A NEW SIGMA APPROACH TO MULTI-SUMMATION

CARSTEN SCHNEIDER

Dedicated to the memory of David Robbins

Abstract. We present a general algorithmic framework that allows not only to deal with summation problems over summands being rational expressions in indefinite nested sums and products (Karr 1981), but also over ∂-finite and holonomic summand expressions that are given by a linear recurrence. This approach implies new computer algebra tools implemented in Sigma to solve multi-summation problems efficiently. For instance, the extended Sigma package has been applied successively to provide a computer-assisted proof of Stembridge's TSPP theorem.

1. Introduction

Gosper's indefinite summation algorithm [14] and Zeilberger's method of creative telescoping [37] for hypergeometric terms can be seen as a major breakthrough in symbolic summation [23]. These ideas have been generalized in various directions.

Based on Karr's difference field theory of $\Pi\Sigma$-fields [15, 16] and ideas from [6] algorithms have been developed [26, 28, 27, 31, 30] and implemented in the summation package Sigma [25, 29] that not only can deal with telescoping and creative telescoping in (q–)hypergeometric terms, as shown in [32], but more generally in so-called $\Pi\Sigma$-fields. $\Pi\Sigma$-fields allow us to describe rational expressions involving indefinite nested sums and products. The wide applicability of this approach is illustrated for instance in [20, 10, 11, 29].

Another general approach is [8] that extends hypergeometric to general holonomic creative telescoping and, in particular, to ∂-finite functions. A crucial observation is that the difference field machinery [26] can be embedded in this general approach [9, 8] based on [38]. More precisely, we are able to develop a common framework in Sigma in which both, Karr's summation theory [15] and ideas of the ∂-finite algorithms [9, 8] are combined. This combined approach enables one to treat indefinite and definite summation problems that could not be treated so far. In particular, by restricting the input class of [8], we were able to simplify and streamline ideas in [8] which results in algorithms which are free of any Gröbner bases computations. Another new feature concerns the fact that no uncoupling algorithm for systems of difference equations is needed. For further remarks relating to Chyzak's approach see below of Example 11.

All these ideas allow us to derive a new computer assisted proof [5] of Stembridge's TSPP Theorem [33]. These highly non-trivial applications, together with other examples, will illustrate our results throughout this paper.

The general structure is as follows. At the end of this section we introduce the paradigms on which all our summation algorithms are based. In Section 2 we supplement the discussion of the key problem (*GPTRT*) by various illustrative examples. In Section 3 we present the algorithms that allow to solve our problem in general difference fields. In Section 4 we apply these techniques by showing how a huge class of multi-sum identities can be proven. In Section 5 we describe the usage of our extended Mathematica package Sigma which contains implementations of all the algorithms described.

Supported by the Austrian Academy of Sciences, the SFB-grant F1305 of the Austrian FWF and by grant P16613-N12 of the Austrian FWF.

Subsequently $\mathbb{N}$ denotes the non-negative integers and $\boldsymbol{n}$ denotes a vector of variables $(n_1, \dots, n_r)$ ranging over the integers. All our summation algorithms are based on the paradigm of

Generalized Parameterized Telescoping (*GPT*).

- **Given** $f_i(\boldsymbol{n}, k)$ for $0 \leq i \leq d$;
- **find** $c_0(\boldsymbol{n}), \dots, c_d(\boldsymbol{n})$, free of k and not all zero, and $g(\boldsymbol{n}, k)$ such that

$$g(\boldsymbol{n}, k+1) - g(\boldsymbol{n}, k) = c_0(\boldsymbol{n})\, f_0(\boldsymbol{n}, k) + \dots + c_d(\boldsymbol{n})\, f_d(\boldsymbol{n}, k) \tag{1}$$

holds for all $\boldsymbol{n}$ and k in a certain range.

Summing (1) over all k from a to b gives

$$c_0(\boldsymbol{n}) \sum_{k=a}^{b} f_0(\boldsymbol{n}, k) + \dots + c_d(\boldsymbol{n}) \sum_{k=a}^{b} f_d(\boldsymbol{n}, k) = g(\boldsymbol{n}, b+1) - g(\boldsymbol{n}, a), \quad b - a \geq 0 \tag{2}$$

which specializes to indefinite and definite summation as follows. For the special case $d = 0$ one obtains a representation for the indefinite sum, namely

$$\sum_{k=a}^{b} f_0(\boldsymbol{n}, k) = \frac{g(\boldsymbol{n}, b+1) - g(\boldsymbol{n}, a)}{c_0(\boldsymbol{n})}. \tag{3}$$

In order to arrive at definite summation, one specializes $f_i(\boldsymbol{n}, k) := f(\boldsymbol{n} + \boldsymbol{\gamma_i}, k)$ for a given $f(\boldsymbol{n}, k)$ and where the $\boldsymbol{\gamma_i} \in \mathbb{N}^r$ specify the non-negative integer shifts. This reduces *GPT* to

Specialized Parameterized Telescoping (*SPT*).

- **Given** $f(\boldsymbol{n}, k)$ and $\{\boldsymbol{\gamma_0}, \dots, \boldsymbol{\gamma_d}\} \subseteq \mathbb{N}^r$;
- **find** $c_0(\boldsymbol{n}), \dots, c_d(\boldsymbol{n})$, free of k and not all zero, and $g(\boldsymbol{n}, k)$ such that

$$g(\boldsymbol{n}, k+1) - g(\boldsymbol{n}, k) = c_0(\boldsymbol{n})\, f(\boldsymbol{n} + \boldsymbol{\gamma_0}, k) + \dots + c_d(\boldsymbol{n})\, f(\boldsymbol{n} + \boldsymbol{\gamma_d}, k) \tag{4}$$

holds for all $\boldsymbol{n}$ and k in a certain range.

We say that α is integer linear in $\boldsymbol{n}$, if $\alpha = \sum_{i=1}^{r} \gamma_i\, n_i + \gamma_0$ for integers γ_i. Defining

$$S(\boldsymbol{n}) = \sum_{k=\alpha}^{\beta} f(\boldsymbol{n}, k), \quad \alpha \text{ and } \beta \text{ integer linear in } \boldsymbol{n}, \tag{5}$$

and summing (4) over all k from a sufficiently large interval, one obtains a not necessarily homogeneous recurrence relation

$$c_0(\boldsymbol{n}) S(\boldsymbol{n} + \boldsymbol{\gamma_0}) + \dots + c_d(\boldsymbol{n}) S(\boldsymbol{n} + \boldsymbol{\gamma_d}) = h(\boldsymbol{n}). \tag{6}$$

Observe that all methods based on the *SPT*-paradigm, like [37, 19, 8], not only deliver recurrence relations of the type (6) but provide all the information needed to verify the computed result independently of the steps of the algorithm. Namely, given the solutions $c_i(\boldsymbol{n})$ and $g(\boldsymbol{n}, k)$ for problem *SPT*, one verifies the summand equation (4). This implies the correctness of the recurrence (6) itself.

2. The basic mechanism

We are interested in the following summation problem. **Given** $S(\boldsymbol{n}) = \sum_{k=\alpha}^{\beta} f(\boldsymbol{n}, k)$ as in (5) where for the summand $f(\boldsymbol{n}, k)$ the following properties hold: For a fixed non-negative integer s,

$$f(\boldsymbol{n}, k) = h_0(\boldsymbol{n}, k) T(\boldsymbol{n}, k) + \dots + h_s(\boldsymbol{n}, k) T(\boldsymbol{n}, k+s) + h_{s+1}(\boldsymbol{n}, k); \tag{7}$$

in addition, $T(\boldsymbol{n}, k)$ satisfies a recurrence of order $s+1$ of the form

$$T(\boldsymbol{n}, k+s+1) = a_0(\boldsymbol{n}, k) T(\boldsymbol{n}, k) + \dots a_s(\boldsymbol{n}, k) T(\boldsymbol{n}, k+s) + a_{s+1}(\boldsymbol{n}, k) \tag{8}$$

and recurrences of the form

$$T(\boldsymbol{n} + \boldsymbol{e_i}, k) = b_0^{(i)}(\boldsymbol{n}, k) T(\boldsymbol{n}, k) + \dots + b_s^{(i)}(\boldsymbol{n}, k) T(\boldsymbol{n}, k+s) + b_{s+1}^{(i)}(\boldsymbol{n}, k) \tag{9}$$

for any unit vector $\boldsymbol{e_i}$. **Find** a recurrence of the type (6) with given $\boldsymbol{\gamma_i}$. Moreover, **deliver** proof certificates that allow us to verify the derived recurrence (6).

Subsequently we try to tackle this problem by developing tools that allow us to solve

SPT with a Recurrence System (*SPTRS*).

• **Given** $\{\boldsymbol{\gamma_0},\dots,\boldsymbol{\gamma_d}\}\subseteq\mathbb{N}^r$ and $f(\boldsymbol{n},k)$ as in (7) for a fixed non-negative integer s where $T(\boldsymbol{n},k)$ satisfies a recurrence of the form (8) and recurrences of the form (9) for any unit vector $\boldsymbol{e_i}$;

• **find** $c_0(\boldsymbol{n}),\dots,c_d(\boldsymbol{n})$, free of k and not all zero, and $g(\boldsymbol{n},k)$ of the form

$$g(\boldsymbol{n},k)=g_0(\boldsymbol{n},k)T(\boldsymbol{n},k)+\dots+g_s(\boldsymbol{n},k)T(\boldsymbol{n},k+s)+g_{s+1}(\boldsymbol{n},k)$$

such that (4) holds for all $\boldsymbol{n}$ and k in a certain range.

Observe that in our specification of problem *SPTRS* the term $T(\boldsymbol{n},k)$ stands for any sequence that satisfies (8) and (9). Therefore, solving a concrete problem of *SPTRS* actually means to provide solutions for a whole class of sequences that is represented by $f(\boldsymbol{n},k)$ in terms of $T(\boldsymbol{n},k)$.

Example 1. Our methods deliver a direct proof of the double sum identity

$$\sum_{k=0}^{n}\underbrace{\sum_{s=0}^{n}(-1)^{n+k+s}\binom{n}{k}\binom{n}{s}\binom{n+k}{k}\binom{n+s}{s}\binom{2n-s-k}{n}}_{=T(n,k)}=\sum_{k=0}^{n}\binom{n}{k}^4 \tag{10}$$

from [23, page 33]. Namely, with the summation package Sigma, see Subsection 5.2, or any implementation of Zeilberger's algorithm [37], like [22], one can derive the recurrence

$$\begin{aligned}T(n,k+2)=&\frac{(n-k)^3(1+k+n)(2+k+n)}{(1+k)^2(2+k)^2(k-3n)}T(n,k)\\&+\frac{(1+k)^2(2+k+n)(k+2k^2-3n-6kn+3n^2)}{(1+k)^2(2+k)^2(k-3n)}T(n,k+1)\end{aligned} \tag{11}$$

for the inner sum $T(n,k)$ on the left hand side of (10). Similarly, with Sigma, see Subsection 5.2, or an extended version of Zeilberger's algorithm [18] one can compute the recurrence

$$\begin{aligned}T(n+1,k)=&-(1+k+n)(-5k+12k^2-10k^3+3k^4+3n-32kn+42k^2n-16k^3n\\&+15n^2-57kn^2+33k^2n^2+21n^3-30kn^3+9n^4)\Big/((1-k+n)^3(1+n)^2)T(n,k)\\&+\frac{(1+k)^2(-1+k-3n)(6-8k+3k^2+12n-8kn+6n^2)}{(1-k+n)^3(1+n)^2}T(n,k+1).\end{aligned} \tag{12}$$

Note that all these approaches [37, 18, 26] are based on the *SPT*-paradigm and therefore allow us to verify independently the correctness of the recurrence relations (11) and (12) for $0\le k\le n$. Taking those recurrences as input, our algorithm computes $c_0(n)=-4(1+n)(3+4n)(5+4n)$, $c_1(n)=2(3+2n)(7+9n+3n^2)$, $c_2(n)=(2+n)^3$ and

$$g(n,k)=g_0(n,k)T(n,k)+g_1(n,k)T(n,k+1)+g_2(n,k)T(n,k+2) \tag{13}$$

for some rational functions $g_i(n,k)$ in n and k such that

$$g(n,k+1)-g(n,k)=c_0(n)T(n,k)+c_1(n)T(n+1,k)+c_2(n)T(n+2,k) \tag{14}$$

holds for all $0\le k\le n$. The expressions $g_i(n,k)$ can be found explicitly in Subsection 5.2. Finally, summing equation (14) over the summation range gives the recurrence

$$-4(1+n)(3+4n)(5+4n)S(n)-2(3+2n)(7+9n+3n^2)S(1+n)+(2+n)^3S(2+n)=0 \tag{15}$$

for the double sum on the left hand side of (10). Applying Zeilberger's algorithm in its standard form returns the same recurrence (15) for the right hand side of (10). Checking that both sides are equal for $n=0,1$ proves the identity.

Verification of (14). Observe that so far our proof relies on the fact that the computed $c_i(n)$ and $g(n,k)$ satisfy (14) for all $0 \leq k \leq n$. For the verification of this fact we proceed as follows. First note that $f_1(n,k) := T(n+1,k)$ can be expressed as

$$f_1(n,k) = h_0^{(1)}(n,k)T(n,k) + h_1^{(1)}(n,k)T(n,k+1) + h_2^{(1)}(n,k)T(n,k+2) \tag{16}$$

where the $h_i^{(1)}(n,k)$ denote the coefficients in (12). Similarly, the expression $f_2(n,k) := T(n+2,k)$ can be expressed by a linear combination in $T(n+1,k)$, $T(n+1,k+1)$ and $T(n+2,k+2)$ which itself can be expressed by a linear combination in terms of $T(n,k)$, $T(n,k+1)$, $T(n,k+2)$ by using the "rewrite rules" (11) and (12). In other words, we can write $f_2(n,k)$ in the form

$$f_2(n,k) = h_0^{(2)}(n,k)T(n,k) + h_1^{(2)}(n,k)T(n,k+1) + h_2^{(2)}(n,k)T(n,k+2) \tag{17}$$

for some rational functions $h_i^{(2)}(n,k)$ in n and k. Moreover, the expression $g'(n,k) := g(n,k+1)$ can be rewritten to the expression

$$g'(n,k) = g'_0(n,k)T(n,k) + g'_1(n,k)T(n,k+1) + g'_2(n,k)T(n,k+2)$$

by using (11). Hence, after setting $f_0(n,k) := T(n,k)$, (14) holds for all $0 \leq k \leq n$ if and only if

$$g'(n,k) - g(n,k) - (c_0(n)f_0(n,k)(n,k) + c_1(n)f_1(n,k) + c_2 f_2(n,k)) = 0$$

holds in the same range. Finally, we are able to verify this last equation by elementary polynomial arithmetic. □

THE KEY PROBLEM: The crucial idea in our approach is that problem *SPTRS* can be reduced to a simpler problem. Namely, as illustrated in the previous example, any expression $f(\boldsymbol{n}+\boldsymbol{\gamma_i},k)$ given by (7) and $\boldsymbol{\gamma_i} \in \mathbb{N}^r$ can be equivalently written in the form (18) by using the recurrence relations (8) and (9). Hence, in order to solve problem *SPTRS*, it suffices to develop methods that can solve the problem

GPT over a Recurrence Term (*GPTRT*).

- **Given** $f_i(\boldsymbol{n},k)$ for $1 \leq i \leq d$ with

$$f_i(\boldsymbol{n},k) := h_0^{(i)}(\boldsymbol{n},k)T(\boldsymbol{n},k) + \cdots + h_s^{(i)}(\boldsymbol{n},k)T(\boldsymbol{n},k+s) + h_{s+1}^{(i)}(\boldsymbol{n},k), \tag{18}$$

where $T(\boldsymbol{n},k)$ satisfies a recurrence of order $s+1$ of the form (8);

- **find** $c_i(\boldsymbol{n})$ for $1 \leq i \leq d$ and $g(\boldsymbol{n},k)$ of the type

$$g(\boldsymbol{n},k) = g_0(\boldsymbol{n},k)T(\boldsymbol{n},k) + \cdots + g_s(\boldsymbol{n},k)T(\boldsymbol{n},k+s) + g_{s+1}(\boldsymbol{n},k) \tag{19}$$

such that (1) holds.

Summarizing, any solution of *SPTRS* is also a solution of *GPTRT*, and vice versa — under the assumption that the recurrence relations (8) and (9) are valid in the required range. In this context it is important to mention that the way in which we will solve *GPTRT*, see problem *GPTHO* (page 7), gives **always** a recipe to verify (19). Namely, as in Example 1, represent $g'(\boldsymbol{n},k) := g(\boldsymbol{n},k+1)$ in the form

$$g'(\boldsymbol{n},k) = g'_1(\boldsymbol{n},k)T(\boldsymbol{n},k) + \cdots + g'_s(\boldsymbol{n},k)T(\boldsymbol{n},k+s) + g'_{s+1}(\boldsymbol{n},k)$$

by using (8); then verify by coefficient comparison w.r.t. the $T(\boldsymbol{n},k+i)$ that the expression

$$g'(\boldsymbol{n},k) - g(\boldsymbol{n},k) - \big[c_0(\boldsymbol{n},k)f_0(\boldsymbol{n},k) + \cdots + c_d(\boldsymbol{n},k)f_d(\boldsymbol{n},k)\big]$$

collapses to 0.

Besides definite summation (*SPTRS*) also indefinite summation is covered in *GPTRT*:

Example 2 (TSPP)**.** Within our computer assisted proof [5] of the TSPP-Theorem [33] there arises the following problem in Lemma 4. Given the triple sum $S(n) = \sum_{k=0}^{2n} T(n,k)$ with

$$T(n,k) = \sum_{s=0}^{\lfloor\frac{2n-k}{2}\rfloor} \left(\binom{n-s-1}{2n-2s-k} + \binom{n-s}{2n-2s-k}\right) \frac{(-1)^{s+k}}{2n\,4^s} \sum_{r=0}^{s} \frac{(n-r)(n)_r(-3n-1)_r}{r!(\frac{1}{2}-2n)_r}, \tag{20}$$

eliminate the outermost summation quantifier of $S(n)$. To accomplish this task, we first compute with Sigma a recurrence for $T(n,k)$ with shifts in k. Namely, by solving the corresponding problem SPT in the $\Pi\Sigma$-field setting we obtain the recurrence[1]

$$T(n,k+3)=a_0(n,k)T(n,k)+a_1(n,k)T(n,k+1)+a_2(n,k)T(n,k+2),\quad (\forall n,k\geq 0)\qquad (21)$$

where the coefficients $a_i(n,k)\in\mathbb{Q}(n,k)$ can be found in Subsection 5.1; see `In[2]`. In the next step we solve the $GPTRT$ problem for the case $d=0$ and $f_0(n,k):=T(n,k)$, i.e., we try to find a $g(n,k)=g_0(n,k)\,T(n,k)+g_1(n,k)\,T(n,k+1)+g_2(n,k)\,T(n,k+2)$ with

$$g(n,k+1)-g(n,k)=T(n,k).$$

Sigma returns

$$\begin{aligned}g(n,k)=-2(k(1+k)(2+k)&-(3+k)n-2(3+k)n^2)T(n,k)\\ &+(3k(1+k)(2+k)-2(1+2k)n-4(1+2k)n^2)T(n,k+1)\\ &-k(1+k-2n)(2+k+2n)T(n,k+2)\Big/(2(1+k)n(1+2n))\end{aligned}\qquad (22)$$

which allows us to verify that

$$S(n)=\frac{2n+5}{2n+1}T(n,2n+1)-\frac{2}{2n+1}T(n,2n+2)-(3T(n,0)-T(n,1))\quad (n\geq 1);\qquad (23)$$

see [5, Subsection 5.2]. Evaluation of $T(n,k)$ at its bounds gives $\sum_{k=0}^{2n}T(n,k)=-T(n,0)$. This proves Lemma 4 in [5]. □

So far we have considered examples of $GPTRT$ only for the rational case, i.e., where the $a_i(\boldsymbol{n},k)$ and $h_j^{(i)}(\boldsymbol{n},k)$ are given in $\mathbb{Q}(\boldsymbol{n},k)$, and $c_i(\boldsymbol{n})$ and $g_i(\boldsymbol{n},k)$ are searched in $\mathbb{Q}(\boldsymbol{n})$ and $\mathbb{Q}(\boldsymbol{n},k)$ respectively. More generally, we will be able to solve problem $GPTRT$ in the algebraic domain of $\Pi\Sigma$-fields [15], see Section 3.3, which means that $a_i(\boldsymbol{n},k)$, $h_j^{(i)}(\boldsymbol{n},k)$, $c_i(\boldsymbol{n})$ and $g_i(\boldsymbol{n},k)$ may be represented by rational expressions involving indefinite nested sums and products.

Example 3. Consider a sequence $T(k)$ for $k\geq 1$ that satisfies the recurrence relation

$$T(k+2)=\frac{-3(3+2k+H_k(2+3k+k^2))}{H_k(1+k)(2+k)}T(k)-\frac{4(3+2k+H_k(2+3k+k^2))}{(2+k)(1+H_k(1+k))}T(k+1)$$

where H_k denotes the harmonic numbers $\sum_{i=1}^{k}\frac{1}{i}$. In this example the goal is to find a recurrence for the sum expression $S(n)=\sum_{k=1}^{n}\binom{n}{k}T(k)$. To accomplish this task, we compute for problem GPTRT with $d=2$ and $f_i(n,k)=f(n+i,k)=\prod_{j=1}^{i}\frac{n+j}{n+j-k}\binom{n}{k}T(k)$ the solution $c_0(n)=4n^2(1+n)^2$, $c_1(n)=2n^2(1+n)(3+2n)$, $c_2(n)=n^2(1+n)(2+n)$, and $g(n,k)=g_0(n,k)\,T(k)+g_1(n,k)\,T(k+1)$ where

$$\begin{aligned}g_1(n,k)=-(1+k)(2k^2(1+n)^2&+n(2+8n+9n^2+3n^3)-k(2+8n+13n^2+6n^3)\\ &+kn(1+n)(-2-6n-3n^2+2k(1+n))H_k)\binom{n}{k}\Big/\big((-1+k-n)(1+(1+k)H_k)\big)\end{aligned}$$

and $g_0(n,k)=\frac{-3(3+2k+H_k(2+3k+k^2))}{H_k(1+k)(2+k)}\,g_1(n,k+1)-\sum_{i=0}^{2}c_i(n)\binom{n+i}{k}$. Finally, with these ingredients one can derive (together with a correctness proof as in Example 1) the recurrence

$$\begin{aligned}12n(1+n)^2S(n)+6n(1+n)(3+2n)S(1+n)&+3n(1+n)(2+n)S(2+n)\\ &=3(6+22n+13n^2)T(1)+2(2+7n+4n^2)T(2).\end{aligned}$$

Remark. Given this information, one can discover the identity

$$S(n)=\frac{27\,T(1)+6\,T(2)}{18n}+\frac{1}{18}\big(3T(1)+2T(2)\big)(-2)^n\Big[H_n-\sum_{i=1}^{n}\frac{1}{i(-2)^i}\Big],\quad n\geq 1\qquad (24)$$

by using the tool box of Sigma described in [26, 29]. □

[1] For the explicit creative telescoping solution and a rigorous correctness proof we refer to [4, Remark 6].

3. A Method for the GPTRT Problem in Difference Fields

As motivated in the previous section, a huge class of summation problems (*SPTRS*) can be handled if one knows how to solve *GPTRT*. In this section we will present algorithms working in general difference fields that solve problem *GPTRT* under the assumption that one can solve parameterized linear difference equations. This will result in a new summation algorithm in the difference field setting of $\Pi\Sigma$-fields by applying algorithms developed in [6, 28, 27, 31].

3.1. **Translation to difference fields.** In a first step we reformulate problem *GPTRT* by introducing the shift operator S_k with respect to k and denoting $x_i := T(\boldsymbol{n}, k+i)$ for $0 \leq i \leq s$. Then we have $S_k\, x_i = x_{i+1}$ for $1 \leq i < s$ and (8) reads as

$$S_k\, x_s = a_0(\boldsymbol{n}, k)x_0 + \cdots + a_s(\boldsymbol{n}, k)x_s + a_{s+1}(\boldsymbol{n}, k). \tag{25}$$

Moreover, (18) and (19) can be expressed in the form

$$f_i(\boldsymbol{n}, k) = h_0^{(i)}(\boldsymbol{n}, k)x_0 + \cdots + h_s^{(i)}(\boldsymbol{n}, k)x_s + h_{s+1}^{(i)}(\boldsymbol{n}, k), \tag{26}$$

$$g(\boldsymbol{n}, k) = g_0(\boldsymbol{n}, k)x_0 + \cdots + g_s(\boldsymbol{n}, k)x_s + g_{s+1}(\boldsymbol{n}, k). \tag{27}$$

Now the essential step consists in representing the sequences in (25), (26), (27) in terms of a field $\mathbb{F}$ where the shift operator S_k acting on those sequences can be described by a field automorphism $\sigma : \mathbb{F} \to \mathbb{F}$. More precisely, we shall describe our sequences in difference fields $(\mathbb{F}, \sigma)$, i.e., a field[2] $\mathbb{F}$ together with a field automorphism σ. The constant field of $(\mathbb{F}, \sigma)$ is defined as $\mathrm{const}_\sigma\mathbb{F} = \{c \in \mathbb{F} \,|\, \sigma(c) = c\}$.

Example 4 (TSPP cont.)**.** For Example 2 this translation can be carried out as follows. Consider the field of rational functions $\mathbb{F} := \mathbb{Q}(n)(k)(x_0, x_1, x_2)$ and the field automorphism $\sigma : \mathbb{F} \to \mathbb{F}$ with $\sigma(p) = p$ for all $p \in \mathbb{Q}(n)$, $\sigma(k) = k + 1$, $\sigma(x_0) = x_1$, $\sigma(x_1) = x_2$ and $\sigma(x_2) = a_0x_0 + a_1x_1 + a_2x_2$ where the a_i are taken from (21). Then problem *GPTRT* can be stated in the difference field $(\mathbb{F}, \sigma)$ as follows. Find $g = g_0\, x_0 + g_1\, x_1 + g_2\, x_2$ with $g_i \in \mathbb{Q}(n)(k)$ such that

$$\sigma(g) - g = x_0. \tag{28}$$

With our algorithm, given below, we are able to compute the solution

$$\begin{aligned} g(n, k) = -2(k(1 + k)(2 + k) - (3 + k)n - 2(3 + k)n^2)x_0 + (3k(1 + k)(2 + k) \\ - 2(1 + 2k)n - 4(1 + 2k)n^2)x_1 - k(1 + k - 2n)(2 + k + 2n)x_2 \Big/ (2(1 + k)n(1 + 2n)) \end{aligned}$$

Reinterpreting this result as a sequence $g(n, k)$ gives the solution (22). □

Example 5. For Example 1 we can construct the following difference field $(\mathbb{F}, \sigma)$. Take the field of rational functions $\mathbb{F} := \mathbb{Q}(n)(k)(x_0, x_1)$ where the automorphism σ is defined as $\sigma(p) = p$ for all $p \in \mathbb{Q}(n)$, $\sigma(k) = k + 1$, $\sigma(x_0) = x_1$ and

$$\sigma(x_1) = \frac{(n - k)^3(1 + k + n)(2 + k + n)}{(1 + k)^2(2 + k)^2(k - 3n)}x_0 + \frac{(1 + k)^2(2 + k + n)(k + 2k^2 - 3n - 6kn + 3n^2)}{(1 + k)^2(2 + k)^2(k - 3n)}x_1.$$

Observe that $\mathbb{Q}(n)$ is the constant field of $(\mathbb{F}, \sigma)$. In this algebraic domain $\mathbb{F}$ we define

$$f_0 = x_0, \qquad f_1 = h^{(1)}x_0 + h_1^{(1)}x_1 + h_2^{(1)}x_2, \qquad f_2 = h^{(2)}x_0 + h_1^{(2)}x_1 + h_2^{(2)}x_2$$

where the coefficients $h_j^{(i)}$ are taken from (16) and (17). Then with our algorithms, see below, we find constants $c_i \in \mathbb{Q}(n)$ and a $g = g_0\, x_0 + g_1\, x_1$ with $g_i \in \mathbb{Q}(n)(k)$ such that

$$\sigma(g) - g = c_0\, f_0 + c_1\, f_1 + c_2\, f_2. \tag{29}$$

Reinterpreting c_i and g as sequences gives the solutions $c_i(n)$ and $g(n, k)$ from (13). □

[2] Throughout this paper all fields will have characteristic 0.

Example 6. For Example 3 consider first the field of rational functions $\mathbb{F} := \mathbb{Q}(n)(k)(B)(H)$, and define the difference field $(\mathbb{F}, \sigma)$ with constant field $\mathbb{Q}(n)$ where $\sigma(k) = k+1$, $\sigma(B) = \frac{n-k}{k+1} B$ and $\sigma(H) = H + \frac{1}{k+1}$. Note that the shift $S_k \binom{n}{k} = \frac{n-k}{k+1}\binom{n}{k}$ and $S_k H_k = H_k + \frac{1}{k+1}$ is reflected by the action of σ on H and B. Now consider the rational function field extension $\mathbb{E} = \mathbb{F}(x_0, x_1, x_2)$ of $\mathbb{F}$ and extend σ to a field automorphism $\sigma : \mathbb{E} \to \mathbb{E}$ which acts on $\mathbb{F}$ as in $(\mathbb{F}, \sigma)$ and where we have $\sigma(x_0) = x_1$, $\sigma(x_1) = x_2$ and $\sigma(x_2) = a_0 x_0 + a_1 x_1$ with

$$a_0 = \frac{-3(3+2k+H(2+3k+k^2))}{H(1+k)(2+k)} \qquad \text{and} \qquad a_1 \frac{4(3+2k+H(2+3k+k^2))}{(2+k)(1+H(1+k))}. \tag{30}$$

In this difference field extension $(\mathbb{E}, \sigma)$ of $(\mathbb{F}, \sigma)$ we define $f_i = \prod_{j=1}^{i} \frac{n+j}{n+j-k} B\, x_0$ for $0 \le i \le 2$. Then we compute with our algorithms, see below, $c_i \in \mathbb{Q}(n)$ and $g = g_0\, x_0 + g_1\, x_1$, $g_i \in \mathbb{F}$ such that $\sigma(g) - g = \sum_{i=0}^{2} c_i f_i$ holds. The found solution, translated back in terms of H_n and $\binom{n}{k}$, gives the solution in Example 3. □

More generally, suppose that for a problem of the type *GPTRT* we managed to construct a difference field $(\mathbb{F}, \sigma)$ in which the sequences $a_i(\boldsymbol{n}, k)$ and $h_j^{(i)}(\boldsymbol{n}, k)$ can be described with $a_i, h_j^{(i)} \in \mathbb{F}$. Then we try to solve *GPTRT* in so-called *higher order linear extensions*, in short *h.o.l. extension.* Namely, in the rational function field extension $\mathbb{E} := \mathbb{F}(x_0, \dots, x_s)$ of $\mathbb{F}$ with the field automorphism $\sigma : \mathbb{E} \to \mathbb{E}$ that is canonically defined as follows: σ acts on $\mathbb{F}$ like in the difference field $(\mathbb{F}, \sigma)$, $\sigma(x_i) = x_{i+1}$ for $0 \le i < s$ and

$$\sigma(x_s) = a_0\, x_0 + \dots + a_s\, x_s + a_{s+1}, \quad a_i \in \mathbb{F}. \tag{31}$$

Then, given such an h.o.l. extension $(\mathbb{E}, \sigma)$ of $(\mathbb{F}, \sigma)$, we represent $f_i(\boldsymbol{n}, k)$ in the form

$$f_i = h_0^{(i)}\, x_0 + \dots + h_s^{(i)}\, x_s + h_{s+1}^{(i)} \in \mathbb{F}\, x_0 \oplus \dots \oplus \mathbb{F}\, x_s \oplus \mathbb{F} \subseteq \mathbb{E} \tag{32}$$

and we try to solve problem

GPT in higher order extensions (*GPTHO*).

- **Given**: A h.o.l. extension $(\mathbb{F}(x_0, \dots, x_s), \sigma)$ of $(\mathbb{F}, \sigma)$ with (31) where $\mathbb{K} := \text{const}_\sigma \mathbb{F}$, $\mathbb{V} := \big(\mathbb{F}\, x_0 \oplus \dots \oplus \mathbb{F}\, x_s \oplus \mathbb{F}\big)$ and $f_0, \dots, f_d \in \mathbb{V}$;
- **find** $c_0, \dots, c_d \in \mathbb{K}$, not all zero, and a $g \in \mathbb{V}$ such that $\sigma(g) - g = c_0 f_0 + \dots + c_d f_d$.

3.2. **Our method in general difference fields.** Finally, we develop an algorithm that allows us to solve problem *GPTHO* under the assumption that one knows how to solve

Parameterized Linear Difference Equations (*PLDE*).

- **Given** a difference field $(\mathbb{F}, \sigma)$ with constant field $\mathbb{K}$, $a_0, \dots, a_m \in \mathbb{F}$, and $f_0, \dots, f_d \in \mathbb{F}$;
- **find all** $g \in \mathbb{F}$ and **all** $c_0, \dots, c_d \in \mathbb{K}$ with $a_m\, \sigma^m(g) + \dots + a_0\, g = c_0\, f_0 + \dots + c_d\, f_d$.

For simplicity let us consider first the special case $d = 0$ of problem *GPTHO*, i.e., given $f \in \mathbb{V}$, find a $g \in \mathbb{V}$ such that

$$\sigma(g) - g = f. \tag{33}$$

Example 7 (TSPP cont.)**.** Consider the TSPP problem (28) from Example 4. Then by taking $g = g_0\, x_0 + g_1\, x_1 + g_2\, x_2 \in \mathbb{Q}(n)(k)(x_0, x_1, x_2)$ and matching coefficients one obtains the equations

$$a_0\, \sigma(g_2) - g_0 = 1 \qquad \sigma(g_0) + a_1\, \sigma(g_2) - g_1 = 0 \qquad \sigma(g_1) + a_2\, \sigma(g_2) - g_2 = 0. \tag{34}$$

Note that any g_0, g_1, g_2 with (34) will produce a solution $g = g_0\, x_0 + g_1\, x_1 + g_2\, x_2$ with $\sigma(g) - g = x$. Now applying σ to the second equation of (34) gives $\sigma(g_1) = \sigma^2(g_0) + \sigma(a_1)\, \sigma^2(g_2)$ which allows us to transform the third equation of (34) to

$$\sigma^2(g_0) + \sigma(a_1)\, \sigma^2(g_2) + a_2\, \sigma(g_2) - g_2 = 0. \tag{35}$$

Finally, applying σ^2 to the first equation of (34) gives $\sigma^2(g_0) = \sigma^2(a_0)\, \sigma^3(g_2) - 1$ which turns equation (35) into

$$\sigma^2(a_0)\, \sigma^3(g_2) + \sigma(a_1)\, \sigma^2(g_2) + a_2\, \sigma(g_2) - g_2 = 1.$$

The crucial point is that we derived a linear difference equation in g_2 with known coefficients $\sigma^2(a_0)$, $\sigma(a_1)$ and a_2 in $\mathbb{Q}(n)(k)$ with $\sigma(k) = k+1$. Hence we can apply a refined version of the algorithm [1], which is a sub-algorithm in Sigma, and derive the solution $g_2 = \frac{k(2n-k-1)(2n+k+2)}{2(k+1)n(2n+1)} \in \mathbb{Q}(n)(k)$. Now observe that the first equation in (34) tells us how to compute g_0 from the already computed g_2. Moreover, the second equation of (34) allows us to compute g_1 from the already computed g_0. Furthermore observe that $g_0, g_2 \in \mathbb{F}$ satisfy the first equation of (34). Summarizing, the derived $g = g_0\,x_0 + g_1\,x_1 + g_2\,x_2$, given in (22), is a solution of (28). □

The following two lemmas give us a general recipe how the above problem (33) can be solved.

Lemma 1. *Let* $(\mathbb{F}(x_0,\dots,x_s),\sigma)$ *be a h.o.l. extension of* $(\mathbb{F},\sigma)$ *with* (32) *and let* $f, g \in \mathbb{F}\,x_0 \oplus \dots \oplus \mathbb{F}\,x_s \oplus \mathbb{F}$ *with* $f = h_0\,x_0 + \dots + h_s\,x_s + h_{s+1}$ *and* $g = g_0\,x_0 + \dots + g_s\,x_s + g_{s+1}$. *Then* $\sigma(g) - g = f$ *if and only if*

$$\sigma(g_{s+1}) - g_{s+1} = h_{s+1} - a_{s+1}\,\sigma(g_s), \tag{36}$$

$$g_0 = a_0\,\sigma(g_s) - h_0, \tag{37}$$

and for $1 \le i \le s$ *we have*

$$g_i = \sigma(g_{i-1}) + a_i\,\sigma(g_s) - h_i. \tag{38}$$

Proof: Define $L := \sigma(g) - g - f$. Then

$$L = \sum_{i=0}^{s-1}\big[\sigma(g_i)\,x_{i+1} - g_i\,x_i\big] + \sigma(g_s)\big[\sum_{i=0}^{s} a_i\,x_i + a_{s+1}\big] - g_s\,x_s + \sigma(g_{s+1}) - g_{s+1} - \sum_{i=0}^{s} h_i\,x_i - h_{s+1}$$

and therefore $L = d_{s+1} + d_0\,x_0 + \dots + d_s\,x_s$ with $d_0 = a_0\,\sigma(g_s) - g_0 - h_0$, $d_i = \sigma(g_{i-1}) + a_i\,\sigma(g_s) - g_i - h_i$ for $1 \le i \le s$, and $d_{s+1} = \sigma(g_{s+1}) - g_{s+1} + a_{s+1}\,\sigma(g_s) - h_{s+1}$. Since the x_i are transcendental over $\mathbb{F}$, the lemma is immediate. □

The crucial observation is that this system of first order linear difference equations (37) and (38) can be brought in an *uncoupled* (triangulated) form by the following

Lemma 2. *Let* $(\mathbb{F},\sigma)$ *be a difference field,* $h_i \in \mathbb{F}$ *for* $1 \le i \le e$ *and* $g_e \in \mathbb{F}$. *Then*

$$\sum_{j=0}^{s} \sigma^{s-j}(a_j)\sigma^{s-j+1}(g_s) - g_s = \sum_{j=0}^{s} \sigma^{s-j}(h_j) \tag{39}$$

if and only if there are $g_0 \dots, g_{s-1} \in \mathbb{F}$ *with* (37) *and* (38) *for* $0 < i \le s$.

Proof: Let $h_0, \dots, h_s \in \mathbb{F}$ and $g_s \in \mathbb{F}$. We show by induction on k for $1 \le k \le s$ with $g_k \in \mathbb{F}$ the following: there exist $g_0, \dots, g_{k-1} \in \mathbb{F}$ with (37) and (38) for $0 \le i < k$ if and only if

$$g_k = \sum_{j=0}^{k} \sigma^{k-j}(a_j)\sigma^{k-j+1}(g_s) - \sum_{j=0}^{k} \sigma^{k-j}(h_j). \tag{40}$$

Then for the particular choice $k = s$ the lemma is proven. First note that for $k = 0$ equation (37) is equivalent to (40), which proves the base case. In particular, if $s = 0$, we are already done. Now suppose that $0 \le k < s$, let $g_k \in \mathbb{F}$ and assume that we have shown already that equation (40) holds if and only if there are $g_0, \dots, g_k \in \mathbb{F}$ with (37) and (38) for $0 \le i \le k$. First suppose that there are $g_0, \dots, g_{k+1} \in \mathbb{F}$ with (37) and (38) for $0 \le i \le k+1$. Then by the induction assumption we may assume that (40) holds. Then plugging in the right hand side of (40) into $\sigma(g_k) + a_{k+1}\,\sigma(g_s) - g_{k+1} = h_{k+1}$ gives

$$\sum_{j=0}^{k} \sigma^{k-j+1}(a_j)\sigma^{k-j+2}(g_s) - \sum_{j=0}^{k} \sigma^{k-j+1}(h_j) + a_{k+1}\,\sigma(g_s) - g_{k+1} = h_{k+1} \tag{41}$$

which is equivalent to

$$g_{k+1} = \sum_{j=0}^{k+1} \sigma^{k-j+1}(a_j)\sigma^{k-j+2}(g_s) - \sum_{j=0}^{k+1} \sigma^{k-j+1}(h_j). \tag{42}$$

Contrary suppose that we are given a $g_{k+1} \in \mathbb{F}$ with (42) or equivalently (41). We can construct $g_0, \dots, g_k \in \mathbb{F}$ such that equations (37) and (38) for $1 \leq i \leq k$ hold. Hence by the induction assumption (40) follows. (40) and (41) imply $\sigma(g_k) + a_{k+1}\,\sigma(g_s) - g_{k+1} = h_{k+1}$. □

Example 8 (TSPP cont.)**.** Essential use of Lemma 2 has been made in [5] to prove hypergeometric multi-sum identities. □

Consequently the telescoping equation (33) for $g = g_0\,x_0 + \dots + g_s\,x_s + g_{s+1} \in \mathbb{V}$ holds if and only if we have (39), (36),(37), and (38) for $0 < i < s$. This fact produces immediately an algorithm to find such a $g \in \mathbb{V}$ with (39) if an algorithm is given that can solve linear difference equations.

Algorithm 1. Indefinite summation (telescoping).

`Telescoping`$((\mathbb{F}(x_0, \dots, x_s), \sigma), f)$

Input: A h.o.l. extension $(\mathbb{F}(x_0, \dots, x_s), \sigma)$ of $(\mathbb{F}, \sigma)$ and $f = h_0\,x_0 + \dots h_s\,x_s + x_{s+1} \in \mathbb{V}$ where $\mathbb{V} = \mathbb{F}\,x_0 \oplus \dots \oplus \mathbb{F}\,x_s \oplus \mathbb{F}$.

Output: A solution $g \in \mathbb{V}$ with $\sigma(g) - g = f$ if it exists.

(1) Decide constructively[3] if there is a solution $g_s \in \mathbb{F}$ for (39). If no, RETURN “No solution”.
(2) Otherwise, take such a g_s and decide constructively[3] if there is a solution $g_{s+1} \in \mathbb{F}$ for (36). If no, RETURN “No solution”.
(3) Otherwise, take such a g_{s+1} and compute g_0 by (37) and derive successively the remaining g_i by (38).
(4) RETURN $g = g_0\,x_0 + \dots g_s\,x_s + g_{s+1}$.

Remark. A special case of Algorithm 1 can be related to [3].

Next, we generalize this algorithm to solve Problem *GPTHO* for the homogeneous case, i.e., $\alpha_{s+1} = 0$ and $h_{s+1}^{(j)} = 0$ for $0 \leq j \leq d$. The main idea is to take indeterminates c_i, replace $c_0\,f_0 + \dots + c_d\,f_d$ with f, and to look simultaneously for solutions $g \in \mathbb{V}$ and $c_i \in \mathbb{K}$. More precisely, there is the following algorithm.

(1) Write $f_i \in \mathbb{V}$ as in (32) with $h_j^{(i)} \in \mathbb{F}$. Then compute[3] all solutions $(c_0, \dots, c_d, g) \in \mathbb{K}^{d+1} \times \mathbb{F}$ s.t.

$$\sum_{j=0}^{s} \sigma^{s-j}(a_j)\sigma^{s-j+1}(g) - g = \sum_{i=0}^{d} c_i \sum_{j=0}^{s} \sigma^{s-j}(h_j^{(i)}). \tag{43}$$

(2) If there are only solutions where all $c_i = 0$, there is no solution for problem *GPTHO*.
(3) Otherwise we take such a solution, say $(c_0, \dots, c_d, g_s)$, with some $c_i \neq 0$ and set $f := \sum_{i=0}^{d} c_i\,f_i \in \mathbb{F}\,x_0 \oplus \dots \oplus \mathbb{F}\,x_s$. Now we compute a $g \in \mathbb{V}$ with (33) by applying[4] Algorithm 1.

Example 9. Take the difference field $(\mathbb{F}(x_0, x_1, x_2), \sigma)$ with $\mathbb{F} := \mathbb{Q}(n)(k)(B)(H)$ and the $f_i = h_i^{(0)}\,x_0 + h_i^{(1)}\,x_1 \in \mathbb{F}\,x_0 \oplus \mathbb{F}\,x_1$ from Example 6, i.e., $h_i^{(0)} = \prod_{j=1}^{i} \frac{n+j}{n+j-k} B$ and $h_i^{(1)} = 0$. In order to obtain $c_i \in \mathbb{Q}(n)$ and $g \in \mathbb{F}\,x_0 \oplus \mathbb{F}\,x_1$ with $\sigma(g) - g = \sum_{i=0}^{2} c_i f_i$, we compute the solution $c_0(n) = 4n^2(1+n)^2$, $c_1(n) = 2n^2(1+n)(3+2n)$, $c_2(n) = n^2(1+n)(2+n)$, and

$$\begin{aligned} g_1 = &-(1+k)(2k^2(1+n)^2 + n(2+8n+9n^2+3n^3) - k(2+8n+13n^2+6n^3) \\ &+ kn(1+n)(-2-6n-3n^2+2k(1+n))H)B \Big/ \big((-1+k-n)(1+(1+k)H)\big) \in \mathbb{F} \end{aligned}$$

for $\sigma(a_0)\sigma^2(g_1) + a_1\sigma(g_1) - g_1 = \sum_{i=0}^{3} c_i\sigma(h_i^{(0)})$ where the a_i are given in (30). Now define $f = \sum_{i=0}^{2} c_i\,f_i = x_0 \sum_{i=0}^{2} c_i\,h_i^{(0)}$. Then by Lemma 1 and 2 it follows that $g = g_0\,x_0 + g_1\,x_1$ with $g_0 = a_0\sigma(g_1) - \sum_{i=0}^{2} c_i\,h_i^{(0)}$ is a solution for $\sigma(g) - g = f = \sum_{i=0}^{2} c_i\,f_i$; see Example 3. □

---

[3] By assumption this is possible by solving a specific instance of problem *PLDE* in the difference field $(\mathbb{F}, \sigma)$.

[4] The solution is guaranteed: we can skip step (1) in Algorithm 1, since the already computed $g_s \in \mathbb{F}$ satisfies (39). With $g_{s+1} = 0$ in step (2) the computed output $g = g_0x_0 + \dots + g_sx_s$ gives the desired solution.

Remark 1. If α_{s+1} and $h_{s+1}^{(j)}$ are not 0, we can extend these ideas by using reduction techniques from [15] based on linear algebra. More precisely, by following the ideas from above one first computes all $h = g_0\,x_0 + \dots + g_s\,x_s$ with $g_i \in \mathbb{F}$ and all $c_i \in \mathbb{K}$ s.t. $\sigma(h) - h - \sum_{i=0}^{d} c_i\,f_i \in \mathbb{F}$; all those solutions $(c_0, \dots, c_d, h)$ form a finite dimensional vector space over $\mathbb{K}$. After computing a basis, say $\{(c_{i0}, \dots, c_{id}, h_i)\}_{1\leq i\leq u}$, one looks for all constants $k_1, \dots, k_u \in \mathbb{K}$ and all $g_{s+1} \in \mathbb{F}$ s.t.

$$\sigma(k_1 h_1 + \dots k_u h_u + g_{s+1}) - (k_1 h_1 + \dots k_u h_u + g_{s+1}) = k_1 \sum_{j=0}^{d} c_{1j} f_j + \dots + k_u \sum_{j=0}^{d} c_{uj} f_j$$

$$\Leftrightarrow \quad \sigma(g_{s+1}) - g_{s+1} = k_1 \Big[\sum_{j=0}^{d} c_{1j} f_j - \sigma(h_1) + h_1\Big] + \dots + k_u \Big[\sum_{j=0}^{d} c_{uj} f_j - \sigma(h_u) + h_u\Big] \qquad (44)$$

holds. More precisely, one solves a certain instance of problem *PLDE* with $m = 1$. Then any solution $k_i \in \mathbb{K}$ and $g_{s+1} \in \mathbb{F}$ of (44) gives a solution $c_i := \sum_{j=1}^{u} k_j c_{ji} \in \mathbb{K}$ and $g := k_1 h_1 + \dots + k_u h_u + g_{s+1} \in \mathbb{V}$ for *GPTRT*. Note that with linear algebra arguments one can show that this approach gives us **all** solutions for *GPTRT*. $\diamondsuit$

Summarizing, we obtain the following

Theorem 1. *There is an algorithm that solves problem GPTHO if one can solve problem PLDE.*

Observe that this theorem is contained in [8] if one restricts to h.o.l. extensions of the form (32) with $a_{s+1} = 0$. The improvement in our result is that we can avoid uncoupling algorithms; see [13]. Instead, in Lemma 2 we provide a generic formula for an uncoupled system that is equivalent to the given one.

3.3. **A new algorithm for special difference fields.** So far we have shown that one can handle problem *GPTHO* for any difference field in which problem *PLDE* can be solved. As worked out in [8] this can be achieved for the rational case $(\mathbb{F}, \sigma)$ with $\mathbb{F} = \mathbb{K}(k)$ and $\sigma(k) = k + 1$ or the q-case with $\mathbb{F} = \mathbb{K}(q)(x)$ and $\sigma(x) = q\,x$ by extended versions of the algorithms [1, 2].

More generally, due to recent algorithmic results [6, 28, 27, 31] one can solve[5] problem *PLDE* and therefore problem *GPTHO* in $\Pi\Sigma$-fields. With this algorithmic difference field machinery, implemented in Sigma, one has new algorithms in hand that allow us to solve problem *GPTRT* and *SPTRS* over rational expressions involving indefinite nested sums and products.

Remark 2. Informally, a $\Pi\Sigma$-field is nothing else than a difference field $(\mathbb{F}, \sigma)$ with constant field $\mathbb{K}$ where $\mathbb{F} := \mathbb{K}(t_1)\dots,(t_e)$ is a rational function field and the application of σ on the t_i's is recursively defined over $1 \leq i \leq e$ with $\sigma(t_i) = \alpha_i\,t_i + \beta_i$ for $\alpha_i, \beta_i \in \mathbb{K}(t_1)\dots(t_{i-1})$; we omitted some technical conditions given e.g. in [15, 26, 32].
For instance, all difference fields $(\mathbb{F}, \sigma)$ in the Examples 4, 5 and 6 form $\Pi\Sigma$-fields. This means that sums like $H_k = \sum_{i=1}^{k} \frac{1}{i}$, products like $\binom{n}{k} = \prod_{i=1}^{k} \frac{n-i+1}{i}$, or expressions like the summand in (20) can be expressed in $\Pi\Sigma$-fields. Observe that all such expressions represented in a $\Pi\Sigma$-field have the following property: their sums and products shifted in k can be expressed by their unshifted sums and products, like $S_k\,\mathrm{H}_k = \mathrm{H}_k + \frac{1}{k+1}$ or $S_k \binom{n}{k} = \frac{n-k}{k+1}\binom{n}{k}$. $\diamondsuit$

We emphasize that Karr' original summation algorithm [15] solves problem *PLDE* for the case $m = 1$ in a given $\Pi\Sigma$-field, and hence implements problem *GPT* in the $\Pi\Sigma$-field setting; see (47). Since our algorithm can solve Problem *GPTRT* over such a $\Pi\Sigma$-field, it completely covers Karr's algorithm — actually, Sigma contains a simplified version of Karr's summation algorithm; see [30]. On the other side, our algorithm restricted to the homogeneous case, see above, can be embedded in the general setting of [8]. In some sense, we have introduced a common framework that combines both, Karr's algorithm and big parts of Chyzak's ∂-finite tool box [8].

[5] More precisely, with the techniques introduced in [28], one eventually finds all solutions of parameterized linear difference equations by increasing incrementally the search space.

4. The Recurrence Method for Multi-Summation

Consider the following multi-summation problem. **Given** $S(\boldsymbol{n}) = \sum_{k=\alpha}^{\beta} f(\boldsymbol{n}, k)T(\boldsymbol{n}, k)$ where α and β are integer-linear in $\boldsymbol{n}$ and where for the summand $f(\boldsymbol{n}, k)T(\boldsymbol{n}, k)$ the following properties hold. $T(\boldsymbol{n}, k)$ might be a multi-sum of the form

$$T(\boldsymbol{n}, k) = \sum_{k_1} h_1(\boldsymbol{n}, k, k_1) \sum_{k_2} h_2(\boldsymbol{n}, k, k_1, k_2) \cdots \sum_{k_u} h_u(\boldsymbol{n}, k, k_1, \dots, k_u)$$

where we assume that the summation bounds in all the sums $\sum_{k_i}$ are integer-linear in $\boldsymbol{n}$ and $k, k_1, \dots, k_{i-1}$; moreover, we suppose that $f(\boldsymbol{n}, k)$ and the $h_i(\boldsymbol{n}, k, k_1, \dots, k_i)$ can be represented in a $\Pi\Sigma$-field, i.e., in rational expressions involving indefinite nested sums and products. **Find** a recurrence of the type (6) for given $\boldsymbol{\gamma_i} \in \mathbb{N}^r$.

Example 10 (TSPP cont.)**.** All our summation problems in [5] fit into this problem class; see for instance (62). The following ideas were crucial to handle all these problems. □

In this section we discuss how such a problem could be attacked using the tools described in the previous sections. First one tries to derive recurrences of the type (8) and (9) for the summand $T(\boldsymbol{n}, k)$, then, if necessary, reduces problem *SPTRS* to the simpler problem *GPTRT*, and afterwards applies our algorithms from Section 3 to solve problem *GPTRT*.

Hence, in order to follow this strategy with our methods, we only have to explain how we can derive recurrences of the type (8) and (9). For the sake of simplicity we suppress additional parameters and focus on the following problem. **Given** $S(m, n) = \sum_k f(m, n, k)T(m, n, k)$ with

$$T(m, n, k) = \sum_{k_1} h_1(m, n, k, k_1) \sum_{k_2} h_2(m, n, k, k_1, k_2) \cdots \sum_{k_u} h_u(m, n, k, k_1, \dots, k_u)$$

where $f(m, n, k)$ and the h_i can be expressed in a $\Pi\Sigma$-field. **Find** recurrences of the type

$$S(m, n + d + 1) = a_0(m, n)S(m, n) + \dots + a_d(m, n)S(m, n + d) + a_{d+1}(m, n) \tag{45}$$

or

$$S(m + 1, n) = b_0(m, n)S(m, n) + \dots + b_d(m, n)S(m, n + d) + b_{d+1}(m, n). \tag{46}$$

To accomplish this task, we propose the following *recurrence method* based on recursion.

• **Base case:** $T(m, n) = 1$. In this case, the summand $f(m, n, k)$ of $S(m, n)$ can be expressed in a $\Pi\Sigma$-field, say $(\mathbb{F}, \sigma)$ with $\mathbb{K} := \text{const}_\sigma\mathbb{F}$. Hence we try to find (45), resp. (46), with *SPT* in our $\Pi\Sigma$-field; i.e., we try to find $c_i \in \mathbb{K}$ and $g \in \mathbb{F}$ such that

$$\sigma(g) - g = c_{-1}\, f_{-1} + c_0\, f_0 + \dots + c_d\, f_d \tag{47}$$

where $f_i \in \mathbb{F}$ stands for $f(m, n + i, k)$, $0 \leq i \leq d$, and $f_{-1} \in \mathbb{F}$ stands for $f(m + 1, n, k)$, or is 0, respectively. More precisely, starting from $d = 0$ for our problem (47) one increments the order d until a non-trivial solution is found, i.e., some c_i are non-zero. In this case, RETURN the resulting recurrence (45) or (46). If d gets too large without any solution, STOP with the comment "Failure". — With Sigma we can accomplish this task by using the function call (61).

• **Recursion:** $T(m, n, k) \neq 0$. Before we can proceed to find (45), resp. (46), with *SPTRS*, we have to derive recurrences for $T(m, n, k)$. For the case (45) we need recurrences of the form

$$T(m, n, k + \delta + 1) = a'_0(m, n, k)T(m, n, k) + \dots + a'_\delta(m, n, k)T(m, n, k + \delta) + a'_{\delta+1}(m, n, k) \tag{48}$$

and

$$T(m, n + 1, k) = b'_0(m, n, k)T(m, n, k) + \dots + b'_\delta(m, n, k)T(m, n, k + \delta) + b'_{\delta+1}(m, n, k). \tag{49}$$

For the case (46) we need, besides (48) and (49), a recurrence of the form

$$T(m + 1, n, k) = b^*_0(m, n, k)T(m, n, k) + \dots + b^*_\delta(m, n, k)T(m, n, k + \delta) + b^*_{\delta+1}(m, n, k). \tag{50}$$

In order to accomplish this task, we apply again our *recurrence method* on the sub-problems (48), (49) or (48), (49), (50), respectively. If we fail, STOP with the comment “Failure”. Otherwise we proceed as follows.

• **Solving the problem:** We try to solve the corresponding problem *SPTRS*, namely, find $c_i(m,n)$ and $g(m,n,k)$ such that

$$\begin{aligned} g(m,n,k+1)-g(m,n,k) &= c_{-1}(m,n)f(m+1,n,k)T(m+1,n,k)\\ &\quad + c_0(m,n)f(m,n,k)T(m,n,k)+\cdots+c_d(m,n)f(m,n+d,k)T(m,n+d,k) \end{aligned} \tag{51}$$

or

$$\begin{aligned} &g(m,n,k+1)-g(m,n,k)\\ &\quad = c_0(m,n)f(m,n,k)T(m,n,k)+\cdots+c_d(m,n)f(m,n+d,k)T(m,n+d,k), \end{aligned} \tag{52}$$

respectively. Now we go on as proposed in Section 2: We reduce problem *SPTRS* to *GPTRT* and try to solve Problem *GPTHO* — if possible — in an appropriate $\Pi\Sigma$-field. Namely, given such a $\Pi\Sigma$-field $(\mathbb{F},\sigma)$, we increase $d\geq 0$ in our *GPTHO* problem step by step until a nontrivial solution for (51), resp. (52), is found. In this case RETURN the resulting recurrence (45) or (46). Otherwise, if d gets too large without any solution, STOP with the comment “Failure”. — With Sigma we can accomplish this task by using the function call (60).

Observe that the basic idea of the *recurrence method* has been applied already in Examples 1 and 2. In particular, looking at Example 1, our method can be specialized to hypergeometric multi-summation, i.e.,

$$S(\boldsymbol{n})=\sum_{k_1}\cdots\sum_{k_u}h(\boldsymbol{n},k_1,\dots,k_u), \tag{53}$$

where[6] $h(\boldsymbol{n},k_1,\dots,k_u)$ is hypergeometric in all parameters, as follows.
• If we run into the base case, we try to compute homogeneous recurrences for the inner most sum by applying [37, 18] or Sigma.
• Otherwise, we have to solve problem *SPTRS* of the type (51) or (52). Namely, assuming that the $a'_i(m,n,k), b'_i(m,n,k)$ and $b^*_i(m,n,k)$ from (48), (49) and (50) are rational functions in m, n and k, we have to solve problem *GPTHO* in a difference field of the type $(\mathbb{K}(m,n)(k),\sigma)$ with constant field $\mathbb{K}(m,n)$ and $\sigma(k)=k+1$. This can be achieved with our algorithm in Section 3.2 by using a variation of algorithm [1], that is contained in Sigma. In [21] these ideas are analysed in further details.

The non-trivial examples in [5] illustrate the successful application of our method. In particular, we want to emphasize that we managed to find straightforward alternative proofs for all double sum identities in [35], like identity (10) from Example 1.

Example 11. Following our strategy in Example 1, the recurrence (15) for the double sum given in (10) can be computed in 12 seconds by using Sigma (Mathematica 5); see Section 5.2.
In comparison, the Wegschaider/Riese package MultiSum [35, 17][7] needs about 510 seconds to compute the same recurrence (15) on the same computer platform using Mathematica 5. Moreover the intermediate result of the summand recurrence fills several pages, see [35, Section 5.7.6], whereas our result is rather compact, see `Out[12]`. □

In [8] three strategies for tackling hypergeometric multi-summation have been indicated. From these strategies only the following one[8] has been implemented so far: Fix $d\in\mathbb{N}$ and look for all linear recurrences with shifts in $T(\boldsymbol{n}+\boldsymbol{\gamma},k+\delta)$ where $\boldsymbol{\gamma}\in\mathbb{N}^r$, $\delta\in\mathbb{N}$ and $\gamma_1+\cdots+\gamma_r+\delta\leq d$. Concerning such strategies the following remarks are in place.

---

[6] For the sake of simplicity we restrict ourselves to sums where all summations are taken over finite summand supports. With this restriction *homogeneous* sum recurrences are guaranteed.

[7] This approach is based on ideas of Sister Celine and Wilf/Zeilberger [36] and supplemented by [34] and random parameter substitution [24].

[8] Following this strategy we managed to derive recurrence (15) for the double sum given in (10) in more than 2300 seconds by using the package Mgfun in Maple 8.

(1) Looking for each recurrence (8) or (9) separately, like in our approach, amounts to keeping the underlying linear algebra problems as small as possible. But, looking in one stroke for a whole system of recurrences results in a drastic increase of complexity.
(2) Moreover, one usually does not have any control over the structure of the derived recurrence system. In particular those systems usually do not allow to represent $T(\boldsymbol{n}+\boldsymbol{\gamma}, k+\delta)$ in a normalized form. Hence Gröbner basis algorithms [9, 7] must be used in order to transform such recurrence systems into an appropriate shape; see [8].

Besides this, in [8] an extension of the FGLM algorithm [12] is proposed which allows to compute the recurrences iteratively. Following these ideas and using a lexicographical monomial ordering on the shifts, one essentially ends up with a strategy which reduces to our recurrence method for hypergeometric multi-sums of the form (53). Summarizing, our surprisingly simple instantiation of Chyzak's method [8] enables us to tackle a huge class of multi-sum problems in a very efficient manner.

We also remark that our *recurrence method* based on problem *SPTRS* can be easily carried over from the shift/difference field case to the differential field case; this aspect might also contribute to the multi-integration approach.

5. The Implementation Within Our Summation Package Sigma

In this section we will describe the usage of our extended Mathematica package Sigma that not only solves *GPT* for rational expressions involving indefinite nested sums and products, see [20, 10, 11, 29], but the more general Problems *SPTRS* and *GPTRT*. Subsequently, we will illustrate all these new features of Sigma.

First we load our package into the Mathematica system by typing

In[1]:= << **Sigma**`

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

Sigma splits into two main parts, namely indefinite and definite summation.

5.1. Indefinite summation.

As a first introductory example we consider the TSPP problem from Example 2. Namely, eliminate the outermost summation quantifier in $S(n) = \sum_{k=0}^{2n} T(n,k)$ where the double sum $T(n,k) = \mathtt{T}[\mathrm{k}]$ given in (20) satisfies the recurrence:

In[2]:= $\mathbf{recDS = 2\ (2+k)^2\ (k-2\ n)\ (1+k+2\ n)\ T[k] + \big(-12-46\ k-58\ k^2-29\ k^3-5\ k^4 +12\ n+20\ k\ n+6\ k^2\ n+24\ n^2+40\ k\ n^2+12\ k^2\ n^2\big)\ T[1+k]+ \big(18+55\ k+59\ k^2+26\ k^3+4\ k^4-6\ n-14\ k\ n-6\ k^2\ n-12\ n^2-28\ k\ n^2- 12\ k^2\ n^2\big)\ T[2+k]-(2+k-2\ n)\ (3+k+2\ n)\ T[3+k]==0;}$

We set up our summation problem as follows[9].

In[3]:= $\mathbf{mySum = SigmaSum[T[k], \{k, 0, 2n\}];}$

Out[3]= $\sum_{k=0}^{2n} \mathtt{T}[k]$

Remark 3. Generally, the functions `SigmaSum` and `SigmaProduct` are used to define rational expressions involving indefinite nested sums and products that can be represented in $\Pi\Sigma$-fields. We also provide several other functions, like `SigmaHNumber`, `SigmaBinomial` or `SigmaPower`, to define harmonic numbers, binomials or powers. Internally, these objects are also represented in terms of sums and products that can be converted into $\Pi\Sigma$-fields. For instance, `SigmaHNumber`[k] produces the kth harmonic number H_k which alternatively could be described by `SigmaSum[1/i,{i,1,k}]`. ◊

[9] Note that the initial values $T[0]$, $T[1]$ and $T[2]$ are not specified further. Nevertheless we can evaluate $T[k]$ with $\langle T[0], T[1], T[2], 3T[2] - \frac{6(n+1)(2n-1)}{(2n+3)(n-1)}T[1] + \frac{8n(2n+1)}{(n-1)2n+3)}T[0], \dots\rangle$ by linear combinations of $T[0]$, $T[1]$ and $T[2]$.

Next, our indefinite summation algorithm is applied using the function call

In[4]:= $\textbf{SigmaReduce}[\textbf{mySum}, \{\textbf{recDS}, \textbf{T}[\textbf{k}]\}]$

Out[4]= $\dfrac{-(1+2\,n)\,(3\,T[0]-T[1])-2\,T[2\,(1+n)]+(5+2\,n)\,T[1+2\,n]}{1+2\,n}$

which gives the identity (23). Internally, we solve the corresponding telescoping problem, see Example 2, by first translating it into the underlying difference field, see Example 4, and afterwards solving it in this setting with our algorithms, see Example 7.

In the next example we derive a closed form evaluation of the sum

In[5]:= $\textbf{mySum} = \sum_{k=2}^{n} \dfrac{\textbf{HE}[k]\,(-2\,x+(2+3\,k+2\,k^2-2\,x)\,H_k - k\,(1+k)\,(3+2\,k-2\,x)\,H_k^2)}{H_k\,(-1+k\,H_k)\,(1+H_k+k\,H_k)};$

where HE[k] stands for the Hermite polynomials that can be defined as follows.

In[6]:= $\textbf{recHE} = \textbf{HE}[k+2] == 2x\,\textbf{HE}[k+1]-2(k+1)\textbf{HE}[k];\qquad \textbf{HE}[0]=1;\quad \textbf{HE}[1]=2x;$

After inserting our summation problem we eliminate the summation quantifier by executing:

In[7]:= $\textbf{SigmaReduce}[\textbf{mySum}, \{\textbf{recHE}, \textbf{HE}[\textbf{k}]\}]$

Out[7]= $-\dfrac{2}{3}\,(-3-8\,x+6\,x^2)+\dfrac{HE[1+n]}{H_n}-\dfrac{2\,(1+n)^2\,HE[n]}{1+H_n+n\,H_n}$

Remark 4. In general, suppose that we are given a recurrence `rec` of the form

$$a_0T[k]+\cdots+a_sT[k+s]+a_{s+1}==0 \tag{54}$$

and a sum

$$\texttt{mySum} = \sum_{k=\alpha}^{\beta} \underbrace{\Big(f_0T[k]+\cdots+f_sT[k+s]+f_{s+1}\Big)}_{=f(k)} \tag{55}$$

where the a_i and f_i are rational expressions involving indefinite nested sums and products. In order to insert such a summation problem, we provide various functions; see Remark 3. Note that the a_i and f_i may also depend on extra parameters. Then, after defining such a summation problem, with the function call

$$\texttt{SigmaReduce}[\texttt{mySum}, \{\texttt{rec}, \texttt{T}[\texttt{k}]\}] \tag{56}$$

one tries to eliminate the outermost summation quantifier by following the strategy as in problem *GPTRT* with $d=0$; more precisely, one tries to solve problem *GPTHO* for the underlying $\Pi\Sigma$-field. If the summand $f(k)$ of (55) is free of $T[k]$, i.e., $f_i=0$ for $0\leq i\leq s$, one can skip $\{\texttt{rec}, \texttt{T}[\texttt{k}]\}$ in (56). In this case our algorithm reduces to the former version of Sigma [26, 29]. ◊

5.2. **Definite summation.** In our first example we will prove identity (10) by following the strategy described in Example 1. Namely, we first compute a recurrence for the double sum on the left hand side of (10) by following our recurrence method; see Section 4. More precisely, we insert the inner sum $T(n,k)$ of our double sum

In[8]:= $\textbf{sumT} = \sum_{s=0}^{n} \binom{n}{k}\cdot\binom{n}{s}\cdot\binom{k+n}{k}\cdot\binom{-k+2\,n-s}{n}\cdot\binom{n+s}{s}\cdot(-1)^{k+n+s}\cdot;$

and compute the recurrence (11) with the function call:

In[9]:= $\textbf{rec} = \textbf{GenerateRecurrence}[\textbf{sumT}, \textbf{k}, \textbf{RecOrder} \to \textbf{2}]/.\textbf{SUM} \to \textbf{T}$

Out[9]= $(k-n)^3\,(1+k+n)\,(2+k+n)\,T[k]-(1+k)^2\,(2+k+n)\,(k+2\,k^2-3\,n- 6\,k\,n+3\,n^2)\,T[1+k]+(1+k)^2\,(2+k)^2\,(k-3\,n)\,T[2+k]==0$

This means that $\texttt{T}[\texttt{k}] = T(n,k) = \texttt{sumT}$ satisfies the output recurrence `Out[9]`. Note that this result could be also obtained by any implementation of Zeilberger's algorithm, like for instance [22]. Similarly, we derive recurrence (12) either with a variation of Zeilberger's algorithm [18], or with Sigma by setting in addition the option $\texttt{OneShiftIn} \to \texttt{n}$:

In[10]:= $\mathbf{recInN = GenerateRecurrence[sumT, k, OneShiftIn \to n, RecOrder \to 1]/.SUM \to T}$

Out[10]= $-(1+k+n)\,\big(-5\,k+12\,k^2-10\,k^3+3\,k^4+3\,n-32\,k\,n+42\,k^2\,n-16\,k^3\,n+ 15\,n^2-57\,k\,n^2+33\,k^2\,n^2+21\,n^3-30\,k\,n^3+9\,n^4\big)\,T[k]+ (1+k)^2\,(-1+k-3\,n)\,\big(6-8\,k+3\,k^2+12\,n-8\,k\,n+6\,n^2\big)\,T[1+k]+ (-1+k-n)^3\,(1+n)^2\,T[1+n,k]==0$

Given all these ingredients we finally compute the creative telescoping solution for the double sum

In[11]:= $\mathbf{mySum} = \sum_{\mathbf{k=0}}^{\mathbf{n}} \mathbf{T[k]};$

by typing in:

In[12]:= $\mathbf{creaSol = CreativeTelescoping[mySum, n, \{rec, T[k]\}, recInN, RecOrder \to 2]}$

Out[12]= $\big\{\{0,0,0,1\},\big\{-4\,(1+n)^3\,(3+4\,n)\,(5+4\,n),-2\,(1+n)^2\,(3+2\,n)\,\big(7+9\,n+3\,n^2\big),(1+n)^2\,(2+n)^3, \big(k^2\,\big(960-3192\,k+3680\,k^2-2042\,k^3+1248\,k^4-1112\,k^5+582\,k^6-134\,k^7+10\,k^8+7536\,n- 21720\,k\,n+21304\,k^2\,n-10982\,k^3\,n+6404\,k^4\,n-4095\,k^5\,n+1421\,k^6\,n-199\,k^7\,n+7\,k^8\,n+ 25804\,n^2-63504\,k\,n^2+52698\,k^2\,n^2-24334\,k^3\,n^2+12025\,k^4\,n^2-5292\,k^5\,n^2+1123\,k^6\,n^2- 74\,k^7\,n^2+50716\,n^3-104481\,k\,n^3+71985\,k^2\,n^3-28139\,k^3\,n^3+10608\,k^4\,n^3-2905\,k^5\,n^3+ 290\,k^6\,n^3+63175\,n^4-106032\,k\,n^4+58545\,k^2\,n^4-17878\,k^3\,n^4+4469\,k^4\,n^4-578\,k^5\,n^4+ 51793\,n^5-68088\,k\,n^5+28333\,k^2\,n^5-5928\,k^3\,n^5+727\,k^4\,n^5+27970\,n^6-27054\,k\,n^6+ 7556\,k^2\,n^6-804\,k^3\,n^6+9598\,n^7-6087\,k\,n^7+857\,k^2\,n^7+1899\,n^8-594\,k\,n^8+165\,n^9\big)\,T[k]- k^2\,(1+k)^2\,(-1+k-3\,n)\,\big(-200+952\,k-1182\,k^2+610\,k^3-136\,k^4+10\,k^5-836\,n+3260\,k\,n- 3183\,k^2\,n+1220\,k^3\,n-182\,k^4\,n+7\,k^5\,n-1426\,n^2+4386\,k\,n^2-3174\,k^2\,n^2+808\,k^3\,n^2- 61\,k^4\,n^2-1271\,n^3+2901\,k\,n^3-1389\,k^2\,n^3+177\,k^3\,n^3-625\,n^4+ 944\,k\,n^4-225\,k^2\,n^4-161\,n^5+121\,k\,n^5-17\,n^6\big)\,T[1+k]\big)/\big((-2+k-n)^3\,(-1+k-n)^3\big)\big\}\big\}$

This means that each entry $\{c_0, c_1, c_2, g\}$ in `Out[12]` gives one particular solution of (14). Afterwards we sum this telescoping equation (14) over k from 0 to n and obtain the following result.

In[13]:= $\mathbf{TransformToRecurrence[creaSol, mySum, n, \{rec, T[k]\}, recInN]}$

Out[13]= $\big\{-4\,(1+n)\,(3+4\,n)\,(5+4\,n)\,SUM[n]- 2\,(3+2\,n)\,\big(7+9\,n+3\,n^2\big)\,SUM[1+n]+(2+n)^3\,SUM[2+n]==0\big\}$

If we are not interested in the proof certificate given in `Out[12]`, see Example 1, one could immediately derive this recurrence by replacing `CreativeTelescoping` with `GenerateRecurrence` in `In[12]`. To complete our proof of identity (10) we verify that also the right hand side of (10) satisfies the recurrence in `Out[13]` for $n \geq 0$; more precisely we compute this recurrence with the function call `GenerateRecurrence[SigmaSum[SigmaBinomial[n,k]^4,{k,0,n}]]`. Since both sides of (10) are equal for $n = 0, 1$, they represent the same sequence for $n \geq 0$.

Remark 5. In general, our recurrence method from Section 4 can be applied using Sigma as follows. Suppose that we are given a recurrence `rec` of the form (54), recurrences `recInN` and `recInM` of the forms

$$T[n+1,k] == b_0T[k] + \cdots + b_sT[k+s] + b_{s+1}, \tag{57}$$

$$T[m+1,k] == b_0^*T[k] + \cdots + b_s^*T[k+s] + b_{s+1}^*, \tag{58}$$

respectively, and a definite sum

$$\texttt{mySum} = \sum_{k=\gamma_0}^{\gamma_1 m+\gamma_2 n+\alpha} \underbrace{\big(f_0T[k] + \cdots + f_sT[k+s] + f_{s+1}\big)}_{=f(m,n,k)}, \quad \gamma_i \in \mathbb{Z}, \tag{59}$$

where α is an integer that may depend on other parameters. The a_i, b_i, b_i^*, f_i can be rational expressions involving indefinite nested sums and products; to insert such objects see Remark 3.

Moreover, the a_i, b_i, b_i^*, f_i can depend besides m, n, k on any parameter. Then by calling

$$\texttt{CreativeTelescoping}[\texttt{mySum}, \texttt{n}, \{\texttt{rec}, \texttt{T}[\texttt{k}]\}, \texttt{recInN}, \texttt{OneShiftIn} \to \{\texttt{recInM}, \texttt{m}\}, \texttt{RecOrder} \to \texttt{d}] \quad (60)$$

one searches for all creative telescoping solutions $\{c_{-1}(m,n), c_0(m,n), \dots, c_d(m,n), g(m,n,k)\}$ such that (51) holds. Note that g may depend on any parameter whereas the c_i are free of k. Similarly, with `GenerateRecurrence` one computes the corresponding recurrence of the form

$$\text{SUM}[m+1, n] == e_0\,\text{SUM}[n] + \dots + e_d\,\text{SUM}[n+d] + e_{d+1},$$

where the e_i can be usually represented in a $\Pi\Sigma$-field.
If the option $\texttt{OneShiftIn} \to \{\texttt{recInM}, \texttt{m}\}$ is skipped in (60), the additional shift in m is not considered; see for instance `In[12]`. Moreover, if we have the trivial recurrence relation T[n+1,k] == $T[n,k]$ in (57), also the input `recInN` can be omitted in (60); typical examples are given in `In[15]` and `In[18]`.
If the summand $f(m,n,k)$ of (59) is free of $T[k+i]$, i.e., $f_i = 0$ for $0 \le i \le s$, the function call (60) reduces to

$$\texttt{CreativeTelescoping}[\texttt{mySum}, \texttt{n}, \texttt{OneShiftIn} \to \texttt{m}, \texttt{RecOrder} \to \texttt{d}]; \quad (61)$$

the same holds for `GenerateRecurrence`; see `In[10]`. Similarly as above, removing the option $\texttt{OneShiftIn} \to \texttt{m}$ gives only a recurrence in n; see `In[9]`. Note that in this case our algorithm reduces to the former version of Sigma described in [26, 29]. ♢

One of the key steps in our computer algebra proof [5] of the TSPP-Theorem [33] is the derivation of a recurrence in i for the definite triple sum

$$S(n,i) = \sum_{k=0}^{2n} \binom{i+k-3}{i-2} T(n,k) \quad (62)$$

where the double sum $T(n,k)$ defined in (20) satisfies the recurrence `In[2]`. With Sigma this can be easily achieved by setting up the summation problem

In[14]:= $\mathbf{mySum} = \sum_{\mathbf{k=0}}^{\mathbf{2n}} \binom{\mathbf{-3+i+k}}{\mathbf{-2+i}} \mathbf{T[k]};$

and calling the Sigma-function:

In[15]:= $\mathbf{GenerateRecurrence}[\mathbf{mySum}, \mathbf{i}, \{\mathbf{recDS}, \mathbf{T[k]}\}, \mathbf{FiniteSupport} \to \mathbf{True}]$

Out[15]= $\{-(2+i+i^2)(-1+i+2n)(i-2(1+n))\,\text{SUM}[i]+$
$(3+i)(-2+2i-i^2+i^3+2n+4n^2)\,\text{SUM}[1+i]+$
$(-3+i)(2+2i+i^2+i^3-2n-4n^2)\,\text{SUM}[2+i]-$
$(2-i+i^2)(1+i-2n)(2+i+2n)\,\text{SUM}[3+i]==0\}$

With the underlying creative telescoping solution a rigorous correctness proof is given in [4, Remark 7] which is similar to the proof in [5, Subsection 5.3].

Finally we illustrate Example 3 by deriving a recurrence for the sum

In[16]:= $\mathbf{mySum} = \sum_{\mathbf{k=1}}^{\mathbf{n}} \binom{\mathbf{n}}{\mathbf{k}} \mathbf{T[k]};$

where `T[k]` is defined by the recurrence relation

In[17]:= $\mathbf{recT} = \mathbf{3\,(1+(1+k)\,H_k)\,(3+2\,k+(2+3\,k+k^2)\,H_k)\,T[k]+4\,(1+k)\,H_k\,(3+2\,k+}$
$\mathbf{(2+3\,k+k^2)\,H_k)\,T[1+k]+(1+k)\,(2+k)\,H_k\,(1+(1+k)\,H_k)\,T[2+k]==0};$

and its initial values `T[1]` and [2]. More precisely, we apply our creative telescoping algorithm, see Example 3, with respect to the underlying difference field, see Example 9, and obtain the recurrence relation:

In[18]:= $\mathbf{GenerateRecurrence}[\mathbf{mySum}, \mathbf{n}, \{\mathbf{recT}, \mathbf{T[k]}\}]$

Out[18]= $\{12\ n\ (1+n)^2\ \mathtt{SUM}[n] + 6\ n\ (1+n)\ (3+2\ n)\ \mathtt{SUM}[1+n] + 3\ n\ (1+n)\ (2+n)\ \mathtt{SUM}[2+n] ==$
$3\ (6+22\ n+13\ n^2)\ \mathtt{T}[1] + 2\ (2+7\ n+4\ n^2)\ \mathtt{T}[2]\}$

This finally allows us to discover identity (24) by using the tool box of Sigma described in [29].

References

[1] 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.

[2] 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.

[3] S.A. Abramov and M. van Hoeij. Integration of solutions of linear functional equations. *Integral Transform. Spec. Funct.*, 8(1-2):3–12, 1999.

[4] G.E. Andrews, P. Paule, and C. Schneider. Plane partitions VI: Stembridge's TSPP Theorem — a detailed algorithmic proof. Technical Report 04-08, RISC-Linz, J. Kepler University, 2004.

[5] G.E. Andrews, P. Paule, and C. Schneider. Plane partitions VI: Stembridge's TSPP Theorem. *To appear in the Dave Robbins memorial issue of Advances in Applied Math.*, 2005.

[6] M. Bronstein. On solutions of linear ordinary difference equations in their coefficient field. *J. Symbolic Comput.*, 29(6):841–877, June 2000.

[7] F. Chyzak. Groebner bases, symbolic summation and symbolic integration. In B. Buchberger and F. Winkler, editors, *Groebner Bases and Applications*, pages 32–60. Cambridge University Press, 1998. Proceedings of the Conference 33 Years of Gröbner Bases.

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

[9] F. Chyzak and B. Salvy. Non-commutative elimination in ore algebras proves multivariate identities. *J. Symbolic Comput.*, 26(2):187–227, 1998.

[10] K. Driver, H. Prodinger, C. Schneider, and A. Weideman. Padé approximations to the logarithm II: Identities, recurrences, and symbolic computation. *To appear in Ramanujan Journal*, 2005.

[11] K. Driver, H. Prodinger, C. Schneider, and A. Weideman. Padé approximations to the logarithm III: Alternative methods and additional results. *To appear in Ramanujan Journal*, 2005.

[12] J.C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner basis by change of ordering. *J. Symbolic Comput.*, 16(4):329–344, October 1993.

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

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

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

[16] M. Karr. Theory of summation in finite terms. *J. Symbolic Comput.*, 1:303–315, 1985.

[17] 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.

[18] P. Paule. Contiguous relations and creative telescoping. *Preprint*, 2005.

[19] P. Paule and A. Riese. A Mathematica q-analogue of Zeilberger's algorithm based on an algebraically motivated aproach to q-hypergeometric telescoping. In M. Ismail and M. Rahman, editors, *Special Functions, q-Series and Related Topics*, volume 14, pages 179–210. Fields Institute Toronto, AMS, 1997.

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

[21] P. Paule and C. Schneider. Creative telescoping for hypergeometric double sums. *Preprint*, 2005.

[22] 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.

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

[24] A. Riese and B. Zimmermann. Randomization speeds up hypergeometric summation. *Preprint*, 2005.

[25] C. Schneider. An implementation of Karr's summation algorithm in Mathematica. *Sém. Lothar. Combin.*, S43b:1–10, 2000.

[26] C. Schneider. Symbolic summation in difference fields. Technical Report 01-17, RISC-Linz, J. Kepler University, November 2001. PhD Thesis.

[27] C. Schneider. A collection of denominator bounds to solve parameterized linear difference equations in ΠΣ-extensions. In D. Petcu, V. Negru, D. Zaharie, and T. Jebelean, editors, *Proc. SYNASC04, 6th Internat. Symposium on Symbolic and Numeric Algorithms for Scientific Computation*, pages 269–282. Mirton Publishing, 2004.

[28] C. Schneider. Solving parameterized linear difference equations in terms of indefinite nested sums and products. SFB-Report 2004-29, J. Kepler University, Linz, 2004.

[29] C. Schneider. The summation package Sigma: Underlying principles and a rhombus tiling application. *Discrete Math. Theor. Comput. Sci.*, 6(2):365–386, 2004.

[30] C. Schneider. Symbolic summation with single-nested sum extensions. In J. Gutierrez, editor, *Proc. ISSAC'04*, pages 282–289. ACM Press, 2004.

[31] C. Schneider. Degree bounds to find polynomial solutions of parameterized linear difference equations in ΠΣ-fields. *To appear in Appl. Algebra Engrg. Comm. Comput.*, 2005.

[32] C. Schneider. Product representations in ΠΣ-fields. *Annals of Combinatorics*, 9(1):75–99, 2005.

[33] J. Stembridge. The enumeration of totally symmetric plane partitions. *Advances in Math.*, 111:227–243, 1995.

[34] P. Verbaeten. The automatic construction of pure recurrence equations. *ACM-SIGSAM Bulletin*, 8:96–98, 1974.

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

[36] H. Wilf and D. Zeilberger. An algorithmic proof theory for hypergeometric (ordinary and "q") multi-sum/integral identities. *Invent. Math.*, 108:575–633, 1992.

[37] D. Zeilberger. A fast algorithm for proving terminating hypergeometric identities. *Discrete Math.*, 80(2):207–211, 1990.

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

(C. Schneider) Research Institute for Symbolic Computation, Johannes Kepler University Linz, Altenberger Str. 69, A–4040 Linz, Austria

E-mail address: Carsten.Schneider@risc.uni-linz.ac.at