3F_2\left(\begin{matrix} a, b, c \\ d, e \end{matrix}; 1\right)_k + c_1 \cdot {}_3F_2\left(\begin{matrix} a-1, b-1, c \\ d, e-1 \end{matrix}; 1\right)_k + c_2 \cdot {}_3F_2\left(\begin{matrix} a, b, c-1 \\ d, e-1 \end{matrix}; 1\right)_k \\ &= \Delta_k\, C(k)\, {}_3F_2\left(\begin{matrix} a-1, b-1, c-1 \\ d, e \end{matrix}; 1\right)_k, \quad k \geq 0, \end{aligned} \tag{19}$$

where $C(x)$ is predicted to be a polynomial such that $C(0) = 0$ and $\deg C(x) \leq 1 + M = 4$.

The computation of c_0, c_1, c_2 and $C(x)$ can be done with the summation package `Sigma` or, alternatively, with the Paule-Schorn implementation [PS95] of Zeilberger's algorithm, both written in Mathematica. Concerning the latter, in [Pau21] the reader finds various detailed examples how to do this. For our concrete case (19), the program finds:

$$\begin{aligned} c_0 &= -(-1+a)(-1+b)(-1+c) \\ &\quad (a^2b + ab^2 + c - a^2c - abc - b^2c - d - abd + acd + bcd - abe - ce + ace + bce + de - cde), \\ c_1 &= -(-1+a)(-1+b)(-1+c)(a-d)(b-d)(-1+e), \\ c_2 &= (-1+a)(-1+b)(-1+c)(-1+a+b-d)(c-d)(-1+e), \end{aligned}$$

and

$$\begin{aligned} C(x) = -x(-1+d+x)(-1+e+x) \\ &(-2ab + a^2b + ab^2 - c + 2ac - a^2c + 2bc - abc - b^2c + d - abd - 2cd + acd + bcd \\ &+ abx + cx - acx - bcx - dx + cdx). \end{aligned}$$

Setting $a = -n, b = -n, c = -s, d = 1$, and $e = -n - s$ results in

$$\begin{aligned} (c_0, c_1, c_2) = \big(&-(1+n)^3(1+s)(1+n^2+2s-2ns+2s^2), -(1+n)^4(1+s)(1+n+s), \\ &2(1+n)^3(1+s)^2(1+n+s)\big), \end{aligned}$$

and summing (19) over k from 0 to $s+1$ produces (15) which is equivalent to (7). Note that with the function call In[8] this recurrence has been produced directly with the specialization $a = -n, b = -n, c = -s, d = 1$, and $e = -n - s$.

We conclude this section with a couple of remarks. First, the theorems from [Pau21] guarantee the existence of hook-type recurrences for hypergeometric summands of the form as in (8). In addition, the respective telescoping contiguous relations can be computed by any implementation of parameterized telescoping. Finally, we remark that recurrences of Zeilberger-type are covered as a special case. For example, the relation (4),

$$\begin{aligned} &-(1+s)^2 f_1(n,s) + (5+6s+2s^2+n+n^2) f_1(n,s+1) - (2+s)^2 f_1(n,s+2) \\ &= -(1+s)^2 F_1(n,s) + (5+6s+2s^2+n+n^2)\frac{\binom{n+s+1}{n}}{\binom{n+s}{n}} F_1(n,s+1) \\ &\quad - (2+s)^2 \frac{\binom{n+s+2}{n}}{\binom{n+s}{n}} F_1(n,s+2) \\ &= -(1+s)^2 F_1(n,s) + (5+6s+2s^2+n+n^2)\frac{n+s+1}{s+1} F_1(n,s+1) \\ &\quad - (2+s)\frac{(n+s+1)(n+s+2)}{s+1} F_1(n,s+2) = 0, \end{aligned} \tag{20}$$

is predicted by another special case of Theorem 1. Namely,

$$\begin{aligned} &c_0 \cdot {}_3F_2\left({a,b,c \atop d,e};1\right)_k + c_1 \cdot {}_3F_2\left({a,b,c-1 \atop d,e-1};1\right)_k + c_2 \cdot {}_3F_2\left({a,b,c-2 \atop d,e-2};1\right)_k \\ &\quad = \Delta_k\, C(k)\, {}_3F_2\left({a,b,c-2 \atop d,e};1\right)_k, \quad k \geq 0, \end{aligned} \tag{21}$$

for which Theorem 1 predicts for the polynomial $C(x)$ that $C(0) = 0$ and $\deg C(x) \leq 1 + M = 5$. With parameterized telescoping one finds:

$$\begin{aligned} c_0 &= (-2+c)(-1+c)(1+a-e)(1+b-e), \\ c_1 &= -(-2+c)(-1+e)(3+a+b+ab-3c-ac-bc+2d-2e+2ce-de), \\ c_2 &= (-2+c)(-1+c-d)(-2+e)(-1+e), \end{aligned}$$

and

$$C(x) = (c-e)x(-1+d+x)(-1+e+x).$$

Note that besides providing a bound on the order of Zeilberger-type and hook-type recurrences (and, in general, of recurrences stemming from contiguous relations with arbitray shift pattern) such kind of prediction also includes a bound on the degree of the polynomial $C(x)$ in the Δ_k part of the telescoping contiguous relation.

4. The Double Sum Method

Here the basic task is as follows.
Given a summand $F(n,r,s)$ which is hypergeometric in n, r and s, **compute** a P-finite recurrence (1) which is satisfied by the sum $S(n) := \sum_r \sum_s F(n,r,s)$.

Example 3. With our method under consideration we can solve the following problem. Given the double sum

$$S(n) = \sum_{r=0}^{n} \sum_{s=0}^{r} \binom{n}{r}\binom{n+r}{r}\binom{k}{s}^3, \tag{22}$$

find a recurrence of the form (1) with $\gamma = 2$. □

The overall goal of the method is to compute a recurrence of type (2) where $f(n,r)$ is defined to be the inner sum, i.e.,

$$f(n,r) := \sum_s F(n,r,s).$$

Note that $g(n,r)$ no longer needs to be a rational function multiple of $f(n,r)$, hence a suitable ansatz for $g(n,r)$ has to be introduced; see ANSATZ below. From (2) the desired recurrence (1) for $S(n)$ is obtained by summing over all r — as in Zeilberger's algorithm for single sums.

In order to find (2) we propose the following method:
First one computes recurrences of the following form,

$$f(n, r+\delta+1) = \lambda_0(n,r)f(n,r) + \cdots + \lambda_\delta(n,r)f(n,r+\delta), \tag{23}$$

and

$$f(n+1,r) = \mu_0(n,r)f(n,r) + \cdots + \mu_\delta(n,r)f(n,r+\delta), \tag{24}$$

where the $\lambda_i(n,r)$ and $\mu_i(n,r)$ are rational functions in n and r. This can be accomplished by following Section 2; the existence is discussed in Section 3.

Example 4 (Cont.)**.** For $f(n,r) = \sum_{s=0}^{r} \binom{n}{r}\binom{n+r}{r}\binom{k}{s}^3$ we can compute with [PWZ96] the recurrences

$$\begin{aligned} &8(-1+n-r)(n-r)(1+n+r)(2+n+r)f(n,r) \\ &\quad + (-1+n-r)(2+n+r)(16+21r+7r^2)f(n,r+1) - (2+r)^4 f(n,r+2) = 0 \end{aligned} \tag{25}$$

and

$$(1+n+r)f(n,r)+(-1-n+r)f(n+1,r)=0, \tag{26}$$

i.e., we are in the case $\delta = 1$. With `Sigma` (alternatively, one may use the Paule-Schorn implementation), this can be carried out as follows.

In[10]:= $\mathbf{innerSum} = \sum_{\mathbf{s=0}}^{\mathbf{r}} \binom{\mathbf{n}}{\mathbf{r}}\binom{\mathbf{n+r}}{\mathbf{r}}\binom{\mathbf{k}}{\mathbf{s}}^{\mathbf{3}};$

In[11]:= **recR = GenerateRecurrence[innerSum][[1]]/.SUM → f**

Out[11]= $8(-1+n-r)(n-r)(1+n+r)(2+n+r)f[r]+(-1+n-r)(2+n+r)(16+21r+7r^2)f[1+r]-(2+r)^4f[2+r]==0$

In[12]:= **recRN = GenerateRecurrence[innerSum, OneShiftIn → n][[1]]/.SUM → f**

Out[12]= $(1+n+r)f[r]+(-1-n+r)f[1+n,r]==0$

□

ANSATZ: For $g(n,r)$ one starts with an expression with undetermined coefficients of the following form,

$$g(n,r) = \phi_0(n,r)f(n,r)+\cdots+\phi_\delta(n,r)f(n,r+\delta). \tag{27}$$

Then the unknown polynomials $p_i(n)$, free of r, and the unknown rational function coefficients $\phi_i(n,r)$ for $g(n,r)$ are computed such that the certificate recurrence (2) holds. In view of (23) and (24), the key observation is that any shift in n and r of $f(n,r)$ and also $g(n,r)$ can be represented as a linear combination of $f(n,r),\dots,f(n,r+\delta)$ over rational functions in n and r. Then rewriting both sides of (2) in terms of these generators, allows us to compute the unknown data by comparing the coefficients of all the $f(n,r+i)$ involved.

More precisely, we proceed as follows. After computing the recurrences (23) and (24), in a **second step** we rewrite the right hand side of (2) as a linear combination in $f(n,r)$, $f(n,r+1)$ up to $f(n,r+\delta)$. Namely, due to (23) and (24) there exist rational functions $\psi_i^{(j)}(n,r)$ in n and r such that for all nonnegative integers i,

$$f(n+j,r) = \sum_{i=0}^{\delta}\psi_i^{(j)}(n,r)f(n,r+i). \tag{28}$$

Consequently,

$$\sum_{j=0}^{\gamma}p_j(n)f(n+j,r) = \sum_{i=0}^{\delta}f(n,r+i)\sum_{j=0}^{\gamma}p_j(n)\psi_i^{(j)}(n,r). \tag{29}$$

Example 5 (Cont.)**.** We make the ansatz

$$g(n,r)=\phi_0(n,r)f(n,r)+\phi_1(n,r)f(n,r+1) \tag{30}$$

with

$$p_0(n)f(n,r)+p_1(n)f(n+1,r)+p_2(n)f(n+2,r)=\Delta_r g(n,r). \tag{31}$$

Then using (26), i.e., using $f(n+1,r)=\frac{(n+1+r)}{(n+1-r)}f(n,r)$ we rewrite the left hand side of (31) to

$$\begin{aligned}p_0(n)f(n,r)&+p_1(n)f(n+1,r)+p_2(n)f(n+2,r)\\ &= f(n,r)\Big(p_0(n)+p_1(n)\frac{n+r+1}{n-r+1}+p_2(n)\frac{(n+r+1)(n+r+2)}{(n-r+1)(n-r+2)}\Big).\end{aligned}$$

□

OBSERVATION: To compare coefficients we represent also $\Delta_r g(n,r)$ as a linear combination in $f(n,r)$, $f(n,r+1)$ up to $f(n,r+\delta)$. We get

$$\begin{aligned}
\Delta_r g(n,r) &\overset{(27)}{=} \sum_{i=0}^{\delta}\phi_i(n,r+1)f(n,r+i+1) - \sum_{i=0}^{\delta}\phi_i(n,r)f(n,r+i)\\
&\overset{(23)}{=} \sum_{i=0}^{\delta-1}\phi_i(n,r+1)f(n,r+i+1) + \phi_\delta(n,r+1)\sum_{i=0}^{\delta}\lambda_i(n,r)f(n,r+i) - \sum_{i=0}^{\delta}\phi_i(n,r)f(n,r+i)\\
&= \sum_{i=1}^{\delta}\big(\phi_{i-1}(n,r+1)+\phi_\delta(n,r+1)\lambda_i(n,r)-\phi_i(n,r)\big)f(n,r+i)\\
&\qquad + \big(\phi_\delta(n,r+1)\lambda_0(n,r)-\phi_0(n,r)\big)f(n,r).
\end{aligned}$$

Comparing the coefficients of the $f(n,r+i)$ to those in (29) results in the coupled system

$$\phi_0(n,r) = \lambda_0(n,r)\phi_\delta(n,r+1) - \sum_{j=0}^{\gamma}p_j(n)\psi_0^{(j)}(n,r) \tag{32}$$

and

$$\phi_i(n,r) = \phi_{i-1}(n,r+1) + \lambda_i(n,r)\phi_\delta(n,r+1) - \sum_{j=0}^{\gamma}p_j(n)\psi_i^{(j)}(n,r) \tag{33}$$

for $1 \le i \le \delta$; see [Sch05, Lemma 1]. This system can be uncoupled by simple linear algebra, i.e., after triangularization we arrive at the equivalent system consisting of the equations (32), (33) for $1 \le i < \delta$ and

$$-\phi_\delta(n,r) + \sum_{j=0}^{\delta}\lambda_j(n,r+\delta-j)\phi_\delta(n,r+\delta+1-j) = \sum_{j=0}^{\gamma}p_j(n)\sum_{i=0}^{\delta}\psi_i^{(j)}(n,r+\delta-i); \tag{34}$$

see [Sch05, Lemma 2]. Summarizing, any solution $\phi_i(n,r)$ and $p_i(n)$ with (32), (33) for $1 \le i < \delta$ and (34) gives a solution $g(n,k)$ with (27) and $p_i(n)$ for (2).

Example 6 (Cont.)**.** After rewriting $\Delta_r g(n,r)$ as explained above we can express (31) in the form

$$\begin{aligned}
&f(n,r)\Big(\phi_1(n,r+1)\frac{8(-1+n-r)(n-r)(1+n+r)(2+n+r)}{(2+r)^4} - \phi_0(n,r)\Big)\\
&+ f(n,r+1)\Big(\phi_0(n,r+1) + \phi_1(n,r+1)\frac{(-1+n-r)(2+n+r)(16+21r+7r^2)}{(2+r)^4} - \phi_1(n,r)\Big)\\
&\qquad = f(n,r)\Big(p_0(n) + p_1(n)\frac{n+r+1}{n-r+1} + p_2(n)\frac{(n+r+1)(n+r+2)}{(n-r+1)(n-r+2)}\Big).
\end{aligned} \tag{35}$$

By coefficient comparison of the $f(n,r)$ and $f(n,r+1)$ we get the coupled system

$$\begin{aligned}
&\phi_1(n,r+1)\frac{8(-1+n-r)(n-r)(1+n+r)(2+n+r)}{(2+r)^4} - \phi_0(n,r)\\
&\qquad = p_0(n) + p_1(n)\frac{n+r+1}{n-r+1} + p_2(n)\frac{(n+r+1)(n+r+2)}{(n-r+1)(n-r+2)}
\end{aligned} \tag{36}$$

and

$$\phi_0(n,r+1) = -\phi_1(n,r+1)\frac{(-1+n-r)(2+n+r)(16+21r+7r^2)}{(2+r)^4} + \phi_1(n,r). \tag{37}$$

Finally, shifting (36) in r and replacing $\phi_0(n, r+1)$ with (37) gives

$$\frac{8(1-n+r)(2-n+r)(2+n+r)(3+n+r)}{(3+r)^4}\phi_1(n,r+2) - \left(\frac{(1-n+r)(2+n+r)(16+21r+7r^2)}{(2+r)^4}\right)\phi_1(n,r+1) - \phi_1(n,r) = p_0(n) + p_1(n)\frac{2+n+r}{n-r} + p_2(n)\frac{(2+n+r)(3+n+r)}{(n-r)(1+n-r)}. \quad (38)$$

This means that any solution $\phi_i(n, r)$ and $p_i(n)$ with (38) and (37) gives a solution for (31). Note that the previous transformation steps have been carried out only for illustrative purpose; the equations (38) and (37) can be obtained directly from the explicit formulas (32), (33), and (34). □

Hence, in our **third step** we go on as follows. By using the algorithms given in Section 7 we try to find a rational function $\phi_\delta(n, r)$ in n and r and polynomials $p_j(n)$ such that (34) holds. If we succeed in this task, then we can compute $\phi_0(r)$ by (32). Finally, by successive application of (33), we compute the remaining $\phi_i(r)$.

Example 7 (Cont.)**.** We apply the algorithm given in Section 7 and compute the solution

$$p_0(n) = (1+n)^3, \quad p_1(n) = (-3-2n)(39+51n+17n^2), \quad p_2(n) = (2+n)^3, \quad \text{and}$$
$$\phi_1(n,r) = -2(3+2n)(1+r)^4 \Big/ \big((n-r)(1+n-r)\big) \quad (39)$$

for (38); see Example 12. Together with (32) we obtain the solution

$$g(n,r) = \Big(2((3+2n)(n-r)(4+6n+2n^2+16r+21nr+7n^2r+19r^2+21nr^2+7n^2r^2-8r^4)f(n,r) - (3+2n)(2+n-r)(1+r)^4 f(1+r))\Big) \Big/ \big((n-r)(1+n-r)(2+n-r)\big) \quad (40)$$

for (31). Using `Sigma` this result can be obtained with the following function calls.

In[13]:= $\mathbf{mySum} = \sum_{r=0}^{n} \mathbf{f[r];}$

In[14]:= **CreativeTelescoping[mySum, n, {{recR, f[r]}}, recRN]**

Out[14]= $\{\{\frac{(1+n)^3}{3+2n}, -39-51n-17n^2, \frac{(2+n)^3}{3+2n}, \frac{2(4+6n+2n^2+16r+21nr+7n^2r+19r^2+21nr^2+7n^2r^2-8r^4)}{(1+n-r)(2+n-r)}f[r] - \frac{2(1+4r+6r^2+4r^3+r^4)}{(n-r)(1+n-r)}f[1+r]\}\}$

We note that the correctness of the summand recurrence (31) follows by the derivation given above. Namely, the solution (39) and (40) for (31) can be verified by simply plugging the solution (39) into (38) and verifies correctness by rational function arithmetic. If one does not trust this derivation, one may repeat the rewrite rules to get the coupled system (32) and (33) and to verify that the computed ϕ_0, ϕ_1 and ϕ_2 are indeed a solution. To this end, we can compute the recurrence

$$(n+1)^3 S(n+2) - (2n+3)(17n^2+51n+39)S(n+1) + (n+2)^3 S(n) = 0 \quad (41)$$

by summing the equation (31) with the explicitly given expressions (39) and (40) over given summation range. Most of these steps can be carried out automatically with `Sigma` by executing the following command.

In[15]:= **GenerateRecurrence[mySum, n, {{recR, f[r]}}, recRN]**

Out[15]= $\{(1+n)^3\mathtt{SUM}[n] - (3+2n)(39+51n+17n^2)\mathtt{SUM}[1+n] + (2+n)^3\mathtt{SUM}[2+n] == -4(3+2n)f[0] + \frac{2(3+2n)}{n(1+n)}f[1]\}$

Finally, we use the knowledge that $f[0] = f(n, 0) = 1$ and $f[1] = f(n, 1) = 2n(n+1)$ holds which shows that the right hand side reduces to 0. In short we computed (together with a proof) the recurrence (41) for the left hand side of the Apéry–Schmidt–Strehl identity [Str94]

$$\sum_{r=0}^{n}\sum_{s=0}^{r}\binom{n}{r}\binom{n+r}{r}\binom{k}{s}^3 = \sum_{r=0}^{n}\binom{n}{r}^2\binom{n+r}{r}^2.$$

In total we needed 1.7 seconds on a standard notebook (see Footnote 3) to produce this recurrence; more precisely, it took 1.1 seconds to get the recurrences Out[11] and Out[12] and 0.6 seconds to obtain Out[15]. Lastly, using the Zeilberger's algorithm or `Sigma` we can compute (again together with proof certificates) the same recurrence (41). Finally, checking two initial values proves the identity. □

SUMMARY: If we succeed in all our three steps, we manage to compute polynomials $p(n)$, free of r, and $g(n,r)$ with (27) such that (2) holds. By telescoping we arrive at (1). Summarizing, the steps of our algorithm are as follows.

Method 1. Creative telescoping for hypergeometric double sums.
Input: A summand $F(n,r,s)$ which is hypergeometric in n, r, and s; in addition[2] $\gamma \in \mathbb{Z}_{\geq 1}$.
Output: A recurrence of the form (1) for the sum $S(n) = \sum_r \sum_s F(n,r,s)$.

(1) Compute recurrences of the form (23) and (24) for the sum $f(n,r) := \sum_s F(n,r,s)$ by parameterized creative telescoping: Zeilberger's algorithm and its extension for hook-type recurrences. If not possible, output the comment "Failure".
(2) Based on (23) and (24), compute rational functions $\psi_i^{(j)}(n,r)$ to set up the linear system consisting of the equations (32), (33) for $1 \leq i < \delta$, and (34).
(3) Try to find a rational function $\phi_\delta(n,r)$ in n and r and polynomials $p_j(n)$, free of r, with (34); see Section 7. If not possible, output the comment "Failure".
(4) Given $\phi_\delta(n,r)$, compute the remaining $\phi_i(n,r)$ by using (32) and (33).
(5) Take $g(n,r)$ according to (27), and sum (2) over all r. RETURN the resulting recurrence (1) for $S(n)$.

4.1. **More Flexibility in Specifying Hypergeometric Double Sums.** The goal is to compute a recurrence of type (2) where for the summand $f(n,r) = h(n,r)f'(n,r)$ the following property holds. $h(n,r)$ is an expression (e.g., given as a product of binomial coefficients, factorials and Pochhammer symbols) that is hypergeometric in n and r and $f'(n,r)$ is defined to be the inner sum, i.e.,

$$f'(n,r) := \sum_s F(n,r,s).$$

Example 8. We rewrite the double sum given in Example 3 to

$$S(n) = \sum_{r=0}^{n} \binom{n}{r}\binom{n+r}{r} \sum_{s=0}^{r} \binom{k}{s}^3, \tag{42}$$

i.e., we have $f(n,k) = h(n,k)f'(n,k)$ with $h(n,r) = \binom{n}{r}\binom{n+r}{r}$ and $f'(n,r) = \sum_{s=0}^{r}\binom{k}{s}^3$. With our refined method we can compute a recurrence of the type (2) with $\gamma = 2$. □

In order to find (2) we propose a refined version of the method described above.
First one computes, as above, recurrences of the following form,

$$f'(n, r+\delta+1) = \lambda_0(n,r)f'(n,r) + \cdots + \lambda_\delta(n,r)f'(n,r+\delta), \tag{43}$$

and

$$f'(n+1,r) = \mu_0(n,r)f'(n,r) + \cdots + \mu_\delta(n,r)f'(n,r+\delta), \tag{44}$$

where the $\lambda_i(n,r)$ and $\mu_i(n,r)$ are rational functions in n and r. Moreover, since $h(n,r)$ is a hypergeometric term in n and r, we can compute rational functions $\rho_i(n,r)$ and $\nu_i(n,r)$ such that

$$h(n,r+i) = \rho_i(n,r)h(n,r) \quad \text{and} \quad h(n+i,r) = \nu_i(n,r)h(n,r) \tag{45}$$

for $i \geq 0$.

[2]One may also consider the special case $\gamma = 0$. In this case the creative telescoping problem reduces to the telescoping problem; for an example in the context of double sums we refer to Section 5.1.

Example 9 (Cont.)**.** With [PWZ96] we compute the recurrences

$$8(1+r)^2 f'(n,r) + (16+21r+7r^2) f'(n,r+1) - (2+r)^2 f'(n,r+2) = 0 \tag{46}$$

and

$$f'(n+1,r) - f'(n,r) = 0; \tag{47}$$

i.e., we are in the case $\delta = 1$. Moreover we have

$$\rho_0 = 1, \qquad \rho_1 = \frac{(n-r)(n+r+1)}{(r+1)^2}, \qquad \rho_2 = \frac{(n-r-1)(n-r)(n-r+1)(n-r+2)}{(r+1)^2(r+2)^2},$$
$$\nu_0 = 1, \qquad \nu_1 = \frac{n+r+1}{n-r+1}, \qquad \nu_2 = \frac{(n+r+1)(n+r+2)}{(n-r+1)(n-r+2)}$$

for the relations (45). □

Now we follow the same ideas as above. Namely, due to (43), (44) and (45) one can compute rational functions $\psi_i^{(j)}(n,r)$ in n and r such that for all nonnegative integers i,

$$f(n+j,r) = h(n+j,r) f'(n+j,r) = h(n,r) \sum_{i=0}^{\delta} \psi_i^{(j)}(n,r) f'(n,r+i). \tag{48}$$

To this end, one looks for polynomials $p_i(n)$, free of r, and rational function coefficients $\phi_i(n,r)$ such that with

$$g(n,r) = h(n,r)\Big(\phi_0(n,r) f'(n,r) + \cdots + \phi_\delta(n,r) f'(n,r+\delta)\Big) \tag{49}$$

the certificate recurrence (2) holds. More precisely, we look for $p_i(n)$ and $\phi_i(n,r)$ such that the relations

$$\phi_0(n,r) = \lambda_0(n,r)\rho_1(n,r)\phi_\delta(n,r+1) - \sum_{j=0}^{\gamma} p_j(n)\psi_0^{(j)}(n,r), \tag{50}$$

$$\phi_i(n,r) = \rho_1(n,r)\phi_{i-1}(n,r+1) + \rho_1(n,r)\lambda_i(n,r)\phi_\delta(n,r+1) - \sum_{j=0}^{\gamma} p_j(n)\psi_i^{(j)}(n,r) \tag{51}$$

for $1 \le i < \delta$, and

$$-\phi_\delta(n,r) + \sum_{j=0}^{\delta} \lambda_j(n,r+\delta-j)\rho_{\delta+1-j}(n,r)\phi_\delta(n,r+\delta+1-j)$$
$$= \sum_{j=0}^{\gamma} p_j(n) \sum_{i=0}^{\delta} \rho_{\delta-i}(n,r)\psi_i^{(j)}(n,r+\delta-i). \tag{52}$$

hold.

Example 10 (Cont.)**.** The $\psi(n,r)$ in (48) are given by $\psi_i(n,r) := \nu_i(n,r)$. Hence (52) reads as

$$\begin{aligned}&\frac{8(-1+n-r)(n-r)(1+n+r)(2+n+r)}{(1+r)^2(3+r)^2}\phi_1(n,r+2)\\ &\quad+\frac{(n-r)(1+n+r)(16+21r+7r^2)}{(1+r)^2(2+r)^2}\phi_1(n,r+1) - \phi_1(n,r) = p_0(n)\frac{(n-r)(1+n+r)}{(1+r)^2}\\ &\qquad + p_1(n)\frac{(1+n+r)(2+n+r)}{(1+r)^2} + p_2(n)\frac{(1+n+r)(2+n+r)(3+n+r)}{(1+n-r)(1+r)^2}.\end{aligned} \tag{53}$$

Applying the algorithm given in Section 7 we compute the solution

$$p_0(n) = (1+n)^3, \quad p_1(n) = (-3-2n)(39+51n+17n^2), \quad p_2(n) = (2+n)^3, \quad \text{and}$$
$$\phi_1(n,r) = \frac{2(3+2n)(1+r)^2(1+n+r)}{1+n-r}; \tag{54}$$

see Example 13. Using (50) we compute

$$\phi_0(n,r) = \frac{-2(3+2n)(4+6n+2n^2+16r+21nr+7n^2r+19r^2+21nr^2+7n^2r^2-8r^4)}{(1+n-r)(2+n-r)}.$$

Altogether we obtained the solution $p_i(n)$ and

$$g(n,r) = \binom{n}{r}\binom{n+r}{r}\big(\phi_0(n,r)f'(n,r)+\phi_1(n,r)f'(n,r+1)\big)$$

for (2) with $\delta = 1$ and $\gamma = 2$. □

We note that the found result (54) is slightly simpler than the one found in (39), i.e., it contains two factors less. In short, one has to reconstruct two factors less to find a solution which means that the underlying problem to solve a linear recurrence gets simpler. This observation will be further explored in Section 7.1 below for more involved examples.

5. Further Examples

5.1. **Blodgett–Andrews–Paule Sum.** We prove the identity

$$\sum_{r=0}^{n}\sum_{s=0}^{n}\binom{r+s}{r}^2\binom{4n-2r-2s}{2n-2r} = (2n+1)\binom{2n}{n}^2 \tag{55}$$

from [AP93]. Define $f(n,r) := \sum_{s=0}^{n}\binom{r+s}{r}^2\binom{4n-2r-2s}{2n-2r}$. Then by using `Sigma` or the Paule-Schorn implementation [PS95] we compute

$$\begin{aligned}&(n-r)(1+r)(1-2n+2r)f(n,r)+(18+11n+30n^2+32r-4nr+20n^2r+22r^2-8nr^2+6r^3)f(n,r+1)\\&-(2+r)(27+9n+18n^2+23r-4nr+6r^2)f(n,r+2)+2(2+r)(3+r)^2f(n,r+3)=0\end{aligned} \tag{56}$$

which holds for $0 \le r \le n-3$. Next, we compute with our double sum method

$$\begin{aligned}g(n,r) = \frac{1}{2(1+2n)^2}\Big[&(-2-r-3r^2-2r^3-2n^2(5+r)-n(7-5r-4r^2))f(n,r)\\&+(1+r)((10+18n^2+n(13-4r)+9r+4r^2)f(n,r+1)-2(2+r)^2f(n,r+2))\Big]\end{aligned}$$

such that

$$\Delta_r g(n,r) = f(n,r) \tag{57}$$

holds for $0 \le r \le n-3$. This implies that

$$\sum_{r=0}^{n-3} f(n,r) = g(n,n-2) - g(n,0).$$

Using Gosper's algorithm [Gos78] (i.e., the Paule-Schorn implementation) or `Sigma` we obtain $g(n,0) = 0$ and $g(n,n-2)+f(n,n-2)+f(n,n-1)+f(n,n) = (2n+1)\binom{2n}{n}^2$ which proves (55). In total we needed 2.5 seconds to establish this identity.

Remark: Note that the recurrence (56) does not hold for $n-2 \le r \le n$. Hence we are not allowed to sum (57) over $0 \le r \le n$; summing over the whole range would give the wrong result that the left hand side of (55) equals to 0.

5.2. **Ahlgren–Rivoal–Krattenthaler–Sum.** We prove the identity

$$\sum_{r=0}^{n}\binom{n}{r}^2\binom{2n-r}{n}\sum_{s=0}^{r}\binom{n}{s}^2\binom{n+r-s}{n} = \sum_{r=0}^{n}(1-7rH_r+7rH_{n-r})\binom{n}{r}^7 \tag{58}$$

from [KR04] which extends the family of identities from [PS03]. Define

$$f'(n,r) := \sum_{s=0}^{r}\binom{n}{s}^2\binom{n+r-s}{n},$$

$h(n,r) := \binom{n}{r}^2\binom{2n-r}{n}$, $f(n,r) := h(n,r)f'(n,r)$, and $S(n) := \sum_{r=0}^{n} f(n,r)$. Then by using `Sigma`, or the Paule-Schorn implementation of Zeilberger's algorithm [PWZ96] and a variation of it presented in Section 3, we compute the recurrence relations

$$-((1+r)^2 f'(n,r)) + (5+n+n^2+6r+2r^2)f'(n,r+1) - (2+r)^2 f'(n,r+2) = 0 \qquad (59)$$

and

$$(1+n^2+2r-2nr+2r^2)f'(n,r) - 2(1+r)^2 f'(n,r+1) + (1+n)^2 f'(n+1,r) = 0 \qquad (60)$$

that hold for all $0 \leq r \leq n$. Next we compute the certificate recurrence

$$\Delta_k g(n,k) = p_0(n,k)f(n,k) + \cdots + p_3(n,k)f(n+3,k)$$

given by

$$\begin{aligned}
p_0(n,r) &= (1+n)^4(39+33n+7n^2),\\
p_1(n,r) &= -(56667+199575n+290457n^2+223446n^3+95773n^4+21675n^5+2023n^6),\\
p_2(n,r) &= -(29445+89733n+111973n^2+73282n^3+26575n^4+5073n^5+399n^6),\\
p_3(n,r) &= (3+n)^4(13+19n+7n^2),
\end{aligned} \qquad (61)$$

and

$$g(n,r) = -\frac{(2n+1-r)(\phi_0(n,r)f(n,r)+\phi_1(n,r)f(n,r+1))}{(n+1)(n+2)(n+1-r)^3(n+2-r)^3(n+3-r)^3}h(n,r) \qquad (62)$$

where

$\phi_0(n,r) = 1267n^{16} + n^{15}(35590-13937r) + n^{14}(462869-360690r+67228r^2) + n^{13}(3700744-4292363r+1601250r^2-194306r^3) + n^{12}(20368825-31155676r+17403282r^2-4275006r^3+375697r^4) + n^{11}(81899154-154272523r+114318498r^2-42587312r^3+7684974r^4-498204r^5) + n^{10}(249131528-552266458r+506714004r^2-253972267r^3+70678802r^4-9532349r^5+435939r^6) + n^9(585775706-1477875720r+1602466002r^2-1009881505r^3+385686252r^4-81531645r^5+7803065r^6-226688r^7) + n^8(1078149331-3015101902r+3728141948r^2-2822002195r^3+1387276771r^4-410819161r^5+61998191r^6-3794976r^7+48685r^8) + n^7(1562549948-4739718717r+6484904428r^2-5688812977r^3+3453596348r^4-1352286608r^5+287672055r^6-27878506r^7+797964r^8+11193r^9) + n^6(1782583091-5761209036r+8487438846r^2-8355640898r^3+6074061295r^4-3045608700r^5+862369998r^6-117912992r^7+5576692r^8+110678r^9-8232r^{10}) + 78(61128-257256r+424368r^2-342245r^3+133135r^4-605158r^5+874931r^6-481375r^7+108900r^8-3013r^9-2506r^{10}+283r^{11}) + n^5(1588987638-5396124481r+8344272508r^2-8918261604r^3+7554494444r^4-4772804891r^5+1743203551r^6-316363171r^7+21773661r^8+432467r^9-85561r^{10}+1267r^{11}) + n^4(1088253105-3847125006r+6102147702r^2-6815183791r^3+6526854398r^4-5179434740r^5+2403086680r^6-558320797r^7+52079159r^8+813797r^9-366821r^{10}+11517r^{11}) + n^3(554906820-2034079575r+3250687390r^2-3620706197r^3+3753196026r^4-3786628463r^5+2226983897r^6-648088722r^7+78309002r^8+639830r^9-831274r^{10}+41326r^{11}) + n(44801424-178444188r+286511076r^2-268017747r^3+231495788r^4-453726129r^5+455141086r^6-202147723r^7+37738137r^8-523202r^9-704439r^{10}+63933r^{11}) + n^2(198939024-757536768r+1209747438r^2-1270694237r^3+1316530531r^4-1754875242r^5+1324652652r^6-477128371r^7+72428076r^8-131397r^9-1051417r^{10}+73167r^{11})$

and

$\phi_1(n,r) = (1+r)^2(1267n^{14} + n^{13}(34323-13937r) + n^{12}(427279-349287r+64694r^2) + n^{11}(3239142-3997785r+1495784r^2-166432r^3) + n^{10}(16702404-27663162r+15686820r^2-3539388r^3+233555r^4) + n^9(61957608-129089860r+98653180r^2-33877490r^3+4514021r^4-173215r^5) + n^8(170471516-428918244r+414295520r^2-192602705r^3+38897286r^4-3007648r^5+55937r^6) + n^7(353346582-1043782832r+1223781806r^2-722527950r^3+196740999r^4-23008400r^5+870673r^6+3591r^7) + n^6(554331233-1883343858r+2606958078r^2-1877626092r^3+646725800r^4-101747168r^5+5872142r^6+36536r^7-6965r^8) + 78(61128-379512r+1061136r^2-1705493r^3+1411049r^4-592120r^5+110692r^6-366r^7-2789r^8+283r^9) + n^5(654872133-2519285191r+4035037044r^2-3448656883r^3+1443487563r^4-286571923r^5+22400799r^6+149302r^7-75311r^8+1267r^9) + n^4(573379725-2467242453r+4503468974r^2-4476586363r^3+2215350588r^4-533046699r^5+52847951r^6+308630r^7-337012r^8+11517r^9) + n^3(360735780-1719341103r+3534592606r^2-4024522064r^3+2308453137r^4-654819699r^5+78949756r^6+325414r^7-799433r^8+41326r^9) + n(40033440-228909132r+581493852r^2-840158733r^3+621255632r^4-231745735r^5+38124358r^6-26621r^7-746298r^8+63933r^9) + n^2(154137600-807300900r+1851811578r^2-2386528406r^3+1563194205r^4-512350625r^5+$

$72945156r^6 + 134868r^7 - 1060651r^8 + 73167r^9)).$

This shows that the left hand side of (58) fulfills the recurrence relation

$$\begin{aligned}(1+n)^4(39+33n+7n^2)S(n)\\ -(56667+199575n+290457n^2+223446n^3+95773n^4+21675n^5+2023n^6)S(n+1)\\ -(29445+89733n+111973n^2+73282n^3+26575n^4+5073n^5+399n^6)S(n+2)\\ +(3+n)^4(13+19n+7n^2)S(n+3)=0. \quad (63)\end{aligned}$$

The total caluclation time of this recurrence took 4.8 seconds; more precisely, 1.3 seconds for recurrences (59) and (60) of the inner sum and 3.8 seconds for the recurrence (59) of the double sum. In [PS03] the same recurrence relation (63) has been derived for the right hand side of (58). Checking the first three initial values proves (58).

6. The Method Extended to Multiple Sums

Based on what we said about single and double sums we are in the position to deal with the general problem stated at the beginning of Section 1.
Given a summand $F(m, n, r, s_1, \dots, s_e)$ which is hypergeometric in m, n, r and the s_i, **compute** a P-finite recurrence

$$p_\gamma(m,n)S(m,n+\gamma)+\cdots+p_0(m,n)S(m,n)=0 \quad (64)$$

(resp. a P-finite recurrence (6)) which is satisfied by the sum

$$S(m,n)=\sum_r\sum_{s_1}\cdots\sum_{s_e}F(m,n,r,s_1,\dots,s_e).$$

As with double sums the overall goal of the method is to compute a certificate recurrence of the form

$$p_\gamma(m,n)f(m,n+\gamma,r)+\cdots+p_0(m,n)h(m,n,r)=\Delta_r g(m,n,r) \quad (65)$$

(resp. (5)) where we define $f(m,n,r)$ as

$$f(m,n,r):=\sum_{s_1}\cdots\sum_{s_e}F(m,n,r,s_1,\dots,s_e), \quad (66)$$

and where $g(m,n,r)$ is suitably chosen. Then from (65) (resp. (5)) the desired recurrence (64) (resp. (6)) for $S(m,n)$ is obtained by summation over all r.

To find (65) we proceed analogously to the double sum case. Namely, we first try to derive recurrences of the form

$$f(m,n,r+\delta+1)=\lambda_0(m,n,r)f(m,n,r)+\cdots+\lambda_\delta(m,n,r)f(m,n,r+\delta), \quad (67)$$

and

$$f(m,n+1,r)=\mu_0(m,n,r)f(m,n,r)+\cdots+\mu_\delta(m,n,r)f(m,n,r+\delta), \quad (68)$$

where the $\lambda_i(m,n,r)$ and $\mu_i(m,n,r)$ are rational functions in m, n and r. Afterwards we apply the same method as in the double sum case in order to compute all the components for the certificate recurrence (65).

Otherwise, if we look for (6), we suppose that we have computed besides (67) and (68) a hook-type recurrence of the form

$$f(m+1,n,r)=\nu_0(m,n,r)f(m,n,r)+\cdots+\nu_\delta(m,n,r)f(m,n,r+\delta). \quad (69)$$

Then, we can represent the left hand side of (5) in terms of the generators $f(m,n,r), \dots, f(m,n,r+\delta)$. More precisely, as in the double sum case (29) we can compute rational functions $\psi_i^{(j)}(m,n,r)$ in m, n and r such that

$$p_\gamma(m,n)f(m+1,n,r)+\sum_{j=0}^{\gamma-1}p_j(m,n)f(m,n+j,r)=\sum_{i=0}^{\delta}f(m,n,r+i)\sum_{j=0}^{\gamma}p_j(m,n)\psi_i^{(j)}(m,n,r).$$

holds. Given this representation, we can proceed as in the double sum case in order to compute all the components for the certificate recurrence (6).

Note that our refined method in Subsection 4.1 can be carried over analogously to the multiple sum case.

Summarizing, in order to apply the above strategy there remains the task to compute the recurrences of the type (67), (68) and (69). This gives rise to the following situations.

Base case: If $f(n,m,r)$ is a single sum, i.e., $e=1$, we can apply Zeilberger's algorithm to get (67), or a variation of it to obtain (68); see Section 2. Similarly, we can compute (69) by a slightly more general variation; see Section 3.

Reduction: Otherwise, we apply again the method described in this section, but this time for a multi-sum reduced by one sum. This means that by recursion we end up eventually in the base case.

Example 11. We illustrate how one can prove identity

$$\sum_{r=0}^{n}\binom{n}{r}^2\binom{2n-r}{n}\sum_{s=0}^{r}\binom{n}{s}^2\binom{n+r-s}{n}\sum_{k=0}^{s}\binom{n}{k}^2\binom{n+s-k}{n}$$
$$=\sum_{r=0}^{n}(1-9rH_r+9rH_{n-r})\binom{n}{r}^9 \tag{70}$$

from [KR04]. Define

$$f_1(n,r,s):=\sum_{k=0}^{s}\binom{n}{k}^2\binom{n+s-k}{n},$$

$h_1(n,r,s):=\binom{n}{s}^2\binom{n+r-s}{n}$, $f_2(n,r):=\sum_{s=0}^{r}h_1(n,r,s)f_1(n,r,s)$, $h_2(n,r):=\binom{n}{r}^2\binom{2n-r}{n}$, and let $S(n)$ be the left hand side of (70), i.e., $S(n):=\sum_{r=0}^{n}h_2(n,r)f_2(n,r)$. In order to compute a recurrence for $S(n)$, we apply the machinery of `Sigma` or the algorithms [PWZ96] and [Pau21] (see Sections 2 and 3) to obtain the recurrence relations

$$-((1+s)^2f_1(n,r,s))+(5+6s+2s^2+n+n^2)f_1(n,r,s+1)-(2+s)^2f_1(n,r,s+2)=0, \tag{71}$$
$$(1+2s+2s^2-2sn+n^2)f_1(n,r,s)-2(1+s)^2f_1(n,r,s+1)+(1+n)^2f_1(n+1,r,s)=0 \tag{72}$$

which are equivalent to (4) and (7) with $f_1(n,r,s)=f_1(n,s)$. Further we get trivially

$$f_1(n,r,s)-f_1(n,r+1,s)=0. \tag{73}$$

Given (71) and (73) we apply our double sum method from Section 4 to compute the recurrence relation

$$\begin{aligned}(1+r)^2(2+r)^2f_2(n,r)-(2+r)^2(14+3n+3n^2+12r+3r^2)f_2(n,r+1)\\ +(133+n^4+200r+115r^2+30r^3+3r^4-n^3(3+2r)+n^2(13+12r+3r^2)\\ +n(17+14r+3r^2))f_2(n,r+2)-(3+r)^4f_2(n,r+3)=0.\end{aligned} \tag{74}$$

As explained above, we compute in addition the recurrence relation

$$\begin{aligned}-((1+r)^2(2n^4-n^3(7+10r)+n^2(20+42r+24r^2)-n(15+68r+78r^2+28r^3)\\ +2(6+24r+40r^2+28r^3+7r^4))f_2(n,r))+(91+5n^6+450r+971r^2+1084r^3+659r^4+210r^5+28r^6-3n^5(3+8r)\\ +n^4(29+66r+57r^2)-n^3(-9+64r+123r^2+70r^3)+n^2(101+210r+246r^2+174r^3+57r^4)\\ -n(54+362r+633r^2+520r^3+222r^4+42r^5))f_2(n,r+1)\\ -(2+r)^4(5+5n^2+14r+14r^2-2n(2+7r))f_2(n,r+2)=-((1+n)^4(1+r)^2f_2(n+1,r))\end{aligned} \tag{75}$$

by using besides (71) and (73) the recurrence relation (72). Given the two recurrences (74) and (75), we are in the position to apply our method again as in the double sum case. This gives the recurrence

$$
\begin{gathered}
(1+n)^6(2+n)^2(126186232584+359847089412n+447038924854n^2+315988281882n^3\\
+139000794255n^4+38967288138n^5+6799034214n^6+675116208n^7+29211759n^8)S(n)\\
+2(2+n)^2(9449901867223980+65177937447506574n+206795641058521957n^2\\
+400003560150467208n^3+526934624462960841n^4+500054178553882862n^5\\
+352526028922986741n^6+187547382614273601n^7+75664907849081395n^8+23037690482849736n^9\\
+5211078007675644n^{10}+849237300832941n^{11}+94267319550444n^{12}+6380425909278n^{13}\\
+198698384718n^{14})S(n+1)-3(99381765767163760+720338927889449008n+2427055018593335824n^2\\
+5046939121521308492n^3+7251199169750148467n^4+7634448497599004444n^5+6094496182619292815n^6\\
+3763786379996759276n^7+1817742639895041823n^8+688977924255751768n^9+204313397754918826n^{10}\\
+46914883776289584n^{11}+8179105939324551n^{12}+1046803624503588n^{13}+92772291582963n^{14}\\
+5087571879456n^{15}+130079962827n^{16})S(n+2)\\
-(3+n)^2(1657317485213296+10358247512403136n+29676907405770592n^2\\
+51669502990568780n^3+61088527857001943n^4+51897294744470249n^5+32681221486607779n^6\\
+15503112379989763n^7+5569174593112480n^8+1508250655288332n^9+303253251903666n^{10}\\
+43913846933991n^{11}+4331266602147n^{12}+260552661525n^{13}+7215304473n^{14})S(n+3)\\
+(3+n)^2(4+n)^6(3576422026+16265263120n+32031965452n^2+35670510738n^3\\
+24565622625n^4+10714664718n^5+2891150010n^6+441422136n^7+29211759n^8)S(n+4)=0. \qquad (76)
\end{gathered}
$$

For the calculation of the recurrences (71) and (72) of the innermost sum we needed 1.3 seconds, for the recurrences of the double sum we used 5.5 seconds and for the final output recurrence (76) of the triple sum we used 8.1 seconds. Thus the full calculation could be accomplished in less than 15 seconds. By applying the summation package Sigma, see [PS03], we arrive at the same recurrence for the right hand side of (70). Checking the first initial values proves identity (70). □

7. Speeding Up our Multi-Sum Method

The computational backbone concerning efficiency of our method is introduced in this section. As elaborated in Sections 4 and 6 the creative telescoping problem (2) for double sums and more generally multi-sums can be reduced efficiently to the problem to solve parameterized linear recurrences of the form (34) or (52) by using rewrite rules combing from the linear (hook-type) recurrences of the summand. In view of (34) and (52) we consider the following problem:

Given a rational function field $\mathbb{K}(r)$, $a_0(r),\dots,a_\delta(r)\in\mathbb{K}[r]$ with $a_0\,a_r\neq 0$ and $f_0(r),\dots,f_\gamma(r)\in\mathbb{K}[r]$, **find** all solutions $c_0,\dots,c_\gamma\in\mathbb{K}$ and $g(r)\in\mathbb{K}(r)$ of the parameterized recurrence

$$a_\delta(r)\,g(r+\delta)+a_{\delta-1}(r)\,g(r+\delta-1)+\dots+a_0(r)\,g(r)=c_0\,f_0(r)+\dots+c_\gamma\,f_\gamma(r). \qquad (77)$$

In all our examples the full calculation of our proposed summation method, *excluding* the task to find a solution of (77), took at most 2 seconds. In a nutshell, almost all of the calculation time is used to solve the underlying parameterized recurrence.

In the following we introduce the basic mechanism implemented within Sigma and present various improvements that lead to significant speed-ups. For instance, combining all these enhancements finally enabled us to compute recurrences of the double sum and triple sum on the left hand sides of (58) and (70) in less than 5 and 15 seconds, respectively.

The basic algorithm works as follows.

(1) In a first step we compute a *denominator bound for* (77), i.e., a non-zero polynomial $d(r)\in\mathbb{K}[r]$ such that for any solution $g(r)\in\mathbb{K}(r)$ and $c_i\in\mathbb{K}$ with (77) we have $d(r)\,g(r)\in\mathbb{K}[r]$; this task can be accomplished by Abramov's algorithm in [Abr89b], [Abr95]. Here we use the equivalent compact formula given in [CPS08]:

$$d(r) = \gcd\Big(\prod_{i=0}^{D} a_0(r+i), \prod_{i=0}^{D} a_\delta(r-\delta-i)\Big) \tag{78}$$

where $D \in \mathbb{Z} \cup \{-\infty\}$ is the dispersion of the coefficients $a_\delta(r)$ and $a_0(r)$ defined by

$$D = \max(h \in \mathbb{Z}_{\geq 0} \mid \gcd(a_\delta(r-\delta), a_0(r+h)) = 1);$$

for a generalized formula that holds for coupled systems in $\Pi\Sigma$-extensions [Kar81, Sch01] we refer to [MS18]. Note that in basically all our applications the polynomials $a_0(r)$ and $a_\delta(r)$ are already given in factored form and thus the gcd in (78) can be read off. In particular the result $d(r)$ can be also given directly in its factored form, which we will need in (87) below.

Then, given such a denominator bound $d(r)$, it suffices to look for all $g'(r) \in \mathbb{K}[r]$ and $c_i \in \mathbb{K}$ with

$$\frac{a_\delta(r)}{d(r+\delta)}\, g'(r+\delta) + \cdots + \frac{a_1(r)}{d(r+1)}\, g'(r+1) + \frac{a_0(r)}{d(r)}\, g'(r) = c_0\, f_0(r) + \cdots + c_\gamma\, f_\gamma(r). \tag{79}$$

Namely, given all such solutions $g'(r)$ and c_i, we obtain all the solutions of (77) with $\frac{g'(r)}{d(r)}$ and c_i.

(2) The next step consists of bounding the polynomial degree of the possible solutions $g'(r) \in \mathbb{K}[r]$, say with $b \in \mathbb{N}$. In [Abr89a, Pet92, SA95, PWZ96] several algorithms are introduced that find such a *degree bound* b *for* (79). Note that all these algorithms are equivalent; see [PW00].

(3) Finally, substituting the possible solutions $g'(r) = g_b\, r^b + g_{b-1}\, r^{b-1} + \cdots + g_0$ into (79) leads by coefficient comparison to a linear system of equations. Solving this system enables one to construct all the solutions for (79) and hence for (77). More precisely, one can compute a basis of the $\mathbb{K}$-vector space

$$V = \{(c_1, \ldots, c_\delta, g) \in \mathbb{K}^\delta \times \mathbb{K}(r) \mid \text{equation (77) holds}\}$$

whose dimension is at most $\delta + 1 + \gamma$.

Example 12 (Cont. Example 7)**.** Following the algorithm from above we compute $\Phi_1(n,r) \in \mathbb{Q}(n)(r)$ and $p_i(n) \in \mathbb{Q}(n)$ such that (38) holds: First we compute the denominator bound $d(r) = (n-r)(n+1-r)$ using the formula (78). As a result, (79) reads as

$$\begin{aligned}8(-2+n-r)(-1+n-r)(n-r)(1+n-r)(2+r)^4(2+n+r)(3+n+r)g'(r+2)\\ +(-1+n-r)(n-r)(1+n-r)(3+r)^4(2+n+r)(16+21r+7r^2)g'(r+1)\\ -((n-r)(1+n-r)(2+r)^4(3+r)^4)g'(r) = p_0(n-r)(1+n-r)(2+r)^4(3+r)^4\\ +p_1(1+n-r)(2+r)^4(3+r)^4(2+n+r)+p_2(2+r)^4(3+r)^4(2+n+r)(3+n+r).\end{aligned} \tag{80}$$

Next, we compute the degree bound $b = 4$ for the polynomial solutions $g'(r) \in \mathbb{Q}(n)[r]$. Finally, substituting the possible solutions $g'(r) = \sum_{i=0}^4 g_i' r^i$ and $p_i \in \mathbb{Q}(n)$ into (80) leads by coefficient comparison to a linear system with 13 equations in 8 unknowns $(p_0, p_1, p_2, g_0', \ldots, g_4')$. Note that this system requires 23576 bytes of memory in the computer algebra system Mathematica . To this end, solving this system gives the solution $p_0 = (1+n)^3$ $p_1 = (-3-2n)(39+51n+17n^2)$, $p_2(n) = (2+n)^3$, and $g'(r) = -2(3+2n)(1+r)^4$, and hence the solution (39) for (38). □

Example 13 (Cont. Example 10)**.** Completely analogously, we solve (53). Namely, we compute the denominator bound $d(r) = n+1-r$, afterwards we consider the corresponding problem of the form (79), compute the degree bound $b = 3$, and set up a linear system with 10 equations in 7 unknowns. Solving this system gives the solution (54) for (53). Note that in comparison to Example 12 the degree of the denominator bound and hence also the degree bound is reduced by one. This leads us to a smaller equation system, namely 10×3 instead of 13×8; in Mathematica we need only 15408 bytes instead of 23576 bytes to store the system. □

7.1. **Preprocessing of the input sums.** The observation described in Example 13 holds in all our examples. Pulling out expressions from the inner sum, like (22) and (42), and applying our refined summation method from Subsection 4.1 amounts to find a solution (77) with a smaller degree of the denominator. In particular this reduces considerably the size of the linear system and the amount of time to find the solutions.

Example 14 (Cont. Subsection 5.2)**.** In order to compute (61) and (62), we apply our method in Subsection 4.1 which reduces to a problem of the type (77). In order to solve this problem, we compute the denominator bound

$$d(r) = (n+1-r)^3(n+2-r)^6(n+3-r)^3 \tag{81}$$

and the degree bound $b = 15$. This finally gives a linear system with 30 equations in 20 unknowns. In Mathematica this system requires 0.67 MB of memory. Solving this system can be carried out in 5.7 seconds using 28 MB memory; compare the first row of Table 2.

By doing the same computations without pulling out factors from the innermost sum, i.e., considering the sum

$$\sum_{r=0}^{n}\sum_{s=0}^{r}\binom{n}{r}^2\binom{2n-r}{n}\binom{n}{s}^2\binom{n+r-s}{n}, \tag{82}$$

we compute the denominator bound

$$d(r) = (n-r)^3(n+1-r)^6(n+2-r)^6(n+3-r)^3 \tag{83}$$

and the degree bound $b = 21$; for the properties of the linear system see the first row of Table 1. □

Example 15 (Cont. Example 11)**.** In order to find the recurrence (76) for the triple sum $S(n)$, we apply our refined method in Subsection 4.1. We get the denominator bound

$$d(r) = (n+1-r)^3(n+2-r)^6(n+3-r)^6(n+4-r)^3(r+1)^2 \tag{84}$$

and the degree bound $b = 25$. For the linear system see the first row of Table 4.

Applying our method without pulling out factors, i.e., considering the sum

$$\sum_{r=0}^{n}\sum_{s=0}^{r}\sum_{k=0}^{s}\binom{n}{s}^2\binom{n+r-s}{n}\binom{n}{r}^2\binom{2n-r}{n}\binom{n}{k}^2\binom{n+s-k}{n} \tag{85}$$

leads us to a denominator bound

$$d(r) = (n-1-k)^3(n-r)^6(n+1-r)^9(n+2-r)^9(n+3-r)^6(n+4-r)^3 \tag{86}$$

and a degree bound $b = 41$; for the properties of the linear system see the first row of Table 3. □

TABLE 1. Double sum (82) (without preprocessing)

Improvements	equs×unknowns	size of system	total time[3]	total memory[3]
None	38×26	1.1 MB	12.2 s	28 MB
I	26×20	0.36 MB	4.0 s	23 MB
II	25×26	0.65 MB	5.8 s	26 MB
I,II	19×20	0.27 MB	2.9 s	22 MB
I^+	16×14	0.12 MB	1.9 s	16 MB
I^+,II	13×14	0.12 MB	1.8 s	16 MB

TABLE 2. Double sum on the left hand side of (58) (with preprocessing)

Improvements	equs×unknowns	size of system	total time[3]	total memory[3]
None	30×20	0.67 MB	5.7 s	28 MB
I	21×17	0.21 MB	2.7 s	28 MB
II	19×20	0.36 MB	3.3 s	29 MB
I,II	16×17	0.20 MB	2.4 s	29 MB
I^+	16×14	0.12 MB	1.9 s	19 MB
I^+,II	13×14	0.12 MB	1.9 s	19 MB

[3]Total time and total memory means the amount of time and memory that is needed to solve the corresponding problem (77); see (34) and (52); the time to set up this equation is almost constant and thus ignored. All the computations have been done with a standard notebook (11th Gen Intel® Core™ i7-1185G7 @ 3.00GHz × 8 with 16 GB memory) using the computer algebra system Mathematica 13.0.

Table 3. Triple sum (85) (without preprocessing)

Improvements	equs×unknowns	size of system	total time[3]	total memory[3]
None	81×47	12.0 MB	243 s	100 MB
I	45×29	1.7 MB	28 s	43 MB
II	46×47	4.9 MB	67 s	45 MB
I,II	28×29	1.0 MB	13 s	34 MB
I^+	24×20	0.45 MB	6.5 s	24 MB
I^+,II	19×20	0.44 MB	6.5 s	24 MB

Table 4. Triple sum in (70) (with preprocessing)

Improvements	equs×unknowns	size of system	total time[3]	total memory[3]
None	52×31	3.2 MB	46 s	51 MB
I	37×25	1.1 MB	16 s	40 MB
II	30×31	1.4 MB	16 s	34 MB
I,II	24×25	1.0 MB	8.2 s	32 MB
I^+	24×20	0.4 MB	5.6 s	27 MB
I^+,II	19×20	0.4 MB	5.5 s	24 MB

7.2. **Heuristic Check for the number of solutions.** Usually, the field $\mathbb{K}$ contains additional parameters like $\mathbb{K} = \mathbb{Q}(n)$, more generally say $\mathbb{K} = \mathbb{Q}(x_1, \dots, x_e)$. In this case, the bottleneck of the described algorithm is step **(3)**. Suppose that we have computed a denominator bound $d(r) \in \mathbb{K}[r] \setminus \{0\}$ and a degree bound $b \in \mathbb{N}$ as described above. Then one can carry out the following speedups in step **(3)**.

Given $d(r)$ and b, construct the linear system of equations with coefficients being polynomials in $\mathbb{Z}[x_1, \dots, x_e]$ as described in **(3)**. Then, inspired by [MS18, RZ04], we can check with inexpensive computations if a non-trivial solution for (79) and hence for (77) exists. More precisely, take a random prime number p (sufficiently large) and random numbers $q_1, \dots, q_e$ from the finite field $\mathbb{F}_p$ with p elements. Afterwards, replace the parameters x_i with q_i in our linear equation system, and solve the system in the prime field $\mathbb{F}_p$.

Remark 2. Setting up the system with all the variables arising and afterwards performing the substitutions $x_i \mapsto q_i$ requires also a certain amount of computation time. Thus we first carry out the substitution and afterwards derive the linear system with almost negligible effort. $\Diamond$

Suppose that we find s linearly independent solutions of the underlying system in the finite field $\mathbb{F}_p$. Then the crucial observation is that there are at most s solutions for (79) and thus for (77) in the original field $\mathbb{K}(x)$; usually the determined s agrees with the number of solutions for (77) up to some unlucky cases that we have never encountered so far.

Hence with our check we obtain the following result:

- If $s = 0$, there are no non-trivial solutions for (77) and we can stop.
- Otherwise, we conclude, or more precisely, suppose that there are exactly s solutions for (77); if there are less, we will discover this later. With this information we proceed with our next improvement.

Remark 3. In general, one does not know (or is too lazy to predict) in advance the order γ for which a recurrence (1) of the given double or multi-sum can be computed. One therefore starts with $\gamma = 1$ (or even $\gamma = 0$ in case a telescoping solution exist) and increments γ step-wise until one finds the desired solution. In this regard, the heuristic check introduced above is extremely convenient to discover the non-existence of a solution without wasting too much calculation time. E.g., given the (hook-type) recurrences (74) and (75), it takes only 2.1 seconds to find out that our method fails to find a recurrence for the triple sum (70) by taking the instances $\gamma = 0, 1, 2, 3$. $\Diamond$

7.3. **Improvement I: Produce an optimal denominator and degree bound.** Under the assumption that there exist exactly s linearly independent solutions of (77), we try to minimize the degree of the denominator bound $d(r)$ and to minimize the degree bound b as follows. Compute a complete factorization of $d(r)$ given in (78), i.e.,

$$d(r) = d_1(r)^{m_1} \dots d_u(r)^{m_u} \tag{87}$$

where the irreducible polynomials $d_i(r)$ occur with multiplicity $m_i > 0$ in $d(r)$; as stated earlier, this can be done efficiently if the coefficients $a_0(r)$ and $a_\delta(r)$ in (77) are given already in factored form (which is usually the case). Then we test if also $d'(r) := d(r)/d_u(r)$ is a denominator bound by applying our **Heuristic Check** for[4] $d'(r)$ and $b-1$; suppose that we have obtained s' solutions. If s is equal to s', also $d'(r)$ is very likely a denominator bound for (77) — except for some rare cases. In this case, one may go on with $d'(r)$ and $b-1$ and cancel more and more factors $d_u(r)$ until the multiplicity m_u of the factor $d_u(r)$ is zero or in our **Heuristic Check** we get a different number of solutions than s.

Remark 4. In `Sigma` this search is speeded up with a binary search tactic: First, we consider the multiplicity $\lambda = \lfloor m_u/2 \rfloor$. If the number of solutions during our heuristic check remains s, we search recursively for the minimal multiplicity between $\lambda \in \{1, \dots, \lfloor m_u/2 \rfloor\}$. Otherwise, we search within the range $\lambda \in \{\lfloor m_u/2 \rfloor + 1, \dots, m_u\}$. $\diamondsuit$

In this way, we shall obtain an improved denominator bound, say $d_1(r)^{m_1} \dots d_{u-1}^{m_{u-1}}(r)\, d_u(r)^{\mu_u}$, where the multiplicity μ_u of the factor $d_u(r)$ is minimal. Analogously we proceed with the remaining irreducible factors. Finally, we arrive, up to unlucky cases, at a denominator bound, say $d'(r)$, whose degree is minimal.
Next, we fix $d'(r)$ and reduce with the same tactic the degree bound b to b' until our **Heuristic Check** tells us that the degree bound b' is minimal. Note that during this minimization process the number linearly independent solutions within the heuristic check always remains s.

To this end, we go on with step **(3)** by using $d'(r)$ and b'. Suppose that we find s' linearly independent solutions: If $s = s'$, we have found all solutions. Otherwise, only the situation $s' < s$ may arise, i.e., we might have lost some solutions. In this case, we repeat **(3)** with the original denominator bound d and degree bound b; note that this (much more involved) situation never happened in our computations so far.

Remark 5. For our applications in Sections 4 and 6 it suffices to find only one solution for (77), i.e., we do not care if $s \neq s'$ as long as we get non-trivial solutions with $s' > 0$. $\diamondsuit$

Example 16. *Double sum on the left hand side of* (58)*:* Given $d(r)$ from (81) and $b = 21$, we apply the **Heuristic Check**. This test tells us that there are $s = 1$ non-trivial solutions. Applying **Improvement I** gives the *sharp* denominator bound $d'(r) = (n+1-r)^3(n+2-r)^3(n+3-r)^3$ and degree bound $b' = 12$; for the properties of the linear system see the second row of Table 2. With the same strategy we get the following results.
Double sum (82)*:* We get the *sharp* bounds $d'(r) = (n-r)^3(n+1-r)^3(n+2-r)^3(n+3-r)^3$ and $b' = 15$; for the properties of the linear system see the second row of Table 1.
Triple sum in (70)*:* We get the *sharp* bounds $d'(r) = (n+1-r)^3(n+2-r)^3(n+3-r)^3(n+4-r)^3$ and $b' = 19$; for the properties of the linear system see the second row of Table 4.
Triple sum (85)*:* We get the *sharp* bounds $d'(r) = (n-1-r)^3(n-r)^3(n+1-r)^3(n+2-r)^3(n+3-r)^3(n+4-r)^3$, $b' = 23$; for the properties of the linear system see the second row of Table 3. $\square$

7.4. **Improvement II: producing an optimal system.** We observe that for all our linear systems with u equations in v unknowns u is much bigger as v (without **Improvement I** it is almost twice as big). Under the assumption that there are $s > 0$ linearly independent solutions it follows that $u - v - s$ equations can be removed. This observations leads us to remove step by step unnecessary equations. More precisely, we consider iteratively each equation and test if it can be removed without changing the solution set. If yes, we obtain a linear system with one equation less, and continue to check the remaining equations with this system. Otherwise, we go on without removing this equation.

[4]Note that if $d'(r)$ is a denominator bound, also $b-1$ is a degree bound for the solutions of (79). Namely, if we can reduce the degree of the "denominator" $d(r)$ by one we can also reduce the degree of the possible "numerator" by one.

The crucial point is that this test can be carried out cheaply as follows. We test with our **Heuristic Check** if the number of linearly independent solutions s of the system, in which this equation is removed, equals to s. If yes, the set of solutions is the same — up to some rare cases. Otherwise, we obtain more solutions ($s' > s$), i.e., removing this equation is not possible. Following this strategy we obtain a linear system with $v - s$ equations in v unknowns. □

To this end, we may solve the reduced system of equations symbolically.

Remark 6. The following two remarks are in place:
• If one solves the system symbolically with Gauss-elimination, the unnecessary equations would have been eliminated implicitly – but in a quite expensive manner.
• Eliminating the unnecessary equations in different orders leads to different systems. In particular, there are tremendous differences in the time/memory behavior how these different systems can be solved symbolically. After testing various different strategies the following one turned out to be rather convincing: try to eliminate equations first which are given by the coefficients of lowest degree during the coefficient comparison in step **(3)**. ♢

Example 17. For the speedups using **Improvement II**, we refer to the Tables 1–4; more precisely, the entries in the third row (without **Improvement I**) and the entries in the fourth row (together with **Improvement I**). □

Summarizing, applying **Improvements I** and **II** in combination with **Preprocessing** (pulling out factors of the multi-sum) can improve substantially our multi-sum method.

7.5. **Improvement I$^+$: predict contributions of the numerator solution.** The proposed summation algorithms in Sections 4 and 6 start with the calculation of recurrences of univariate hypergeometric sums which can be also carried out, e.g., with the Paule-Schorn implementation [PS95] and its enhancements to deal with parameteried telescoping, as described in Section 3. As it turns out, the specialized algorithms for univariate hypergeometric summation are still superior to our methods with all the above improvements. To understand this exceptional behavior, we note the following. Gosper's algorithm [Gos78, Pau95, PWZ96, CPS08], the backbone of the classical approach, relies on finding a rational solution $g(r) \in \mathbb{K}(r)$ of the first-order linear recurrence

$$b(r)g(r+1) - g(r) = 1 \tag{88}$$

where $t(r)$ is a hypergeometric term (usually built by a product of factorials, binomial coefficients, Pochhammer symbols) with $\frac{t(r+1)}{t(r)} = b(r) \in \mathbb{K}(r)$. In order to find such a rational solution $g(r)$, the following steps are carried out [Gos78, PWZ96, Pau95, CPS08]:
(i) One computes the Gosper representation of $b(r)$, i.e., non-zero polynomials $d(r), p(r), q(r) \in \mathbb{K}[r]$ with

$$b(r) = \frac{d(r+1)}{d(r)} \frac{p(r)}{q(r)}$$

such that $\gcd(p(r), q(r+h)) = 1$ holds for all non-negative integers h.
(ii) Next, one decides constructively, if there exists a polynomial $\gamma(r) \in \mathbb{K}[r]$ such that

$$p(r)\,\gamma(r+1) - q(r-1)\,\gamma(r) = d(r)$$

holds. Here one essentially proceeds as in our general procedure of step (2) given at the beginning of Section 7.
(ii) If there is no $\gamma(r) \in \mathbb{K}[r]$, then this implies that there is no $g(r) \in \mathbb{K}(r)$ with (88). Otherwise one obtains the rational solution

$$g(r) = \frac{q(r)\gamma(r)}{d(r)}. \tag{89}$$

In other words, $d(r)$ is a denominator bound of (88) and $g'(r) = q(r)\,\gamma(r)$ is the numerator contribution where $q(r)$ has been predicted by the Gosper ansatz. This result can be further improved by refining the Gosper representation to computing the Gosper-Petkovšek representation [Pet92, PWZ96, CPS08] in step (i) where in addition $\gcd(p(r), d(r)) = \gcd(q(r), d(r+1)) = 1$ holds. As a consequence the predicted numerator contribution $q(r)$ in $g'(r)$ does not cancel with the denominator bound $d(r)$. Further, as elaborated, e.g., in [PWZ96] this implies that among all possible choices of the Gosper representation, the degree $d(r)$ is minimal, i.e., we come close to a sharp denominator bound. This does not mean that there may still cancellations happen between $\gamma(r)$ and $d(r)$, but we have not found such an example so far. Comparing with our approach above, and knowing that we always obtain the optimal denominator bound $d(r)$, it is precisely the prediction of the numerator contribution $q(r)$ that makes Gosper's algorithm and all their variants, like Zeilberger's creative telescoping approach superior. A natural idea is to incorporate this extra feature to the general case to solve linear difference equations of the form (77). This leads to

Improvement I$^+$: Given the recurrence (77) we set $b(r)=\frac{a_\delta(r)}{a_0(r)}\in\mathbb{K}(r)$ and compute the non-zero polynomials $p(r),q(r),d(r)\in\mathbb{K}[r]$ of the generalized Gosper-Petkovšek representation

$$b(r)=\frac{d(r+\delta)}{d(r)}\frac{p(r)}{q(r)}$$

where $\gcd(p(r),q(r+h\,\delta))=1$ holds for all non-negative integers h and where, in addition, $\gcd(p(r),d(r))=\gcd(q(r),d(r+\delta))=1$. This can be accomplished by the general algorithm presented in [ABPS21, Thm 2] for $\Pi\Sigma$-extensions; the calculation of the polynomial $a(r)$ can be skipped therein. In particular, we suppose that $q(r)$ is given in complete factorization, i.e.,

$$q(r)=q_1(r)^{n_1}\dots q_u(r)^{n_u} \tag{90}$$

where the irreducible polynomials $q_i(r)\in\mathbb{K}[r]$ occur with multiplicity $n_i\in\mathbb{N}$. Now we proceed with step (1) but make the refined ansatz $g'(r)=q(r)\,\gamma(r)$ for some unknown polynomial $\gamma(r)$. Plugging $g'(r)$ into (77) yields

$$\frac{a_\delta(r)q(r+\delta)}{d(r+\delta)}\,\gamma(r+\delta)+\dots+\frac{a_1(r)q(r+1)}{d(r+1)}\,\gamma(r+1)+\frac{a_0(r)q(r)}{d(r)}\,\gamma(r)=c_0\,f_0(r)+\dots+c_\gamma\,f_\gamma(r). \tag{91}$$

Next, we apply our heuristic check if there exist s linearly independent solutions $\gamma(r)\in\mathbb{K}[n]$. If not, we follow the strategy as in **Improvement I** to find the maximal n_i with $1\leq i\leq u$ such that all s solutions can be recovered. Actually, we combine this technique with **Improvement I** and search simultaneously for the minimal $m_i\in\mathbb{N}$ in (87) and the maximal n_i in (90) such that $g(r)=\frac{\gamma(r)q(r)}{d(r)}$ for some polynomial $\gamma(r)\in\mathbb{K}[r]$ yields all solutions for (77) or equivalently for (91). In other words, in **Improvement I$^+$** we search simultaneously for an optimal denominator bound $d'(r)$ and try to predict extra factors of the numerator contribution, say $q'(r)$, together with the optimal degree bound b' for the unknown contribution $\gamma(r)$ in $g=\frac{q'(r)\,\gamma(r)}{d'(r)}$.

Remark 7. Restricting to the creative telescoping case of hypergeometric products, **Improvement I$^+$** finds exactly the predicted factor $q(r)\in\mathbb{K}[r]$ in (89) of Gosper's method but guarantees also that $d(r)$ has minimal degree among all possible denominator bounds and that the degree bound of the unknown polynomial solution $\gamma(r)$ is minimal. In other words, it is the optimal ansatz that yields the same efficient behavior of all variants that utilize Gosper's algorithm; in some rare instances it may even outperform the Gosper-variants if the degree and denominator bounds of Gosper's method are not optimal. $\diamondsuit$

For general linear recurrences to determine the extra contribution $q(r)$ is a heuristic (in contrast to the very special first-order recurrence (88)) and one usually has to filter out wrong factors. Surprisingly enough, the found contributions are often non-trivial and contribute substantially to a speed up of our recurrence solver.

Example 18. *Double sum on the left hand side of* (58)*:* In order to compute (61) and (62), we compute not only the optimal denominator bound (81) but also utilize the above tactic to find a non-trivial numerator contribution. More precisely, we obtain

$$q(r)=(2n+1-r)(2n+2-r)(r-1)^2(r+1)^2$$

and filter out wrong contributions yielding the correct factor

$$q'(r)=(2n+1-r)(r+1)^2$$

of the solution (62). With this modified ansatz (91) the degree bound of $\gamma(r)$ is $b' = 9$. This finally gives a linear system with 16 equations in 14 unknowns which require 0.12 MB of memory. Solving this system can be carried out in 1.9 seconds using 19MB of memory; compare row 5 in Table 2. Applying in addition **Improvement II** enables one to eliminate three redundant constraints which leads basically to the same calculation time; compare row 6 in Table 2.
Double sum (82)*:* We get the numerator contribution $q'(r) = (2n-r)(2n+1-r)(r+1)^4$ and the degree bound $b' = 9$ for the missing numerator part $\gamma(r)$; for the properties of the linear system see the 5th and 6th rows of Table 1.
Triple sum in (70)*:* We find the numerator contribution $q'(r) = (2n+1-r)(r+1)^4$ together with the degree bound $b' = 14$ for $\gamma(r)$; for the properties of the linear system see the 5th and 6th rows of of Table 4.
Triple sum (85)*:* We get the numerator factor $q'(r) = (2n-r-1)(2n-r)(2n-r+1)(r+2)^6$ and the degree bound $b' = 14$ for the unknown polynomial $\gamma(r)$; for the properties of the linear system see the 5th and 6th rows of Table 3. □

The following general remarks for our proposed solving toolbox are in place.

Remark 8. (a) If the number of solutions s is larger than one, Improvements I and I^+ search for denominators $d \in \mathbb{K}[r]$ and numerator contributions $q[r]$ that all solutions have in common. In particular, if $s = 1$, this leads usually to much better bounds.
(b) The above examples show that the derived linear system is almost optimal (see the 5th line of the tables) and **Improvement II** does not gain any further speedup (see the 6th line of the tables). Still this feature remains activated in order to deal with less optimal cases where the guess of the polynomial contributions in the numerator of the solution cannot be predicted sufficiently. We note further that for the special case $\mathbb{K} = \mathbb{Q}$ the linear system solver of Mathematica is so efficient that the gain to solve a system with the minimal number of rows is negligible. In this particular instance, **Improvement II** of `Sigma` is switched off. If more variables are contained in $\mathbb{K}$, it is activated whenever the size of the input system is big enough to gain recognizable speed-ups.
(c) With `Sigma` one can insert also manually[5] extra factors, say $p(r) \in \mathbb{K}[r]$, which is merged with the automatically guessed factor $q'(r)$, i.e., $q'(r)$ is replaced by $\mathrm{lcm}(q'(r), p(r)) \in \mathbb{K}[r]$ and the above mechanism is activated.
(d) In [vH98] an improved version of Abramov's denominator bound algorithm has been introduced that finds a sharper denominator bound but can also provide some factors of the numerator. In all examples presented in this article van Hoeji's bound is exact, i.e., it is equivalent to our result after executing **Improvement I** or **Improvement I$^+$**. Interestingly enough, our approach described as **Improvement I$^+$** succeeds in finding substantially more numerator factors as the method proposed in [vH98]. For instance, for the underlying recurrence (77) of the double sum on the left hand side of (58) the method from [vH98] yields the numerator factor $(r+1)^2$ whereas we find the larger factor $(2n+1-r)(r+1)^2$. Similarly, for the double sum (82) the method of [vH98] delivers no extra factor whereas our approach discovers the extra contribution $(2n-r)(2n+1-r)(r+1)^4$. Moreover, for the triple sum in (70) we find the numerator contribution $(2n+1-r)(r+1)^4$ whereas the method from [vH98] delivers $(r+1)^4$. For all these cases the timings to solve the derived linear system (excluding the calculation time to execute the method in [vH98]) is similar: it takes about 0.5 seconds more when one uses the bound from [vH98]. We conclude this observation by looking at the triple sum (85). Here we get the numerator factor $q'(r) = (2n-r-1)(2n-r)(2n-r+1)(r+2)^6$ whereas the bound from [vH98] is trivially 1 and thus one obtains the same linear system given by **Improvement I**. In particular, solving this system needs 14 seconds (instead of 7 seconds using the factor $q'(r)$ of our approach).
(e) Since the produced denominator bound in [vH98] is rather good (e.g., it produces the optimal bound for all the recurrences under consideration), one may opt to use directly this bound and to optimize it further with our improvements. However the underlying algorithm is much more time consuming than applying the formula (87) with all our improvements. E.g., computing (78),

[5] For the commands `GenerateRecurrence` and `SolveRecurrence` one can insert the additional option `UsePolynomialFactor→p` to pass this factor $p(r) \in \mathbb{K}[r]$ to the internal recurrence solver.

producing the factored form (87) and minimizing the multiplicities m_i takes less than 0.5 seconds in all our examples. But carrying out the method in [vH98] takes with our (maybe not optimal) implementation longer than solving the whole system. Thus we permanently switched off the approach given in [vH98] in the summation package `Sigma` and take as starting point for our simplification the formula (87).
(f) All the improvements carry over straightforwardly to the q-case, i.e., where the coefficients $a_i(r)$ in (77) are from $\mathbb{K}[q^r]$ over the rational function field $\mathbb{K} = \mathbb{K}'(q)$ and one looks for all solutions in the rational function field $\mathbb{K}(q^r)$.
(g) The described recurrence solver with all its improvements is not only the backbone for the multi-sum approach but is also an important key ingredient of the `Sigma` function `SolveRecurrence` to find efficiently all d'Alembertian solutions [AP94, PWZ96, ABPS21] over $\Pi\Sigma$-fields. ♢

8. Conclusion

We presented a fast method to compute linear recurrences for hypergeometric double sums that is also suitable for multiple sums. To guarantee the success of this method, the algorithmic theory of contiguous relations has been exploited. In addition new ideas have been presented to find rational solutions of parameterized recurrences efficiently. All the algorithmic ideas of our summation method and also the improvements of the recurrence solving extend in a straightforward fashion to the q-hypergeometric case and are available within `Sigma`.

References

[ABPS21] S.A. Abramov, M. Bronstein, M. Petkovšek, and C. Schneider. On rational and hypergeometric solutions of linear ordinary difference equations in $\Pi\Sigma^*$-field extensions. *J. Symb. Comput.*, 107:23–66, 2021. arXiv:2005.04944 [cs.SC].

[Abr89a] S.A. Abramov. Problems in computer algebra that are connected with a search for polynomial solutions of linear differential and difference equations. *Moscow Univ. Comput. Math. Cybernet.*, 3:63–68, 1989.

[Abr89b] S.A. Abramov. Rational solutions of linear differential and difference equations with polynomial coefficients. *U.S.S.R. Comput. Math. Math. Phys.*, 29(6):7–12, 1989.

[Abr95] S.A. Abramov. Rational solutions of linear difference and q-difference equations with polynomial coefficients. In T. Levelt, editor, *Proc. ISSAC'95*, pages 285–289. ACM Press, New York, 1995.

[ABRS12] J. Ablinger, J. Blümlein, M. Round, and C. Schneider. Advanced computer algebra algorithms for the expansion of Feynman integrals. In J. Bluemlein, S. Moch, and T. Riemann, editors, *Loops and Legs in Quantum Field Theory 2012*, volume 50 of *PoS(LL2012)*, pages 1–14, 2012.

[AP93] G.E. Andrews and P. Paule. Some questions concerning computer-generated proofs of a binomial double-sum identity. *J. Symbolic Comput.*, 16:147–153, 1993.

[AP94] S.A. Abramov and M. Petkovšek. D'Alembertian solutions of linear differential and difference equations. In J. von zur Gathen, editor, *Proc. ISSAC'94*, pages 169–174. ACM Press, Baltimore, 1994.

[APS05] G.E. Andrews, P. Paule, and C. Schneider. Plane partitions VI: Stembridge's TSPP Theorem. *Advances in Applied Math. Special Issue Dedicated to Dr. David P. Robbins. Edited by D. Bressoud*, 34(4):709–739, 2005. Preliminary version online.

[BBF+14] A. Behring, J. Blümlein, A. De Freitas, T. Pfoh, C. Raab, M. Round, J. Ablinger, A. Hasselhuhn, C. Schneider, F. Wissbrock, and A. von Manteuffel. New results on the 3-loop heavy flavor corrections in deep-inelastic scattering. In *Proc. of RADCOR*, PoS(RADCOR 2013)058, pages 1–21, 2014.

[BRS18] J. Blümlein, M. Round, and C. Schneider. Refined holonomic summation algorithms in particle physics. In E. Zima and C. Schneider, editors, *Advances in Computer Algebra. WWCA 2016.*, volume 226 of *Springer Proceedings in Mathematics & Statistics*, pages 51–91. Springer, 2018. arXiv:1706.03677 [cs.SC].

[Chy00] F. Chyzak. An extension of Zeilberger's fast algorithm to general holonomic functions. *Discrete Math.*, 217:115–134, 2000.

[CPS08] W. Y. Chen, P. Paule, and H.L. Saad. Converging to Gosper's algorithm. *Adv. in Appl. Math.*, 41(3):351–364, 2008.

[Gau13] C.F. Gauß. Disquisitiones generales circa seriem infinitam $1 + \frac{\alpha\beta}{1\cdot\gamma}x + \dots$, pars prior. *Commentationes societatis regiae scientarum Gottingensis recentiores 2 (classis mathematicae)*,`https://gdz.sub.uni-goettingen.de/id/PPN235999628`, 1813.

[Gos78] R.W. Gosper. Decision procedures for indefinite hypergeometric summation. *Proc. Nat. Acad. Sci. U.S.A.*, 75:40–42, 1978.

[Kar81] M. Karr. Summation in finite terms. *J. ACM*, 28:305–350, 1981.

[KR04] C. Krattenthaler and T. Rivoal. Hypergéométrie et fonction zêta de Riemann. *Preprint*, 2004.

[MS18] R. Lyons, P. Paule, and A. Riese. A computer proof of a series evaluation in terms of harmonic numbers. Appl. Algebra Engrg. Comm. Comput., 13:327–333, 2002.

[MS18] J. Middeke and C. Schneider. Denominator bounds for systems of recurrence equations using ΠΣ-extensions. In C. Schneider and E. Zima, editors, *Advances in Computer Algebra. WWCA 2016.*, volume 226 of *Springer Proceedings in Mathematics & Statistics*, pages 149–173. Springer, 2018. arXiv:1705.00280 [cs.SC].

[Pau95] P. Paule. Greatest factorial factorization and symbolic summation. *J. Symbolic Comput.*, 20(3):235–268, 1995.

[Pau21] P. Paule. Contiguous relations and creative telescoping. In J. Blümlein and C. Schneider, editors, *Anti-Differentiation and the Calculation of Feynman Amplitudes*, Texts and Monographs in Symbolic Computation, pages 335–394. Springer, 2021.

[Pet92] M. Petkovšek. Hypergeometric solutions of linear recurrences with polynomial coefficients. *J. Symbolic Comput.*, 14(2-3):243–264, 1992.

[PS95] P. Paule and M. Schorn. A Mathematica version of Zeilberger's algorithm for proving binomial coefficient identities. *J. Symbolic Comput.*, 20(5-6):673–698, 1995.

[PS03] P. Paule and C. Schneider. Computer proofs of a new family of harmonic number identities. *Adv. in Appl. Math.*, 31(2):359–378, 2003.

[PW00] M. Petkovšek and C. Weixlbaumer. A comparison of degree polynomials. `http://www.fmf.uni-lj.si/~petkovsek/`, 2000. Note.

[PWZ96] M. Petkovšek, H.S. Wilf, and D. Zeilberger. $A = B$. A.K. Peters, Wellesley, MA, 1996.

[RZ04] A. Riese and B. Zimmermann. Randomization speeds up hypergeometric summation. *Preprint*, 2004.

[SA95] M. Petkovšek S.A. Abramov, M. Bronstein. On polynomial solutions of linear operator equations. In T. Levelt, editor, *Proc. ISSAC'95*, pages 290–296. ACM Press, New York, 1995.

[Sch01] C. Schneider. *Symbolic Summation in Difference Fields*. PhD thesis, RISC-Linz, J. Kepler University, Linz, May 2001.

[Sch05] C. Schneider. A new Sigma approach to multi-summation. *Advances in Applied Math. Special Issue Dedicated to Dr. David P. Robbins. Edited by D. Bressoud*, 34(4):740–767, 2005. Preliminary version online.

[Sch07] C. Schneider. Symbolic Summation Assists Combinatorics. *Sem. Lothar. Combin.*, 56:1–36, 2007. Article B56b.

[Sch16] C. Schneider. A difference ring theory for symbolic summation. *J. Symb. Comput.*, 72:82–127, 2016.

[Str94] V. Strehl. Binomial identities — combinatorial and algorithmical aspects. *Discrete Math.*, 136:309–346, 1994.

[SZ21] C. Schneider and W. Zudilin. A case study for $\zeta(4)$. In Alin Bostan and Kilian Raschel, editors, *Transcendence in Algebra, Combinatorics, Geometry and Number Theory. TRANS 2019*, volume 373 of *Proceedings in Mathematics & Statistics*, pages 421–435. Springer, 2021. arXiv:2004.08158 [math.NT].

[vH98] M. van Hoeij. Rational solutions of linear difference equations. In O. Gloor, editor, *Proc. ISSAC'98*, pages 120–123. ACM Press, 1998.

[Weg97] K. Wegschaider. Computer generated proofs of binomial multi-sum identities. Diploma thesis, RISC Linz, Johannes Kepler University, May 1997.

[Zei90] D. Zeilberger. A holonomic systems approach to special functions identities. *J. Comput. Appl. Math.*, 32:321–368, 1990.

(Peter Paule) Research Institute for Symbolic Computation (RISC), Johannes Kepler University Linz, Altenbergerstr. 69, A–4040 Linz, Austria, and Center for Applied Mathematics, Tianjin University, Tianjin 300072, P.R. China

Email address: `Peter.Paule@risc.jku.at`

(Carsten Schneider) Research Institute for Symbolic Computation (RISC), Johannes Kepler University Linz, Altenbergerstr. 69, A–4040 Linz, Austria

Email address: `Carsten.Schneider@risc.jku.at`