RESEARCH INSTITUTE FOR SYMBOLIC COMPUTATION

JOHANNES KEPLER UNIVERSITY LINZ

Term Algebras, Canonical Representations and Difference Ring Theory for Symbolic Summation

C. Schneider

February 2021

RISC Report Series No. 21-03

Available at https://doi.org/10.35011/risc.21-03

JOHANNES KEPLER UNIVERSITY LINZ
Altenberger Str. 69
4040 Linz, Austria
www.jku.at
DVR 0093696

Term Algebras, Canonical Representations and Difference Ring Theory for Symbolic Summation

Carsten Schneider*

Abstract A general overview of the existing difference ring theory for symbolic summation is given. Special emphasis is put on the user interface: the translation and back translation of the corresponding representations within the term algebra and the formal difference ring setting. In particular, canonical (unique) representations and their refinements in the introduced term algebra are explored by utilizing the available difference ring theory. Based on that, precise input-output specifications of the available tools of the summation package `Sigma` are provided.

1 Introduction

In the last 40 years exciting results have been accomplished in symbolic summation as elaborated, e.g., in [18, 19, 23, 24, 26, 29, 31, 49, 52, 53, 55, 56, 61, 63, 65–67, 72, 74, 85, 86, 89, 91, 92, 96, 97, 107, 113, 115–117, 127, 129–131] that will be sketched in more details below. In most cases, symbolic summation can be subsumed by the following problem description: given an algorithm that computes/represents a sequence, find a simpler algorithm that computes/represents (from a certain point on) the same sequence. Based on the context of a given problem, simpler can have different meanings: e.g., the output algorithm can be represented uniquely (by a canonical form in the sense of [50]), it might be computed more efficiently, or it can be formulated in terms of certain classes of special functions.

Often symbolic summation is subdivided in the following summation paradigms.

Carsten Schneider
Johannes Kepler University Linz, Research Institute for Symbolic Computation, A-4040 Linz, Austria, e-mail: `Carsten.Schneider@risc.jku.at`

* This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska–Curie grant agreement No. 764850, SAGEX and from the Austrian Science Foundation (FWF) grant SFB F50 (F5005-N15) in the framework of the Special Research Program "Algorithmic and Enumerative Combinatorics".

• *Telescoping:* Given an algorithm $F(k)$ that computes a sequence, find an algorithm $G(k)$, that is not more complicated than $F(k)$, such that

$$F(k) = G(k\quad 1) - G(k) \tag{1}$$

holds for all $k \in \mathbb{Z}_{\geq 0}$ with $k \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$. Then summing this equation over k from δ to n yields a simpler way to compute $S(n) = \sum_{k=\delta}^{n} F(k)$, namely

$$\sum_{k=\delta}^{n} F(k) = G(n\quad 1) - G(\delta). \tag{2}$$

• *Zeilberger's creative telescoping [131]:* Given an algorithm $F(n,k)$ that computes a bivariate sequence, find an algorithm $G(n,k)$, that is not more complicated than $F(n,k)$ and algorithms $c_0(n),\dots,c_d(n)$ (for univariate sequences), such that

$$c_0(n)\,F(n,k)\quad c_1(n)\,F(n\quad 1,k)\quad \cdots\quad c_d(n)\,F(n\quad d,k) = G(n,k\quad 1) - G(n,k) \tag{3}$$

holds for all $n,k \in \mathbb{Z}_{\geq 0}$ with $n,k \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$. Then summing this equation over k from δ to n yields for the definite sum $S(n) = \sum_{k=\delta}^{n} F(n,k)$ the recurrence

$$c_0(n)\,S(n)\quad c_1(n)\,S(n\quad 1)\quad \cdots\quad c_d(n)\,S(n\quad d) = H(n) \tag{4}$$

with $H(n) = G(n\quad 1) - G(\delta)\quad \sum_{i=1}^{d} c_i(n) \sum_{j=1}^{i} F(n\quad i, n\quad j)$. In many cases $H(n)$ collapses to a rather simple "algorithm" and thus (4) yields (together with d initial values and the assumption that $c_d(n)$ is nonzero for $n \geq \delta$) an efficient algorithm to compute the sequence $(S(n))_{n\geq\delta}$.

• *Recurrence solving:* Given a recurrence of the form (4) where the algorithms $c_0(n),\dots,c_d(n)$ and $H(n)$ can be given by expressions in terms of certain classes of special functions (that can be evaluated accordingly) and given d initial values, say $S(\delta), S(\delta\quad 1),\dots,S(\delta\quad d-1)$ which determines the sequence $(S(n))_{n\geq\delta}$, find an expression that computes the sequence $(S(n))_{n\geq\delta}$ in terms of the same class of special functions or an appropriate extension of it.

We emphasize that each of the above summation paradigms are strongly interwoven (as illustrated, e.g., in the book [91]) and they often yield a strong toolbox by combining them in a non-trivial way.

Another natural classification of symbolic summation is based on the input class of algorithms and the focus how they can be formally represented. In most cases they are either given by evaluable expressions in terms of sums/products or linear recurrences accompanied with initial values that uniquely determine/enable one to calculate the underlying sequences. The first breakthrough in this regard has been carried out by Abramov [18, 19] to solve the telescoping problem for a rational function $F(x) \in \mathbb{K}(x)$ and to find all rational solutions of $\mathbb{K}(x)$ of given linear recurrence (4) with $c_i(x), H(x) \in \mathbb{K}(x)$. In particular, Gosper's telescoping algorithm [61] for hypergeometric products $F(n) = \prod_{k=l}^{n} H(k)$ with $H(x) \in \mathbb{K}(x)$ and Zeilberger's extension to definite sums via his creative telescoping paradigm [52, 53, 85, 89, 91, 131] made symbolic summation highly popular in many areas of sciences. In particular, the interplay with M. Petkovšek's algorithm Hyper [90] or van Hoeij's improvements [92] to find all hypergeometric product solutions enables one to simplify definite hypergeometric products to expressions given in terms of hypergeometric products. More generally, one can use these solvers as subroutines to hunt for all d'Alembertian solutions [24, 26] (solutions that are expressible in terms of indefinite nested sums defined over hypergeometric products) and Liouvillian solutions [63] (incorporating in addition the interlacing operator). This successful story has been pushed forward for indefinite and definite summation problems in terms of q-hypergeometric products and their mixed version [23, 31, 86]. Further generalizations opened up substantially the class of applications, like the holonomic approach [55, 74, 130] dealing with objects that can be described by recurrence systems or the multi-summation approach of (q–)hypergeometric products [29, 127, 129]. Even non-holonomic summation problems [56, 67, 72] involving, e.g., Stirling numbers, can be treated nowadays automatically.

In the following we will focus on the difference ring/field approach. It has been initiated by Karr's telescoping algorithm [65, 66] in -fields which can be considered as the discrete analog of Risch's indefinite integration algorithm [48, 94]. This pioneering work has been explored further in [49, 96, 97, 107] and has been pushed forward to a general summation theory in the setting of R -ring extensions [113, 115–117] which is the driving engine of the summation package `Sigma` [106, 111]. In this setting, one cannot only deal with expressions in terms of (q–)hypergeometric products and their mixed versions, but in terms of sums and products that are indefinite nested (that, depending on the ring or field setting, can appear also in the denominator). In particular, it covers a significant class of special functions that arise frequently, e.g., within the calculation of (massive) 2-loop and 3-loop Feynman integrals: harmonic sums [40, 126], generalized harmonic sums [15, 81], cyclotomic sums [14] and binomial sums [9, 58, 128].

Internally, the following construction is performed in `Sigma`.

1. Rephrase the expression in terms of nested sums and products in an appropriate difference ring (built by -field and R -ring extensions).
2. Solve the summation paradigms (given above) in this formal difference ring.
3. Rephrase the found solution from the difference ring to term algebra setting.

The goal of this article is two-fold. First, we will present the existing algorithms in the difference ring setting (step 2) that have been implemented in large part within `Sigma`. In particular, we will summarize the available parameterized telescoping algorithms [99, 101, 105, 107–110, 114] (containing telescoping/creative telescoping as special cases), the multiplicative version of telescoping for the representation of products [25, 51, 83, 84, 103, 113, 119] and recurrence solving algorithms [22, 49, 80, 97, 98, 100, 104] which generalize many contributions of the literature mentioned above. In addition, we will comment on further enhancements in order to treat new classes of summation objects, like unspecified sequences [69, 70, 88] and radical objects [71], or to combine the difference field/ring approach with the holonomic approach in [43, 102] yielding a new toolbox for multi-summation.

Besides these difference ring algorithms and the underlying difference ring theory (step 2), the translation mechanism between the summation objects and the formal representation (step 1 and 3) will be elaborated in detail. In particular, the summation package `Sigma` benefits heavily on this stable toolbox: the user can define expressions in terms of symbolic sums and products in a term algebra and obtains simplifications of the expressions by executing the rather techincal difference ring/field machinery in the background. However, rigorous input/output specifications on the sum-product level are missing: many of the properties that one can extract on the formal level (step 2) are not properly carried over to the user level. The second main contribution of the article will contribute to close this gap. In particular, inspired by [82] and utilizing ideas from [108, 113, 125] we will show that the difference ring theory implies a canonical simplification in the sense of [50]. We can write the sums and product in a σ-reduced basis (see Definition 4) such that two expressions evaluate to the same sequence if and only if they are syntactically equal.

In Section 2 we will define a term algebra in which we will represent our sequences in terms of indefinite nested sums and products. In particular, we will introduce one of the main features of `Sigma` given in Problem `SigmaReduce`: one can represent the expressions of our term algebra in canonical form. In Section 3 we will elaborate how this distinguished representation can be accomplished by exploiting the difference ring theory of R -extensions. In Section 4 we will make this construction precise by using the existing difference ring algorithms. In particular, we will concentrate on refined simplifications, like finding expressions with minimal nesting depth. Finally, we are in the position to specify in Section 5 the above introduced summation paradigms of `Sigma` within the term algebra level. In Section 6 we present the main applications how the presented algorithms can be utilized for the evaluation of Feynman integrals. We conclude the article in Section 7.

2 The term algebra SumProd$(\mathbb{G})$

Inspired by [82] we will refine the construction from [110] to introduce a term algebra for a big class of indefinite nested sums and products.

The basis of our construction (see also [31]) will be the rational function field extension $\mathbb{K} = K(q_1, \dots, q_v)$ over a field K and on top of it the rational function field extension $\mathbb{G}_m := \mathbb{K}(x, x_1, \dots, x_v)$ over $\mathbb{K}$. For any element $f = \frac{p}{q} \in \mathbb{G}_m$ with $p, q \in \mathbb{K}[x, x_1, \dots, x_v]$ where q 0 and p, q being coprime we define

$$\mathrm{ev}(f, k) = \begin{cases} 0 & \text{if } q(k, q_1^k, \dots, q_v^k) = 0 \\ \frac{p(k, q_1^k, \dots, q_v^k)}{q(k, q_1^k, \dots, q_v^k)} & \text{if } q(k, q_1^k, \dots, q_v^k) \quad 0. \end{cases} \tag{5}$$

Note that there is a $\delta \in \mathbb{Z}_{\geq 0}$ with $q(k, q_1^k, \dots, q_v^k)$ 0 for all $k \in \mathbb{Z}_{\geq 0}$ with $k \geq \delta$; for an algorithm that determines δ if one can factorize polynomials over K see [31, Sec. 3.2]. We define $L(f)$ to be the minimal value

$\delta \in \mathbb{Z}_{\geq 0}$ such that $q(k, q_1^k, \dots, q_v^k) \quad 0$ holds for all $k \geq \delta$; further, we define $Z(f) = \max(L(p), L(q))$ for $f \quad 0$. Later we will call $L : \mathbb{G}_m \to \mathbb{Z}_{\geq 0}$ also an *o-function* and $Z : \mathbb{G}_m^* \to \mathbb{Z}_{\geq 0}$ a *z-function*. $\mathbb{G}_m = \mathbb{K}(x, x_1, \dots, x_v)$ represents the *multibasic mixed sequences*. The special cases $\mathbb{G}_r = \mathbb{K}(x)$ and $\mathbb{G}_b = \mathbb{K}(x_1, \dots, x_v)$ represent the *rational* and the *multi-basic sequences*, respectively. If not specified further, $\mathbb{G}$ will stand for one of the three cases $\mathbb{G}_m$, $\mathbb{G}_r$ or $\mathbb{G}_b$.

Now we extend $\mathbb{G}$ to expressions $\mathrm{SumProd}(\mathbb{G})$ in terms of indefinite nested sums defined over indefinite nested products. For the set of non-trivial roots of unity

$$\mathcal{R} = \{r \in \mathbb{K} \setminus \{1\} \mid r \text{ is a root of unity}\}$$

we introduce the function $\mathrm{ord} : \mathcal{R} \to \mathbb{Z}_{\geq 1}$ with $\mathrm{ord}(r) = \min\{n \in \mathbb{Z}_{\geq 1} \mid r^n = 1\}$. Let $\owedge$, $\oplus$, $\odot$, Sum, Prod be operations with the signatures

$$\begin{array}{lll}
\owedge : & \mathrm{SumProd}(\mathbb{G}) \times \mathbb{Z} & \to \mathrm{SumProd}(\mathbb{G}) \\
\oplus : & \mathrm{SumProd}(\mathbb{G}) \times \mathrm{SumProd}(\mathbb{G}) & \to \mathrm{SumProd}(\mathbb{G}) \\
\odot : & \mathrm{SumProd}(\mathbb{G}) \times \mathrm{SumProd}(\mathbb{G}) & \to \mathrm{SumProd}(\mathbb{G}) \\
\mathrm{Sum} : & \mathbb{Z}_{\geq 0} \times \mathrm{SumProd}(\mathbb{G}) & \to \mathrm{SumProd}(\mathbb{G}) \\
\mathrm{Prod} : & \mathbb{Z}_{\geq 0} \times \mathrm{SumProd}(\mathbb{G}) & \to \mathrm{SumProd}(\mathbb{G}) \\
\mathrm{RPow} : & \mathcal{R} & \to \mathrm{SumProd}(\mathbb{G}).
\end{array}$$

In the following we write $\owedge$, $\oplus$ and $\odot$ in infix notation, and Sum and Prod in prefix notation. Further, for $(\dots((f_1 \square f_2) \square f_3) \square \dots \square f_r)$ with $\square \in \{\odot, \oplus\}$ and $f_1, \dots, f_r \in \mathrm{SumProd}(\mathbb{G})$ we write $f_1 \square f_2 \square f_3 \square \dots \square f_r$.

More precisely, we define the following chain of set inclusions:

$$\begin{array}{ccccccccc}
 & & & & \mathrm{Prod}_1(\mathbb{G}) & \subset & \mathrm{SumProd}_1(\mathbb{G}) & & \text{expressions with single nested products} \\
 & & & & \cap & & \cap & & \\
\mathbb{G}^* & \subset & \mathrm{Prod}^*(\mathbb{G}) & \subset & \mathrm{Prod}(\mathbb{G}) & \subset & \mathrm{SumProd}(\mathbb{G}) & & \text{expressions with nested products} \\
 & & \text{power products in products} & & \text{expressions in products} & & \text{expressions in sums and products.} & &
\end{array} \qquad (6)$$

Here we start with the *set of power products of nested products* $\mathit{Prod}^*(\mathbb{G})$ which is the smallest set that contains 1 with the following properties:

1. If $r \in \mathcal{R}$ then $\mathrm{RPow}(r) \in \mathrm{Prod}^*(\mathbb{G})$.
2. If $p \in \mathrm{Prod}^*(\mathbb{G})$, $f \in \mathbb{G}^*$, $l \in \mathbb{Z}_{\geq 0}$ with $l \geq Z(f)$ then $\mathrm{Prod}(l, f \odot p) \in \mathrm{Prod}^*(\mathbb{G})$.
3. If $p, q \in \mathrm{Prod}^*(\mathbb{G})$ then $p \odot q \in \mathrm{Prod}^*(\mathbb{G})$.
4. If $p \in \mathrm{Prod}^*(\mathbb{G})$ and $z \in \mathbb{Z} \setminus \{0\}$ then $p \owedge z \in \mathrm{Prod}^*(\mathbb{G})$.

Later we will also use the sets

$$\begin{aligned}
(\mathbb{G}) &= \{\mathrm{RPow}(r) \mid r \in \mathcal{R}\} \cup \{\mathrm{Prod}(l, f \odot p) \mid l, f, p \text{ as given in item 2}\}, \\
{}_1(\mathbb{G}) &= \{\mathrm{RPow}(r) \mid r \in \mathcal{R}\} \cup \{\mathrm{Prod}(l, f) \mid f \in \mathbb{G}, l \in \mathbb{Z}_{\geq 0}\} \subset \quad (\mathbb{G})
\end{aligned}$$

For a ring $\mathbb{A}$ we denote by $\mathbb{A}^*$ the set of units. If $\mathbb{A}$ is a field, this means $\mathbb{A}^* = \mathbb{A} \setminus \{0\}$.

We also write p instead of $f \odot p$ if $f = 1$; similarly we write f instead of $f \odot p$ if $p = 1$.

where $(\mathbb{G})$ and ${}_1(\mathbb{G})$ contains all nested and single nested products, respectively.

Example 1 In $\text{Prod}^*(\mathbb{G})$ with $\mathbb{G} = \mathbb{Q}(q_1)(x, x_1)$ we get, e.g.,

$$P = x \otimes (\underbrace{\text{Prod}(\text{Prod}(1,x)^{\circledwedge}(-2),x)}_{\in\ (\mathbb{G})}{}^{\circledwedge}2) \otimes \underbrace{\text{Prod}(1, \tfrac{x_1\ x_1^2}{x})}_{\in\ {}_1(\mathbb{G})} \otimes \underbrace{\text{RPow}(-1)}_{{}_1(\mathbb{G})} \in \text{Prod}^*(\mathbb{G}).$$

Finally, we define *SumProd*$(\mathbb{G})$ as the smallest set containing $\{0\} \cup \text{Prod}^*(\mathbb{G})$ with the following properties:

1. For all $f, g \in \text{SumProd}(\mathbb{G})$ we have $f \oplus g \in \text{SumProd}(\mathbb{G})$.
2. For all $f, g \in \text{SumProd}(\mathbb{G})$ we have $f \odot g \in \text{SumProd}(\mathbb{G})$.
3. For all $f \in \text{SumProd}(\mathbb{G})$ and $k \in \mathbb{Z}_{\geq 1}$ we have $f^{\circledwedge}k \in \text{SumProd}(\mathbb{G})$.
4. For all $f \in \text{SumProd}(\mathbb{G})$ and $l \in \mathbb{Z}_{\geq 0}$ we have $\text{Sum}(l, f) \in \text{SumProd}(\mathbb{G})$.

$\text{SumProd}(\mathbb{G})$ is also called the *set of expressions in terms of nested sums over nested products*. In addition, we define the following subsets:

1. the *set Prod*$(\mathbb{G})$ *of expressions in terms of nested products (over* $\mathbb{G}$*)*, i.e., all elements from $\text{SumProd}(\mathbb{G})$ which are free of sums;
2. the *set Prod*$_1(\mathbb{G})$ *of expressions in terms of depth-1 products (over* $\mathbb{G}$*)*, i.e., all elements from $\text{Prod}(\mathbb{G})$ where the arising products are taken from ${}_1(\mathbb{G})$;
3. the *set Sum*$(\mathbb{G})$ *of expressions in terms of nested sums (over* $\mathbb{G}$*)*, i.e., all elements from $\text{SumProd}(\mathbb{G})$ where no products appear;
4. the *set SumProd*$_1(\mathbb{G})$ *of expressions in terms of nested sums over depth-1 products (over* $\mathbb{G}$*)*, i.e., all elements from $\text{SumProd}(\mathbb{G})$ with products taken from ${}_1(\mathbb{G})$.

In other words, besides the chain of set inclusions given in (6) we also get

$$\text{Sum}(\mathbb{G}) \subset \text{SumProd}_1(\mathbb{G}) \subset \text{SumProd}(\mathbb{G}).$$

Furthermore, we introduce the *set of nested sums over nested products* given by

$$(\mathbb{G}) = \{\text{Sum}(l, f) \mid l \in \mathbb{Z}_{\geq 0} \text{ and } f \in \text{SumProd}(\mathbb{G})\},$$

and the *set of nested sums over single nested products* given by

$${}_1(\mathbb{G}) = \{\text{Sum}(l, f) \mid l \in \mathbb{Z}_{\geq 0} \text{ and } f \in \text{SumProd}_1(\mathbb{G})\}.$$

For convenience we will also introduce the *set* $(\mathbb{G}) = (\mathbb{G}) \cup (\mathbb{G})$ *of nested sums and products* by and the *set* ${}_1(\mathbb{G}) = {}_1(\mathbb{G}) \cup {}_1(\mathbb{G})$ *of nested sums and single-nested products*. In short, we obtain the following chain of sets:

${}_1(\mathbb{G})$	$\subset$	${}_1\ {}_1(\mathbb{G})$	$\supset$	${}_1(\mathbb{G})$	with single nested products
$\cap$		$\cap$		$\cap$	
$(\mathbb{G})$	$\subset$	$(\mathbb{G})$	$\supset$	$(\mathbb{G})$	with nested products
products		products and sums over products		sums over products	

Example 2 With $\mathbb{G} = \mathbb{K}(x)$ we get, e.g., the following expressions:

$$\begin{aligned}
E_1 &= \mathrm{Sum}(1, \mathrm{Prod}(1, \tfrac{1}{x})) \in \quad {}_1(\mathbb{G}) \subset \mathrm{SumProd}_1(\mathbb{G}),\\
E_2 &= \mathrm{Sum}(1, \tfrac{1}{x \quad 1} \otimes \mathrm{Sum}(1, \tfrac{1}{x^3}) \otimes \mathrm{Sum}(1, \tfrac{1}{x})) \in \quad (\mathbb{G}) \subset \mathrm{Sum}(\mathbb{G}),\\
E_3 &= (E_1 \oplus E_2) \otimes E_1 \in \mathrm{SumProd}_1(\mathbb{G}).
\end{aligned}$$

Finally, we introduce a function ev (a model of the term algebra) which evaluates a given expression of our term algebra to sequence elements. In addition, we also introduce the depth for our expressions. We start with the evaluation function ev : $\mathbb{G} \times \mathbb{Z} \to \mathbb{K}$ given by (5) and the depth function $\mathtt{d} : \mathbb{G} \to \mathbb{Z}_{\geq 0}$ given by

$$\mathtt{d}(f) = \begin{cases} 0 & \text{if } f \in \mathbb{K} \\ 1 & \text{if } f \in \mathbb{G} \setminus \mathbb{K}. \end{cases}$$

Finally, ev and d are extended recursively from $\mathbb{G}$ to ev: $\mathrm{SumProd}(\mathbb{G}) \times \mathbb{Z}_{\geq 0} \to \mathrm{SumProd}(\mathbb{G})$ and $\mathtt{d} : \mathrm{SumProd}(\mathbb{G}) \to \mathbb{Z}_{\geq 0}$ as follows.

1. For $f, g \in \mathrm{SumProd}(\mathbb{G})$ and $l \in \mathbb{Z}$ we set

$$\begin{aligned}
\mathrm{ev}(f^{\wedge}k, n) &:= \mathrm{ev}(f, n)^k, & \mathtt{d}(f^{\wedge}k) &:= \mathtt{d}(f)\\
\mathrm{ev}(f \oplus g, n) &:= \mathrm{ev}(f, n) \quad \mathrm{ev}(g, n), & \mathtt{d}(f \oplus g) &:= \max(\mathtt{d}(f), \mathtt{d}(g))\\
\mathrm{ev}(f \odot g, n) &:= \mathrm{ev}(f, n)\ \mathrm{ev}(g, n) & \mathtt{d}(f \odot g) &:= \max(\mathtt{d}(f), \mathtt{d}(g)),
\end{aligned}$$

2. and for $l \in \mathbb{Z}_{\geq 0}$, $f \in \mathrm{SumProd}(\mathbb{G})$ we define

$$\begin{aligned}
\mathrm{ev}(\mathrm{RPow}(r), n) &:= \prod_{i=l}^{n} r = r^n, & \mathtt{d}(\mathrm{RPow}(r)) &:= 1.\\
\mathrm{ev}(\mathrm{Sum}(l, f), n) &:= \sum_{i=l}^{n} \mathrm{ev}(f, i), & \mathtt{d}(\mathrm{Sum}(l, f)) &:= \mathtt{d}(f) \quad 1\\
\mathrm{ev}(\mathrm{Prod}(l, f), n) &:= \prod_{i=l}^{n} \mathrm{ev}(f, i), & \mathtt{d}(\mathrm{Prod}(l, f)) &:= \mathtt{d}(f) \quad 1.
\end{aligned}$$

Remark 1 (1) Since $\mathrm{ev}(\mathrm{Prod}(r, l), n) = \mathrm{ev}(\mathrm{RPow}(r), n)$, RPow is redundant. But it will be convenient for the treatment of canonical representations (see Definition 3).
(2) Any evaluation of $\mathrm{Prod}^*(\mathbb{G})$ is well defined and nonzero since the lower bounds of the products are set large enough via the z-function.
(3) $\mathrm{SumProd}_1(\mathbb{G}_r)$ covers as special cases generalized/cyclotomic harmonic sums [14, 15, 40, 81, 126] and binomial sums [9, 58, 128].

In a nutshell, ev *applied to* $f \in$ *SumProd*$(\mathbb{G})$ *represents a sequence. In particular, f can be considered as a simple program and* $\mathrm{ev}(f, n)$ *with* $n \in \mathbb{Z}_{\geq 0}$ *executes it (like an interpreter/compiler) yielding the nth entry of the represented sequence.*

Definition 1 For $F \in \mathrm{SumProd}(\mathbb{G})$ and $n \in \mathbb{Z}_{\geq 0}$ we write $F(n) := \mathrm{ev}(F, n)$.

Example 3 For $E_i \in \mathrm{SumProd}(\mathbb{K}(x))$ with $i = 1, 2, 3$ in Ex. 2 we get $\mathtt{d}(E_i) = 3$ and

$$E_1(n) = \mathrm{ev}(E_1, n) = \sum_{k=1}^{n} \prod_{i=1}^{k} i = \sum_{k=1}^{n} k!, \quad E_2(n) = \mathrm{ev}(E_2, n) = \sum_{k=1}^{n} \frac{1}{1 \;\; k} \Big(\sum_{i=1}^{k} \frac{1}{i^3} \Big) \sum_{i=1}^{k} \frac{1}{i}$$

and $E_3(n) = (E_1(n) \quad E_2(n))E_1(n)$. For $P \in \mathrm{SumProd}(\mathbb{K}(x, x_1))$ in Ex. 1 we get

$$P(n) = \mathrm{ev}(P, n) = \Big(\prod_{k=1}^{n} \Big(\prod_{i=1}^{k} i \Big)^{-2} \Big)^2 \Big(\prod_{k=1}^{n} \frac{q^k \quad q^{2k}}{k} \Big) (-1)^n, \quad \mathtt{d}(P) = 3.$$

Example 4 We show how the expressions of $\mathrm{SumProd}(\mathbb{G})$ with ev are handled in

In[1]:= **<< Sigma.m**

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

Instead of $F = \mathrm{Sum}(1, \frac{1}{x})$ with $F(n) = \mathrm{ev}(F, n) = \sum_{k=1}^{n} \frac{1}{k}$ we introduce the sum by

In[2]:= **F = SigmaSum[$\frac{1}{k}$, {k, 1, n}]**

Out[2]= $\sum_{k=1}^{n} \frac{1}{k}$

where n is kept symbolically. However, if the user replaces n by a concrete integer, say 5, the evaluation mechanism is carried out and we get $F(5) = \mathrm{ev}(F, 5)$:

In[3]:= **F/.n → 5**

Out[3]= $\frac{137}{60}$

Similarly, we can define E_1 from Example 2 as follows:

In[4]:= **E_1 = SigmaSum[SigmaFactorial[k], {k, 1, n}]**

Out[4]= $\sum_{k=1}^{n} k!$

Here `SigmaFactorials` defines the factorials; its full definition is given by:

In[5]:= **GetFullDefinition[E_1]**

Out[5]= $\sum_{k=1}^{n} \prod_{o_1=1}^{k} o_1$

Similarly, one can introduce as shortcuts powers, Pochhammer symbols, binomial coefficients, (generalized) harmonic sums [15] etc. with the function calls `SigmaPower`, `SigmaPower`, `SigmaPochhammer`, `SigmaBinomial` or `S`, respectively; analogously q-versions are available. Together with Ablinger's package `HarmonicSums`, also function calls for cyclotomic sums [14] and binomial sums [9] are available.

In the same fashion, we can define $E_2, E_3 \in \mathrm{SumProd}(\mathbb{Q}(x))$ from Example 2 and $P \in \mathrm{SumProd}(\mathbb{Q}(q)(x, x_1))$ with $q = q_1$ from Example 1 by

In[6]:= **E_2 = SigmaSum[SigmaSum[1/i, {i, 1, k}]SigmaSum[1/i^3, {i, 1, k}]/(k 1), {k, 1, n}]**

Out[6]= $\sum_{k=1}^{n} \frac{\Big(\sum_{i=1}^{k} \frac{1}{i^3} \Big) \sum_{i=1}^{k} \frac{1}{i}}{1 \;\; k}$

In[7]:= **E_3 = (E_1 E_2)E_1**

Out[7]= $\left(\sum_{k=1}^{n} k!\right)\left(\sum_{k=1}^{n} k! \quad \sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^3}\right) \sum_{i=1}^{k} \frac{1}{i}}{1 \quad k}\right)$

In[8]:= **P = SigmaProduct[SigmaProduct[i, {i, 1, k}]$^{-2}$, {k, 1, n}]^2**
SigmaProduct[(SigmaPower[q, k] SigmaPower[q, k]^2)/k, {k, 1, n}]SigmaPower[−1, n]

Out[8]= $\left(\prod_{k=1}^{n}\left(\prod_{i=1}^{k} i\right)^{-2}\right)^{2}\left(\prod_{k=1}^{n} \frac{q^k \quad (q^k)^2}{k}\right)(-1)^n$

Note that within `Sigma` the root of unity product $\mathrm{RPow}(\alpha)$ with $\alpha \in \mathcal{R}$ can be either defined by **`SigmaPower[α,n]`** or **`SigmaProduct[α,{k,1,n}]`**. Whenever α is recognized as an element of $\mathcal{R}$, it is treated as the special product $\mathrm{RPow}(\alpha)$.

Expressions in $\mathrm{SumProd}(\mathbb{G})$ (similarly within Mathematica using `Sigma`) can be written in different ways such that they produce the same sequence. In the remaining part of this section we will elaborate on canonical (unique) representations [50].

In a preprocessing step we can rewrite the expressions to a reduced representation; note that the equivalent definition in the ring setting is given in Definition 10.

Definition 2 An expression $A \in \mathrm{SumProd}(\mathbb{G})$ is in *reduced representation* if

$$A = (f_1 \odot P_1) \oplus (f_2 \odot P_2) \oplus \cdots \oplus (f_r \odot P_r) \tag{7}$$

with $f_i \in \mathbb{G}^*$ and

$$P_i = (a_{i,1} \circledwedge z_{i,1}) \odot (a_{i,2} \circledwedge z_{i,2}) \odot \cdots \odot (a_{i,n_i} \circledwedge z_{i,n_i}) \tag{8}$$

for $1 \le i \le r$ where

- $a_{i,j} = \mathrm{Sum}(l_{i,j}, f_{i,j})$ with $l_{i,j} \in \mathbb{Z}_{\ge 0}$, $f_{i,j} \in \mathrm{SumProd}(\mathbb{G})$ and $z_{i,j} \in \mathbb{Z}_{\ge 1}$,
- $a_{i,j} = \mathrm{Prod}(l_{i,j}, f_{i,j})$ with $l_{i,j} \in \mathbb{Z}_{\ge 0}$, $f_{i,j} \in \mathrm{Prod}^*(\mathbb{G})$ and $z_{i,j} \in \mathbb{Z} \setminus \{0\}$, or
- $a_{i,j} = \mathrm{RPow}(f_{i,j})$ with $f_{i,j} \in \mathcal{R}$ and $1 \le z_{i,j} < \mathrm{ord}(r_{i,j})$

such that the following properties hold:

1. for each $1 \le i \le r$ and $1 \le j < j' < n_i$ we have $a_{i,j} \quad a_{i,j'}$;
2. for each $1 \le i < i' \le r$ with $n_i = n_j$ there does not exist a $\sigma \in S_{n_i}$ with $P_{i'} = (a_{i,\sigma(1)} \circledwedge z_{i,\sigma(1)}) \odot (a_{i,\sigma(2)} \circledwedge z_{i,\sigma(2)}) \odot \cdots \odot (a_{i,\sigma(n_i)} \circledwedge z_{i,\sigma(n_i)})$.

We say that $H \in \mathrm{SumProd}(\mathbb{G})$ is in *sum-product reduced representation* (or in *sum-product reduced form*) if it is in reduced representation and for each $\mathrm{Sum}(l, A)$ and $\mathrm{Prod}(l, A)$ that occur recursively in H the following holds: A is in reduced representation as given in (7), $l \ge \max(L(f_1), \ldots, L(f_r))$ (i.e. the first case of (5) is avoided during evaluations) and the lower bound l is greater than or equal to the lower bounds of the sums and products inside of A.

Example 5 In `Sigma` the reduced representation of E_3 is calculated with the call

In[9]:= **CollectProdSum[E$_3$, 3]**

Out[9]= $\left(\sum_{k=1}^{n} k!\right)^{2} \quad \left(\sum_{k=1}^{n} k!\right) \sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^3}\right) \sum_{i=1}^{k} \frac{1}{i}}{1 \quad k}$

Before we can state one of `Sigma`'s crucial features we need the following definitions.

Definition 3 Let $W \subseteq \quad (\mathbb{G})$. We define *SumProd*$(W, \mathbb{G})$ as the set of elements from SumProd$(\mathbb{G})$ which are in reduced representation and where the arising sums and products are taken from W. More precisely, $A \in$ SumProd(W, G) if and only if it is of the form (7) with (8) where $a_{i,j} \in W$. In the following we seek for a W with the following properties:

- W is called *shift-closed over* $\mathbb{G}$ if for any $A \in$ SumProd$(W, \mathbb{G})$, $s \in \mathbb{Z}$ there are $B \in$ SumProd$(W, \mathbb{G})$ and $\delta \in \mathbb{Z}_{\geq 0}$ such that $A(n \quad s) = B(n)$ holds for all $n \geq \delta$.
- W is called *shift-stable over* $\mathbb{G}$ if for any product or sum in W the multiplicand or summand is built by sums and products from W.
- W is called *canonical reduced over* $\mathbb{G}$ if for any $A, B \in$ SumProd$(W, \mathbb{G})$ with $A(n) = B(n)$ for all $n \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$ the following holds: A and B are the same up to permutations of the operands in $\oplus$ and $\otimes$.

The sum-product reduced form is only a minor simplification, but it will be convenient to connect to the difference ring theory below; see Corollary 1. In Lemma 1 we note further that shift-stability implies shift-closure. In particular, the shift operation can be straightforwardly carried out; the proof will be delivered later on page 22.

Lemma 1 *If a finite set* $W \subset \quad (\mathbb{G})$ *is shift-stable and the elements are in sum-product reduced form , then it is also shift-closed. If* $\mathbb{K}$ *computable then one can compute for* $F \in$ *SumProd*$(W, \mathbb{G})$ *and* $\lambda \in \mathbb{Z}$ *a* $G \in$ *SumProd*$(W, \mathbb{G})$ *such that* $F(n \quad \lambda) = G(n)$ *holds for all* $n \geq \delta$ *for some* δ*. If one can factor polynomials over* $\mathbb{K}$*,* δ *can be determined.*

Based on that observation, we focus on σ-reduced sets which we define as follows.

Definition 4 $W \subseteq \quad (\mathbb{G})$ is called *σ-reduced over* $\mathbb{G}$ if it is canonical reduced, shift-stable and the elements in W are in sum-product reduced form. In particular, $A \in$ SumProd$(W, \mathbb{G})$ is called *σ-reduced (w.r.t. W)* if W is σ-reduced over $\mathbb{G}$.

More precisely, we are interested in the following problem.

Problem SigmaReduce: Compute a σ-reduced representation

Given: $A_1, \ldots, A_u \in$ SumProd$(\mathbb{G})$ with $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$, i,e., $\mathbb{G} = \mathbb{K}(x, x_1, \ldots, x_v)$ or $\mathbb{G} = \mathbb{K}(x_1, \ldots, x_v)$.
Find: a σ-reduced set $W = \{T_1, \ldots, T_e\} \subset \quad (\mathbb{G}')$ in[a] $\mathbb{G}'$ with $B_1 \ldots, B_u \in$ SumProd$(W, \mathbb{G}')$ and $\delta_1, \ldots, \delta_u \in \mathbb{Z}_{\geq 0}$ such that for all $1 \leq i \leq r$ we get

$$A_i(n) = B_i(n) \quad n \geq \delta_i.$$

[a] In general, we might need a larger field $\mathbb{G}' = \mathbb{K}'(x, x_1, \ldots, x_v)$ or $\mathbb{G}' = \mathbb{K}'(x_1, \ldots, x_v)$ where the field $\mathbb{K}$ is extended to $\mathbb{K}'$.

The sum-product form is not necessary, but simplifies the proof given on page 22.

Example 6 Consider the following two expressions from $\mathrm{SumProd}(\mathbb{Q}(x))$:

In[10]:= $\mathbf{A_1 = SigmaSum[SigmaSum[1/i, \{i, 1, k\}]SigmaSum[1/i^3, \{i, 1, k\}]/(k\ \ \ 1), \{k, 1, n\}]}$

Out[10]= $\displaystyle\sum_{k=1}^{n}\frac{\left(\sum_{i=1}^{k}\frac{1}{i^3}\right)\sum_{i=1}^{k}\frac{1}{i}}{1\ \ k}$

In[11]:= $\displaystyle\mathbf{A_2} = \sum_{i=1}^{n}\frac{1}{i^5}-\frac{\sum_{i=1}^{n}\frac{1}{i^4}}{1\ \ n}-\sum_{j=1}^{n}\frac{\sum_{i=1}^{j}\frac{1}{i^4}}{j}-\sum_{j=1}^{n}\frac{\sum_{i=1}^{j}\frac{1}{i^3}}{j^2}\ \ \frac{\sum_{j=1}^{n}\frac{\sum_{i=1}^{j}\frac{1}{i^3}}{j}}{1\ \ n}-\sum_{j=1}^{n}\frac{\sum_{i=1}^{j}\frac{1}{i}}{j^4}\ \ \frac{\sum_{j=1}^{n}\frac{\sum_{i=1}^{j}\frac{1}{i}}{j^3}}{1\ \ n}\ \ \sum_{k=1}^{n}\frac{\sum_{j=1}^{k}\frac{\sum_{i=1}^{j}\frac{1}{i^3}}{j}}{k}\ \ \sum_{k=1}^{n}\frac{\sum_{j=1}^{k}\frac{\sum_{i=1}^{j}\frac{1}{i}}{j^3}}{k};$

Then we solve Problem SigmaReduce by executing:

In[12]:= $\mathbf{\{B_1, B_2\} = SigmaReduce[\{A_1, A_2\}, n]}$

Out[12]= $\displaystyle\left\{\sum_{k=1}^{n}\frac{\left(\sum_{i=1}^{k}\frac{1}{i^3}\right)\sum_{i=1}^{k}\frac{1}{i}}{1\ \ k}, \sum_{k=1}^{n}\frac{\left(\sum_{i=1}^{k}\frac{1}{i^3}\right)\sum_{i=1}^{k}\frac{1}{i}}{1\ \ k}\right\}$

Since $B_1 = B_2$, it follows $A_1 = A_2$. Note that the set W pops up only implicitly. The set of all sums and products in the output, in our case

$$W_0 = \Big\{\sum_{k=1}^{n}\frac{1}{1\ \ k}\Big(\sum_{i=1}^{k}\frac{1}{i^3}\Big)\sum_{i=1}^{k}\frac{1}{i}\Big\}(=\big\{\mathrm{Sum}(1, \tfrac{1}{x\ \ 1}\otimes\mathrm{Sum}(1, \tfrac{1}{x^3})\otimes\mathrm{Sum}(1, \tfrac{1}{x}))\big\})$$

forms a canonical set in which A_1 and A_2 can be represented by B_1 and B_2 respectively. Adjoining in addition all sums and products that arise inside of the elements in W_0 we get $W = \{\sum_{i=1}^{n}\frac{1}{i}, \sum_{i=1}^{n}\frac{1}{i^3}\}\cup W_0$ which is a σ-reduced set. Internally, `SigmaReduce` parses the arising objects from left to right and constructs the underlying σ-reduced set W in which the input expressions can be rephrased.

Reversing the order of the input elements yields the following result:

In[13]:= $\mathbf{\{B_2, B_1\} = SigmaReduce[\{A_2, A_1\}, n]}$

Out[13]= $\displaystyle\Big\{-\Big(\sum_{k=1}^{n}\frac{1}{k^4}\Big)\sum_{k=1}^{n}\frac{1}{k}\ \ \frac{\left(\sum_{k=1}^{n}\frac{1}{k^3}\right)\sum_{k=1}^{n}\frac{1}{k}}{1\ \ n}-\sum_{k=1}^{n}\frac{\sum_{k=1}^{k}\frac{1}{k^3}}{k^2}\ \ \sum_{k=1}^{n}\frac{\sum_{k=1}^{k}\frac{1}{k^4}}{k}\ \ \sum_{k=1}^{n}\frac{\left(\sum_{k=1}^{k}\frac{1}{k^3}\right)\sum_{k=1}^{k}\frac{1}{k}}{k},$

$\displaystyle-\Big(\sum_{k=1}^{n}\frac{1}{k^4}\Big)\sum_{k=1}^{n}\frac{1}{k}\ \ \frac{\left(\sum_{k=1}^{n}\frac{1}{k^3}\right)\sum_{k=1}^{n}\frac{1}{k}}{1\ \ n}-\sum_{k=1}^{n}\frac{\sum_{k=1}^{k}\frac{1}{k^3}}{k^2}\ \ \sum_{k=1}^{n}\frac{\sum_{k=1}^{k}\frac{1}{k^4}}{k}\ \ \sum_{k=1}^{n}\frac{\left(\sum_{k=1}^{k}\frac{1}{k^3}\right)\sum_{k=1}^{k}\frac{1}{k}}{k}\Big\}$

In this case we get the σ-reduced set

$$W = \Big\{\sum_{j=1}^{n}\frac{1}{j^4}, \sum_{j=1}^{n}\frac{1}{j^3}, \sum_{j=1}^{n}\frac{1}{j}, \sum_{j=1}^{n}\frac{\sum_{k=1}^{j}\frac{1}{k^4}}{j}, \sum_{j=1}^{n}\frac{\sum_{k=1}^{j}\frac{1}{k^3}}{j^2}, \sum_{j=1}^{n}\frac{\left(\sum_{k=1}^{j}\frac{1}{k^3}\right)\sum_{k=1}^{j}\frac{1}{k}}{j}\Big\}$$

(expressed in the `Sigma`-language) and since $B_1 = B_2$ we conclude again that $A_1 = A_2$ holds for all $n \geq 0$. To check that $A_1 = A_2$ holds, one can also execute

In[14]:= $\mathbf{SigmaReduce[A_1 - A_2, n]}$

Out[14]= 0

In this instance, $W = \{\}$ is the σ-reduced set in which we can represent $A_1 - A_2 = 0$.

Such a unique representation (up to trivial permutations) immediately gives rise to the following application: One can compare if two expressions A_1 and A_2 evaluate to the same sequences (from a certain point on): simply check if the resulting B_1 and B_2 in $\mathrm{SumProd}(W, \mathbb{G})$ for a σ-reduced W are the same (up to trivial permutations). Alternatively, just check if $A_1 - A_2$ can be reduced to zero. Besides that we will refine the above problem further. E.g., given $A \in \mathrm{SumProd}(\mathbb{G})$, one can find an expression $B \in \mathrm{SumProd}(W, \mathbb{G})$ and $\delta \in \mathbb{Z}_{\geq 0}$ such that $A(n) = B(n)$ holds for all $n \geq \delta$ and such that B is as simple as possible. Here simple can mean that $\mathsf{d}(B)$ is as small as possible. Other aspects might deal with the task to minimize the number of elements in the set W. Finally, we want to emphasize that the above considerations can be generalized such that also unspecified/generic sequences can appear First important steps towards such a summation theory have been elaborated in [88].

As it turns out, the theory of difference rings provides all the techniques to tackle the above problems accordingly. In the next section we introduce all the needed ingredients and will present our main result in Theorem 2 below.

3 The difference ring approach for $\mathrm{SumProd}(\mathbb{G})$

In the following we will rephrase expressions $H \in \mathrm{SumProd}(\mathbb{G})$ to elements h in a formal difference ring. More precisely, we will design

- a ring $\mathbb{A}$ with $\mathbb{A} \supseteq \mathbb{G} \supseteq \mathbb{K}$ in which H can be represented by $h \in \mathbb{A}$;
- an evaluation function $\mathrm{ev} : \mathbb{A} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ such that $H(n) = \mathrm{ev}(h, n)$ holds for sufficiently large $n \in \mathbb{Z}_{\geq 0}$;
- a ring automorphism $\sigma : \mathbb{A} \to \mathbb{A}$ which scopes the shift $H(n \quad 1)$ with $\sigma(h)$.

Example 7 We will rephrase $F = \mathrm{Sum}(1, \frac{1}{x}) \in \mathrm{SumProd}(\mathbb{G}_r)$ with $\mathbb{G}_r = \mathbb{K}(x)$ where $\mathbb{K} = \mathbb{Q}$ in a formal ring. Namely, we take the polynomial ring $\mathbb{A} = \mathbb{G}_r[s] = \mathbb{Q}(x)[s]$ (s transcendental over $\mathbb{G}_r$) and extend $\mathrm{ev} : \mathbb{G}_r \times \mathbb{Z}_{\geq 0} \to \mathbb{Q}$ to $\mathrm{ev}' : \mathbb{A} \times \mathbb{Z}_{\geq 0} \to \mathbb{Q}$ as follows: for $f = \sum_{k=0}^{d} f_k\, s^k$ with $f_k \in \mathbb{G}_r$ we set

$$\mathrm{ev}'(f, n) := \sum_{k=0}^{d} \mathrm{ev}(f_k, n)\, \mathrm{ev}'(s, n)^k \tag{9}$$

with

$$\mathrm{ev}'(s, n) = \sum_{i=1}^{n} \frac{1}{i} =: S_1(n) (= H_n); \tag{10}$$

since ev and ev′ agree on $\mathbb{G}_r$, we do not distinguish them anymore. In particular, for any

$$H = f_0 \oplus (f_1 \otimes (F^{\circledwedge} 1)) \oplus \cdots \oplus (f_d \otimes (F^{\circledwedge} d))$$

with $d \in \mathbb{Z}_{\geq 0}$ and $f_0, \ldots, f_d \in \mathbb{G}_r$ we can take $h = f_0 \quad f_1\, s \quad \cdots \quad f_d\, s^d \in \mathbb{A}$ and get

$$H(n) = \mathrm{ev}(h, n) \quad \forall n \in \mathbb{Z}_{\geq 0}.$$

In addition, we will introduce the shift operator acting on the elements in $\mathbb{A}$. For the field $\mathbb{G}_r$ we simply define the field automorphism $\sigma : \mathbb{G}_r \to \mathbb{G}_r$ with $\sigma(f) = f|_{x\to x\ \ 1}(= f(x\quad 1))$. Moreover, based on the observation that for any $n \in \mathbb{Z}_{\geq 0}$ we have

$$F(n\quad 1) = \sum_{k=1}^{n\ \ 1} \frac{1}{k} = \sum_{k=1}^{n} \frac{1}{k}\quad \frac{1}{n\quad 1},$$

we extend the automorphism $\sigma : \mathbb{G}_r \to \mathbb{G}_r$ to $\sigma' : \mathbb{G}_r[s] \to \mathbb{G}_r[s]$ as follows: for $f = \sum_{k=0}^{d} f_k\, s^k$ with $f_k \in \mathbb{G}_r$ we set

$$\sigma'(f) := \sum_{k=0}^{d} \sigma(f_k)\sigma'(s)^k$$

with $\sigma'(s) = s\quad \frac{1}{x}$; since σ' and σ agree on $\mathbb{G}_r$, we do not distinguish them anymore. We observe that

$$\mathrm{ev}(s, n\quad 1) = \sum_{k=1}^{n\ \ 1} \frac{1}{k} = \sum_{k=1}^{n} \frac{1}{k}\quad \frac{1}{n\quad 1} = \mathrm{ev}(s\quad \tfrac{1}{x\ \ 1}, n) = \mathrm{ev}(\sigma(s), n)$$

holds for all $n \in \mathbb{Z}_{\geq 0}$ and more generally that $\mathrm{ev}(f, n\quad l) = \mathrm{ev}(\sigma^l(f), n)$ holds for all $l \in \mathbb{Z}$ and $n \in \mathbb{Z}_{\geq 0}$ with $n \geq \max(-l, 0)$.

As illustrated in the example above, the following definitions will be relevant.

Definition 5 A *difference ring* is a ring $\mathbb{A}$ equipped with a ring automorphism $\sigma : \mathbb{A} \to \mathbb{A}$ which one also denotes by $(\mathbb{A}, \sigma)$. For such a difference ring $(\mathbb{A}, \sigma)$ and a subfield $\mathbb{K}$ of $\mathbb{A}$ we introduce the following functions.

1. A function $\mathrm{ev}\colon \mathbb{A} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ is called *evaluation function* for $(\mathbb{A}, \sigma)$ if for all $f, g \in \mathbb{A}$ and $c \in \mathbb{K}$ there exists a $\lambda \in \mathbb{Z}_{\geq 0}$ with the following properties:

$$\forall n \geq \lambda : \mathrm{ev}(c, n) = c, \tag{11}$$

$$\forall n \geq \lambda : \mathrm{ev}(f\quad g, n) = \mathrm{ev}(f, n)\quad \mathrm{ev}(g, n), \tag{12}$$

$$\forall n \geq \lambda : \mathrm{ev}(f\, g, n) = \mathrm{ev}(f, n)\ \mathrm{ev}(g, n). \tag{13}$$

   In addition, we require that for all $f \in \mathbb{A}$ and $l \in \mathbb{Z}$ there exists a λ with

$$\forall n \geq \lambda : \mathrm{ev}(\sigma^l(f), n) = \mathrm{ev}(f, n\quad l). \tag{14}$$

2. A function $L\colon \mathbb{A} \to \mathbb{Z}_{\geq 0}$ is called an *operation-function* (in short *o-function*) for $(\mathbb{A}, \sigma)$ and an evaluation function ev if for any $f, g \in \mathbb{A}$ with $\lambda = \max(L(f), L(g))$ the properties (12) and (13) hold and for any $f \in \mathbb{A}$ and $l \in \mathbb{Z}$ with $\lambda = L(f)\quad \max(0, -l)$ property (14) holds.
3. Let G be a subgroup of $\mathbb{A}^*$. $Z\colon G \to \mathbb{Z}_{\geq 0}$ is called a *zero-function* (in short *z-function*) for ev and $\mathbb{G}$ if $\mathrm{ev}(f, n)\quad 0$ holds for any $f \in \mathbb{G}$ and integer $n \geq Z(f)$.

We note that a construction of a map $\text{ev} : \mathbb{A} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ with the properties (11) and (13) is straightforward. It is property (14) that brings in extra complications: the evaluation of the elements in $\mathbb{A}$ must be compatible with the automorphism σ.

In this article we will always start with the following ground field; see [31].

Example 8 Take the rational function field $\mathbb{G}_m := \mathbb{G} = \mathbb{K}(x, x_1, \dots, x_v)$ over $\mathbb{K} = K(q_1, \dots, q_v)$, $v \geq 0$, with the function (5), together with the functions $L : \mathbb{G}_m \to \mathbb{Z}_{\geq 0}$ and $Z\colon \mathbb{G}_m^* \to \mathbb{Z}_{\geq 0}$ from the beginning of Section 2. It is easy to see that $\text{ev} : \mathbb{G}_m \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ satisfies for all $c \in \mathbb{K}$ and $f, g \in \mathbb{G}$ the property (11) for $L(c) = 0$ and the properties (12) and (13) with $\lambda = \max(L(f), L(g))$. Finally, we take the automorphism $\sigma\colon \mathbb{G}_m \to \mathbb{G}_m$ defined by $\sigma|_{\mathbb{K}} = \text{id}$, $\sigma(x) = x \quad 1$ and $\sigma(y_i) = q_i\, y_i$ for $1 \leq i \leq v$. Then one can verify in addition that (14) holds for all $f \in \mathbb{G}_m$ and $l \in \mathbb{Z}$ with $\lambda = \max(-l, L(f))$. Consequently, ev is an evaluation function for $(\mathbb{G}_m, \sigma)$ and L is an o-function for $(\mathbb{G}_m, \sigma)$. In addition, Z is a z-function for ev and $\mathbb{G}_m^*$ by construction. In the following we call $(\mathbb{G}_m, \sigma)$ also a *multibasic mixed difference field*. If $v = 0$, i.e., $\mathbb{G}_r = \mathbb{K}(x) = \mathbb{K}'(x)$, we get the the *rational difference field* $(\mathbb{G}_r, \sigma)$, and if we restrict to $\mathbb{G}_b = \mathbb{K}(x_1, \dots, x_v)$, we get the *multibasic difference field* $(\mathbb{G}_b, \sigma)$.

We continue with the convention from above: if we write $(\mathbb{G}, \sigma)$, then it can be replaced by any of the difference rings $(\mathbb{G}_m, \sigma)$, $(\mathbb{G}_r, \sigma)$ or $(\mathbb{G}_b, \sigma)$.

In the following we seek for such a formal difference ring $(\mathbb{A}, \sigma)$ with a computable evaluation function ev and o-function L in which we can model a finite set of expressions $A_1, \dots, A_u \in \text{SumProd}(\mathbb{G})$ with $a_1, \dots, a_u \in \mathbb{A}$.

Definition 6 Let $F \in \text{SumProd}(\mathbb{G})$ and $(\mathbb{A}, \sigma)$ be a difference ring equation of $(\mathbb{G}, \sigma)$ equipped with an evaluation function ev. We say that $f \in \mathbb{A}$ *models* F if $\text{ev}(f, n) = F(n)$ holds for all $n \geq \lambda$ for some $\lambda \in \mathbb{Z}_{\geq 0}$.

3.1 The naive representation in APS-extensions

As indicated in Example 7 our sum-product expressions will be rephrased in a tower of difference field and ring extensions. We start with the field version which will lead later to -fields [65, 66].

Definition 7 A difference field $(\mathbb{F}, \sigma)$ is called a *PS-field extension* of a difference field $(\mathbb{H}, \sigma)$ if $\mathbb{H} = \mathbb{H}_0 \leq \mathbb{H}_1 \leq \cdots \leq \mathbb{H}_e = \mathbb{F}$ is a tower of field extensions where for all $1 \leq i \leq e$ one of the following holds:

- $\mathbb{H}_i = \mathbb{H}_{i-1}(t_i)$ is a rational function field extension with $\frac{\sigma(t_i)}{t_i} \in (\mathbb{H}_{i-1})^*$ (t_i is called a *P-field monomial*);
- $\mathbb{H}_i = \mathbb{H}_{i-1}(t_i)$ is a rational function extension with $\sigma(t_i) - t_i \in \mathbb{H}_{i-1}$ (t_i is called a *S-field monomial*).

Example 9 Following Example 8, $(\mathbb{G}_m, \sigma)$ with $\mathbb{G}_m = \mathbb{K}(x, x_1, \dots, x_v)$ is a PS-field extension of $(\mathbb{K}, \sigma)$ with the S-field monomial x and the P-monomials $x_1, \dots, x_v$.

Similarly, $(\mathbb{G}_b, \sigma)$ with $\mathbb{G}_b = \mathbb{K}(x_1, \dots, x_v)$ forms a tower of P-field extensions of $(\mathbb{K}, \sigma)$ and $(\mathbb{G}_r, \sigma)$ with $\mathbb{G}_r = \mathbb{K}(x)$ is an S-field extension of $(\mathbb{K}, \sigma)$.

In addition, we will modify the field version to the following ring version (allowing us to model also products over roots of unity).

Definition 8 A difference ring $(\mathbb{E}, \sigma)$ is called an *APS-extension* of a difference ring $(\mathbb{A}, \sigma)$ if $\mathbb{A} = \mathbb{A}_0 \leq \mathbb{A}_1 \leq \dots \leq \mathbb{A}_e = \mathbb{E}$ is a tower of ring extensions where for all $1 \leq i \leq e$ one of the following holds:

- $\mathbb{A}_i = \mathbb{A}_{i-1}[t_i]$ is a ring extension subject to the relation $t_i^{\nu} = 1$ for some $\nu > 1$ where $\frac{\sigma(t_i)}{t_i} \in (\mathbb{A}_{i-1})^*$ is a primitive νth root of unity (t_i is called an *A-monomial*, and ν is called the *order of the A-monomial*);
- $\mathbb{A}_i = \mathbb{A}_{i-1}[t_i, t_i^{-1}]$ is a Laurent polynomial ring extension with $\frac{\sigma(t_i)}{t_i} \in (\mathbb{A}_{i-1})^*$ (t_i is called a *P-monomial*);
- $\mathbb{A}_i = \mathbb{A}_{i-1}[t_i]$ is a polynomial ring extension with $\sigma(t_i) - t_i \in \mathbb{A}_{i-1}$ (t_i is called an *S-monomial*).

Depending on the occurrences of the APS-monomials such an extension is also called *A-/P-/S-/AP-/AS/-/PS-extension*.

Example 10 Take the rational difference ring $(\mathbb{Q}(x), \sigma)$ with $\sigma(x) = x \quad 1$ and $\sigma|_{\mathbb{Q}} = \text{id}$. Then the difference ring $(\mathbb{Q}(x)[s], \sigma)$ with $\sigma(s) = s \quad \frac{1}{x \quad 1}$ defined in Example 7 is an S-extension of $(\mathbb{Q}(x), \sigma)$ and s is an S-monomial over $(\mathbb{Q}(x), \sigma)$.

For the APS-extension $(\mathbb{E}, \sigma)$ of $(\mathbb{A}, \sigma)$ we will also write $\mathbb{E} = \mathbb{A}\langle t_1 \rangle \dots \langle t_e \rangle$. Depending on the case whether t_i with $1 \leq i \leq e$ is an A-monomial, P-monomial or S-monomial, $\mathbb{G}\langle t_i \rangle$ with $\mathbb{G} = \mathbb{A}\langle t_1 \rangle \dots \langle t_{i-1} \rangle$ stands for the algebraic ring extension $\mathbb{G}[t_i]$ with t_i^{ν} for some $\nu > 1$, for the ring of Laurent polynomials $\mathbb{G}[t_1, t_1^{-1}]$ or for the polynomial ring $\mathbb{G}[t_i]$, respectively.

For such a tower of APS-extensions we can use the following lemma iteratively to construct an evaluation function; for the corresponding proofs see [117, Lemma 5.4].

Lemma 2 *Let $(\mathbb{A}, \sigma)$ be a difference ring with a subfield $\mathbb{K} \subseteq \mathbb{A}$ equipped with an evaluation function* ev *and o-function L. Let $(\mathbb{A}\langle t \rangle, \sigma)$ be an APS-extension of $(\mathbb{A}, \sigma)$ with $\sigma(t) = \alpha\, t \quad \beta$ ($\alpha = 1$, $\beta \in \mathbb{A}$ or $\alpha \in \mathbb{A}^*$, $\beta = 0$). Further, suppose that $\text{ev}(\sigma^{-1}(\alpha), n) \quad 0$ for all $n \geq \mu$ for some $\mu \in \mathbb{Z}_{\geq 0}$. Then the following holds.*

1. *Take $l \in \mathbb{Z}_{\geq 0}$ with $l \geq \max(L(\sigma^{-1}(\alpha)), L(\sigma^{-1}(\beta)), \mu)$. If $t^{\lambda} = 1$ for some $\lambda > 1$ (t is an A-monomial), we assume that $l = 1$ holds. Then* ev′ *given by*

$$\text{ev}'(\sum_{i=a}^{b} f_i\, t^i, n) = \sum_{i=a}^{b} \text{ev}(f_i, n)\, \text{ev}'(t, n)^i \quad \forall n \in \mathbb{Z}_{\geq 0} \tag{15}$$

with

$$\text{ev}'(t, n) = \begin{cases} \prod_{i=l}^{n} \text{ev}(\sigma^{-1}(\alpha), i) & \text{if } \sigma(t) = \alpha\, t \\ \sum_{i=l}^{n} \text{ev}(\sigma^{-1}(\beta), i) & \text{if } \sigma(t) = t \quad \beta \end{cases} \tag{16}$$

is an evaluation function for $(\mathbb{A}\langle t\rangle, \sigma)$.

2. *There is an o-function* $L' : \mathbb{A}\langle t\rangle \to \mathbb{Z}_{\geq 0}$ *for* ev' *defined by*

$$L'(f) = \begin{cases} L(f) & \text{if } f \in \mathbb{A}, \\ \max(l-1, L(f_a), \dots, L(f_b)) & \text{if } f = \sum_{i=a}^{b} f_i t^i \quad \mathbb{A}\langle t\rangle \setminus \mathbb{A}. \end{cases} \tag{17}$$

Example 11 In Example 7 we followed precisely the construction (1) of the above lemma to construct for $(\mathbb{Q}(x)[s], \sigma)$ an evaluation function. For this ev we can now apply also the construction (2) to enhance the o function $L : \mathbb{Q}(x) \to \mathbb{Z}_{\geq 0}$ (given in Example 8 with $v = 0$) to $L : \mathbb{Q}(x)[h] \to \mathbb{Z}_{\geq 0}$ by setting $L(f) = \max(L(f_0), \dots, L(f_b))$ for $f = \sum_{i=0}^{b} f_i s^i$.

In general, take a nested APS-extension $(\mathbb{E}, \sigma)$ of our ground field $(\mathbb{G}, \sigma)$ with $\mathbb{E} = \mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle$. Then the main idea is to apply the above lemma iteratively to extend the evaluation function ev from $\mathbb{G}$ to $\mathbb{E}$. However, if one wants to treat, e.g., the next P-monomial t with $\frac{\sigma(t)}{t} = \alpha \in \mathbb{E}^*$, one has to check if there is a $\mu \in \mathbb{Z}_{\geq 0}$ such that $\mathrm{ev}(\sigma^{-1}(\alpha), n)$ 0 holds for all $n \geq \mu$. So far, we are not aware of a general algorithm that can accomplish this task. In order to overcome these difficulties, we will restrict APS-extensions further to a subclass which covers all summation problems that we have encountered in concrete problems so far.

Let G be a multiplicative subgroup of $\mathbb{A}^*$. Following [115, 117] we call

$$\{G\}_{\mathbb{A}}^{\mathbb{E}} := \{h\, t_1^{m_1} \dots t_e^{m_e} \mid f \in G \text{ and } m_i \in \mathbb{Z} \text{ where } m_i = 0 \text{ if } t_i \text{ is an } S\text{-monomial}\}$$

the *simple product group* over G and

$$[G]_{\mathbb{A}}^{\mathbb{E}} := \{h\, t_1^{m_1} \dots t_e^{m_e} \mid f \in G \text{ and } m_i \in \mathbb{Z} \text{ where } m_i = 0 \text{ if } t_i \text{ is an } AS\text{-monomial}\}$$

the *basic product group* over G for the nested APS–extension $(\mathbb{E}, \sigma)$ of $(\mathbb{A}, \sigma)$. Note that we have the chain of subgroups $[G]_{\mathbb{A}}^{\mathbb{E}} \leq \{G\}_{\mathbb{A}}^{\mathbb{E}} \leq \mathbb{E}^*$. In the following we will restrict ourselves to the following subclass of APS-extensions.

Definition 9 Let $(\mathbb{A}, \sigma)$ be a difference ring and let G be a subgroup of $\mathbb{A}^*$. Let $(\mathbb{E}, \sigma)$ be an APS-extension of $(\mathbb{A}, \sigma)$ with $\mathbb{E} = \mathbb{A}\langle t_1\rangle \dots \langle t_e\rangle$.

1. The extension is called G-*basic* if for any P-monomial t_i we have $\frac{\sigma(t_i)}{t_i} \in [G]_{\mathbb{A}}^{\mathbb{A}\langle t_1\rangle \dots \langle t_{i-1}\rangle}$ and for any A-mon. t_i we have $\alpha_i = \frac{\sigma(t_i)}{t_i} \in G$ with $\sigma(\alpha_i) = \alpha_i$.
2. It is called G-*simple* if any AP-monomial t_i we have $\frac{\sigma(t_i)}{t_i} \in \{G\}_{\mathbb{A}}^{\mathbb{A}\langle t_1\rangle \dots \langle t_{i-1}\rangle}$.

If $G = \mathbb{A}^*$, it is also called *basic* (resp. *simple*) instead of $\mathbb{A}^*$-basic (resp. $\mathbb{A}^*$-simple).

By definition any simple APS-extensions is also a basic APS-extension. We will start with the more general setting of simple extensions, but will restrict later mostly to basic extensions. For both cases we can supplement Lemma 2 as follows.

If t is an A-monomial, we have $\mathrm{ev}(t_e\ _1, n) = \alpha^n$.

Lemma 3 *Let $(\mathbb{A}, \sigma)$ be a difference ring with a subfield $\mathbb{K} \subseteq \mathbb{A}$ equipped with an evaluation function* ev *and o-function L. Let G be a subgroup of $\mathbb{A}^*$ and let $(\mathbb{A}\langle t\rangle, \sigma)$ be an APS-extension of $(\mathbb{A}, \sigma)$ with $\sigma(t) = \alpha\, t \quad \beta$ with $\alpha \in G$ and $\beta \in \mathbb{A}$. Suppose that there is in addition a z-function for* ev *and G. Take $l \in \mathbb{Z}_{\geq 0}$ with*

$$l \geq \begin{cases} \max(L(\sigma^{-1}(\alpha)), Z(\sigma^{-1}(\alpha))) & \textit{if } t \textit{ is an } AP\textit{-monomial} \\ L(\sigma^{-1}(\beta)) & \textit{if } t \textit{ is an } S\textit{-monomial}. \end{cases} \tag{18}$$

Then we obtain an evaluation function ev′ *and o-function L' for $(\mathbb{A}\langle t\rangle, \sigma)$ as given in Lemma 2. In addition, we can construct a z-function Z' for $\{G\}_{\mathbb{A}}^{\mathbb{A}\langle t\rangle}$. If* ev*, L and Z are computable,* ev*, L' and Z' are computable.*

Proof For r as defined in (18) the assumptions in Lemma 18 are fulfilled and the ev′ with L' defined in the lemma yield an evaluation function together with an o-function. If t is a S-monomial, $\{G\}_{\mathbb{A}}^{\mathbb{A}\langle t\rangle} = G$ and we can set $Z' := Z$. Otherwise, if t is an AP-monomial, we have $\text{ev}'(t, n) \quad 0$ for all $n \in \mathbb{Z}_{\geq 0}$ by construction. Thus for $f = g\, t^m \in \{G\}_{\mathbb{A}}^{\mathbb{A}\langle t\rangle}$ with $g \in G$ and $m \in \mathbb{Z}$ we have $\text{ev}(f, n) \quad 0$ for all $n \geq Z(g)$. Thus we can define $Z'(f) = Z(g)$. If L and Z are computable, also L' and Z' are computable. In addition, if we can compute ev, then clearly also ev′ is computable.□

More precisely, suppose that we are given a difference ring $(\mathbb{A}, \sigma)$ with a subfield $\mathbb{K} \subseteq \mathbb{A}$ equipped with a (computable) evaluation function $\text{ev} : \mathbb{A} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ together with a (computable) o-function $L : \mathbb{A} \to \mathbb{Z}_{\geq 0}$ and (computable) z-function $Z : \mathbb{A}^* \to \mathbb{Z}_{\geq 0}$. Suppose in addition that we are given a simple APS-extension $(\mathbb{E}, \sigma)$ of $(\mathbb{A}, \sigma)$ with $\mathbb{E} = \mathbb{A}\langle t_1\rangle \dots \langle t_e\rangle$. Then we can apply iteratively Lemmas 2 and 3 and get a (computable) evaluation function $\text{ev} : \mathbb{E} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ together with a (computable) o-function $L : \mathbb{E} \to \mathbb{Z}_{\geq 0}$ and z-function for $\{\mathbb{A}^*\}_{\mathbb{A}}^{\mathbb{A}\langle t_1\rangle \dots \langle t_e\rangle}$; note that $\{\{\mathbb{A}^*\}_{\mathbb{A}}^{\mathbb{H}}\}_{\mathbb{H}}^{\mathbb{H}\langle t_i\rangle} = \{\mathbb{A}^*\}_{\mathbb{A}}^{\mathbb{H}\langle t_i\rangle}$ for all $\mathbb{H} = \mathbb{A}\langle t_1\rangle \dots \langle t_{i-1}\rangle$ with $1 \leq i < e$.

It is natural to define the evaluation function iteratively using Lemma 2 but it is inconvenient to compute the o-function in this iterative fashion. Here the following lemma provides a shortcut for expressions which are given in reduced representation; for the corresponding representation in SumProd($\mathbb{G}$) see Definition 2.

Definition 10 Let $(\mathbb{E}, \sigma)$ be an APS-extension of $(\mathbb{A}, \sigma)$ with $\mathbb{E} = \mathbb{A}\langle t_1\rangle \dots \langle t_e\rangle$. Then we say that $f \in \mathbb{E}$ is in *reduced representation* if it is written in the form

$$f = \sum_{(m_1,\dots,m_e)\in S} f_{(m_1,\dots,m_e)} t_1^{m_1} \dots t_e^{m_e} \tag{19}$$

with $f_{(m_1,\dots,m_e)} \in \mathbb{A}$ and $S \subseteq M_1 \times \dots \times M_e$ finite where

$$M_i = \begin{cases} \{0, \dots, \nu_i - 1\} & \text{if } t_i \text{ is an } A\text{-extension of order } \nu_i, \\ \mathbb{Z} & \text{if } t_i \text{ is a } P\text{-monomial}, \\ \mathbb{Z}_{\geq 0} & \text{if } t_i \text{ is an } S\text{-monomial}. \end{cases}$$

Lemma 4 *Take a difference ring $(\mathbb{A},\sigma)$ with a subfield $\mathbb{K}\subseteq\mathbb{A}$ equipped with an evaluation function* $\mathrm{ev}:\mathbb{A}\times\mathbb{Z}_{\geq 0}\to\mathbb{K}$ *together with an o-function L and z-function Z. Let $(\mathbb{E},\sigma)$ with $\mathbb{E}=\mathbb{A}\langle t_1\rangle\dots\langle t_e\rangle$ be a simple APS-extension of $(\mathbb{A},\sigma)$ and let* ev *be an evaluation function and Z be a z-function (using iteratively Lemmas 2 and 3). Here the $l_i\in\mathbb{Z}_{\geq 0}$ for $1\leq i\leq e$ are the lower bounds of the corresponding sums/products in* (16) *with $t=t_i$. Then for any $f\in\mathbb{E}$ with* (19) *where $f_{(m_1,\dots,m_e)}\in\mathbb{E}$ and $S\subseteq\mathbb{Z}^e$ we have*

$$L(f)=\max(\max_{s\in S}L(f_s),\max_{j\in\sup(f)}l_j-1)$$

where $\sup(f)=\{1\leq j\leq e\mid t_j$ *depends on* $f\}$.

Proof We show the statement by induction on e. If $e=0$, the statement holds trivially. Now suppose that the statement holds for $e\geq 0$ extensions and let $f\in\mathbb{E}\langle t_{e\ 1}\rangle$ with $\mathbb{E}=\mathbb{G}\langle t_1\rangle\dots\langle t_e\rangle$ where $f=\sum_{i=a}^{b}f_i t_{e\ 1}^{i}$ with $f_i\in\mathbb{E}$. If $f\in\mathbb{E}$, then the statement holds by the induction assumption. Otherwise write $f_i=\sum_{(s_1,\dots,s_e)\in S_i}f^{(i)}_{s_1,\dots,s_e}t_1^{s_1}\dots t_e^{s_e}$ with $S_i\subseteq\mathbb{Z}^e$ and $f^{(i)}_s$ for $s\in S_i$ in reduced representation. In particular, we get $f=\sum_{(s_1,\dots,s_{e\ 1})\in S}h_{(s_1\dots,s_{e\ 1})}t_1^{s_1}\dots t_{e\ 1}^{s_{e\ 1}}$ with $h_{(s_1\dots,s_{e\ 1})}=f^{(s_{e\ 1})}_{(s_1,\dots,s_e)}$ and $S=\cup_{a\leq i\leq b}\{(s_1,\dots,s_e,i)\mid(s_1,\dots,s_e)\in S_i\}$. Then by the induction assumption we get $L(f_i)=\max(\max_{s\in S_i}L(f^{(i)}_s),\max_{j\in\sup(f_i)}l_j-1)$. Thus by the definition in (17) we get

$$\begin{aligned}L(f)&=\max(\max_{a\leq i\leq b}L(f_i),l_{e\ 1}-1)\\&=\max(\max_{s\in S_a}L(f^{(a)}_s),\max_{s\in S_{a\ 1}}L(f^{(a\ 1)}_s),\dots,\max_{s\in S_b}L(f^{(b)}_s),\max_{j\in\sup(f)}l_j-1)\\&=\max(\max_{s\in S}L(h_s),\max_{j\in\sup(f)}l_j-1).\end{aligned}$$

□

Given the above constructions, we are now ready to show in Lemmas 5 and 6 given below that the representations in SumProd$(\mathbb{G})$ and in a basic APS-extension are closely related. Their proofs are rather technical (but not very deep). Still we will present all details, since this construction will be crucial for further refinements. This will finally lead to a strategy to solve Problem SigmaReduce.

Lemma 5 *Take the difference field $(\mathbb{G},\sigma)$ with $\mathbb{G}\in\{\mathbb{G}_r,\mathbb{G}_b,\mathbb{G}_m\}$ with the evaluation function* ev*, o-function L and z-function Z from Example 8. Let $(\mathbb{E},\sigma)$ with $\mathbb{E}=\mathbb{G}\langle t_1\rangle\dots\langle t_e\rangle$ be a basic APS-extension of $(\mathbb{G},\sigma)$ and let* ev*, L and Z be extended versions for $(\mathbb{E},\sigma)$ (using Lemmas 2 and 3). Then for each $1\leq i\leq e$ one can construct $T_i\in\ \ (\mathbb{G})$ in sum-product representation form with* $\mathrm{ev}(t_i,n)=T_i(n)$ *for all $n\geq L(t_i)$. In particular, if $f\in\mathbb{E}\setminus\{0\}$, then there is $0\quad F\in SumProd(\{T_1,\dots,T_e\},\mathbb{G})$ with $F(n)=\mathrm{ev}(f,n)$ for all $n\geq L(f)$.*
If $\mathbb{K}$ is computable and polynomials can be factored over $\mathbb{K}$, all components can be computed.

Proof First suppose that we can construct such $T_i\in\ \ (\mathbb{G})$ with $T_i(n)=\mathrm{ev}(t_i,n)$ for all $n\geq L(t_i)$ and $1\leq i\leq e$. Now take $f\in\mathbb{E}$ in reduced representation, i.e., it is given in the form (19) with $S \subseteq \mathbb{Z}^e$. Now replace each $f_{(m_1,\dots,m_e)} \cdot t_1^{m_1} \dots t_e^{m_e}$ by $f_{(m_1,\dots,m_e)} \odot (T_1 \circledwedge m_1) \odot \cdots \odot (T_e \circledwedge m_e)$ and replace   by $\oplus$ in f yielding $F \in \mathrm{SumProd}(\mathbb{G})$ in reduced representation. Then for each $n \geq L(f)$ we get

$$\begin{aligned} \mathrm{ev}(f,n) &= \mathrm{ev}(\sum_{(m_1,\dots,m_e)\in S} f_{(m_1,\dots,m_e)} t_1^{m_1} \dots t_e^{m_e}, n) \\ &= \sum_{(m_1,\dots,m_e)\in S} \mathrm{ev}(f_{(m_1,\dots,m_e)}, n)\, \mathrm{ev}(t_1,n)^{m_1} \dots \mathrm{ev}(t_e,n)^{m_e} \\ &= \sum_{(m_1,\dots,m_e)\in S} \mathrm{ev}(f_{(m_1,\dots,m_e)}, n)\, \mathrm{ev}(T_1,n)^{m_1} \dots \mathrm{ev}(T_e,n)^{m_e} \\ &= \mathrm{ev}(F,n). \end{aligned} \tag{20}$$

Note: if $f\ \ 0$, we can find $(m_1,\dots,m_e) \in S$ with $f_{(m_1,\dots,m_e)} \in \mathbb{G}^*$ which implies that $F\ \ 0$. This shows the second part of statement (1).

Finally we show the existence of the T_i by induction on e. For $e = 0$ nothing has to be shown. Suppose that the statement holds for $e \geq 0$ extensions and consider the APS-monomial $t_{e\ 1}$ over $\mathbb{E} = \mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle$. By assumption we can take $T_i \in \mathrm{SumProd}(\mathbb{G})$ in sum-product representation with $T_i(n) = \mathrm{ev}(t_i,n)$ for all $n \geq L(t_i)$ and $1 \leq i \leq e$. Now consider the APS-monomial $t_{e\ 1}$ with $\sigma(t_{e\ 1}) = \alpha\, t_{e\ 1}\ \ \beta$. By assumption we have (16) (ev′ replaced by ev) with $l \in \mathbb{Z}_{\geq 0}$ where (18) and L is defined by (17) (L' replaced by L). In particular, we have $l \geq \max(L(\sigma^{-1}(\alpha)), L(\sigma^{-1}(\beta)), \mu)$ with $\mu \geq Z(\sigma^{-1}(\alpha))$, and $L(t_{e\ 1}) = l - 1$.

A-monomial case: If $t_{e\ 1}$ is an A-monomial, we have $\sigma(t_{e\ 1}) = \alpha\, t_{e\ 1}$ with $\alpha \in \mathcal{R}$. In particular, we have $\mathrm{ev}(t_{e\ 1}, n) = \alpha^n$. Thus we set $T_{e\ 1} = \mathrm{RPow}(\alpha)$ and get $\mathrm{ev}(t_{e\ 1}, n) = T_{e\ 1}(n)$ for all $n \geq L(t_{e\ 1}) = 0$.

S-monomial case: If $t_{e\ 1}$ is an S-monomial, we have $\sigma(t_{e\ 1}) = t_{e\ 1}\ \ \beta$ with $\beta \in \mathbb{E}$. Now take $f = \sigma^{-1}(\beta)$ in reduced representation. Then by construction $l \geq \max(L(\sigma^{-1}(\beta)), 0) = L(f)$. Further, we can take $F \in \mathrm{SumProd}(\mathbb{G})$ as constructed above with (20) for all $n \geq l \geq L(f)$. Thus for $T_{e\ 1} = \mathrm{Sum}(l, F)$ we get $\mathrm{ev}(t_{e\ 1}, n) = T_{e\ 1}(n)$ for all $n \geq l - 1 = L(t_{e\ 1})$.

P-monomial case: If $t_{e\ 1}$ is a P-monomial, we have $\sigma(t_{e\ 1}) = \alpha\, t_{e\ 1}$ with $\alpha \in [\mathbb{G}]_{\mathbb{G}}^{\mathbb{E}}$, i.e., $\alpha = g\, t_1^{n_1} \dots t_e^{n_e}$ with $g \in \mathbb{G}^*$ and $n_1,\dots,n_e \in \mathbb{Z}$ with $n_i = 0$ if t_i is an AS-monomial. Thus $f = \sigma^{-1}(\alpha) = h\, t_1^{m_1} \dots t_e^{m_e}$ with $h := f_{(m_1,\dots,m_e)} \in \mathbb{G}^*$ and $m_1,\dots,m_e \in \mathbb{Z}$ with $m_i = 0$ if t_i is an AS-monomial. By construction, $l \geq \max(L(f), Z(f)) = \max(L(f), Z(h))$. As above we get $F = h \odot (T_1 \circledwedge m_1) \odot \cdots \odot (T_e \circledwedge m_e) \in \mathrm{SumProd}(\mathbb{G})$ such that $\mathrm{ev}(f,n) = F(n)$ holds for all $n \geq L(f)$ and $\mathrm{ev}(f,n) = F(n)\ \ \ 0$ for all $n \geq l$. Thus for $T_{e\ 1} = \mathrm{Prod}(l, F) \in \mathrm{Prod}(\mathbb{G})$ we get $\mathrm{ev}(t_{e\ 1}, n) = T_{e\ 1}(n)$ for all $n \geq l-1 = L(t_{e\ 1})$.

We note that in the last two cases $T_{e\ 1}$ is in sum-product reduced representation: the arising sums and products in F are in reduced sum-product representation by induction, F given by (20) is in reduced representation and we have $l \geq \max_{k\in S} L(f_k)$ where l is larger than all the lower bounds of the sums and product in F due to Lemma 4. This completes the induction step.

If $\mathbb{K}$ is computable and one can factorize polynomials over $\mathbb{K}$, the functions Z and L are computable and thus all the ingredients can be computed. □

Definition 11 Given $(\mathbb{G}, \sigma)$ where $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ with ev, L and Z from Example 8, let $(\mathbb{E}, \sigma)$ be a basic APS-extension with an evaluation function ev together with L and Z given by iterative application of Lemmas 2 and 3. Let $a \in \mathbb{E}$ be in reduced representation. Then following the construction of Lemma 5 one obtains $A \in \text{SumProd}(\mathbb{G})$ in sum-product reduced representation with $A(n) = \text{ev}(a, n)$ for all $n \geq L(a)$. The derived A is also called the *canonical induced sum-product expression* of a w.r.t. $(\mathbb{A}, \sigma)$ and ev and we write $\text{expr}(a) := A$.

Example 12 (Cont. of Ex. 7) For $a = x \quad \frac{x\ 1}{x} s^4 \in \mathbb{Q}(x)[s]$ with our evaluation function ev we obtain the canonical induced sum-product expression

$$\text{expr}(a) = A = x \oplus \frac{x\ 1}{x} \otimes (\text{Sum}(1, \tfrac{1}{x}) ^ 4) \in \text{Sum}(\mathbb{Q}(x))$$

with $A(n) = \text{ev}(a, n)$ for all $n \geq 1$.

Lemma 6 *Take the difference field* $(\mathbb{G}, \sigma)$ *with* $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ *with the evaluation function* ev*, o-function L and z-function Z from Example 8. Let* $(\mathbb{H}, \sigma)$ *be a basic APS-extension of* $(\mathbb{G}, \sigma)$ *and let* ev*, L and Z be extended versions for* $(\mathbb{H}, \sigma)$ *(using Lemmas 2 and 3). Let* $A \in SumProd(\mathbb{G})$*. Then there is an APS-extension* $(\mathbb{E}, \sigma)$ *of* $(\mathbb{H}, \sigma)$ *which forms a basic APS-extension of* $(\mathbb{G}, \sigma)$ *together with the extended functions* ev*, L and Z (using Lemmas 2 and 3) in which one can model A by* $a \in \mathbb{E}$*: i.e.,* $\text{ev}(a, n) = A(n)$ *holds for all* $n \geq \delta$ *for some* $\delta \in \mathbb{Z}_{\geq 0}$*.*
If $\mathbb{K}$ *is computable and one can factorize polynomials over* $\mathbb{K}$*, all the ingredients can be computed.*

Proof We show the transformation by the depth of the arising sums (Sum) and products (Prod and RPow) in $A \in \text{SumProd}(\mathbb{G})$. If no sums and products arise in A, then $A \in \mathbb{G}$ and the statement clearly holds. Now suppose that the statement holds for all expressions with sums/products whose depth is smaller than or equal to $d \geq 0$. Take all products and sums $T_1, \dots, T_r \in \quad (\mathbb{G})$ that arise in A. We proceed step wise for $i = 1, \dots, r$ with the starting field $\mathbb{H}$. Suppose that we have constructed an APS-extension $(\mathbb{A}, \sigma)$ of $(\mathbb{H}, \sigma)$ which forms a basic APS-extension of $(\mathbb{G}, \sigma)$. Suppose in addition that we are given an extended evaluation function ev, o-function L and z-function Z function (using Lemmas 2 and 3) in which we find $b_1, \dots, b_{i-1}$ with $\text{ev}(b_j, n) = T_j(n)$ for all $n \geq L(b_j)$ and all $1 \leq j < i$. Now we consider T_i.
Book keeping : If T_i has been treated earlier (i.e., by handling sums and products of depth$\leq d$), we get $b_i \in \mathbb{A}$ with $\text{ev}(b_i, n) = T_i(n)$ for all $n \geq L(a_i)$.
RPow-case: If $T_i = \text{RPow}(\alpha)$, we take the A-extension $(\mathbb{A}\langle t\rangle, \sigma)$ of $(\mathbb{A}, \sigma)$ with $\sigma(t) = \alpha t$ of order $\text{ord}(\alpha)$ and extend ev to $\mathbb{A}\langle t\rangle$ by $\text{ev}(t, n) = \alpha^n$. Further, we extend $L : \mathbb{A}\langle t\rangle \to \mathbb{Z}_{\geq 0}$ with (17) and get $L(t) = 0$. Thus we can take $b_i = t$ and get $\text{ev}(b_i, n) = T_i(n)$ for all $n \geq L(b_i) = 0$.

This step is not necessary for the proof, but allows to find APS-extensions without introducing unnecessary copies of APS-monomials. When we will refine this construction later, this step will be highly relevant.

Otherwise, we can write $T_i = \mathrm{Sum}(l, H)$ or $T_i = \mathrm{Prod}(l, H)$ where the sums and products in $H \in \mathrm{SumProd}(\mathbb{G})$ have depth at most d. By assumption we can construct an APS-extension $(\mathbb{A}', \sigma)$ of $(\mathbb{A}, \sigma)$ which is a basic APS-extension of $(\mathbb{G}, \sigma)$ and we can extend ev, L and Z (using Lemmas 2 and 3) and get $h \in \mathbb{A}'$ with $\mathrm{ev}(h, n) = H(n)$ for all $n \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$ with $\delta \geq L(h)$.
Sum-case: If $T_i = \mathrm{Sum}(\lambda, H)$, we take the S-extension $(\mathbb{A}'\langle t \rangle, \sigma)$ of $(\mathbb{A}', \sigma)$ with $\sigma(t) = t \quad \sigma(h)$. In addition, we extend ev to $\mathbb{A}'\langle t \rangle$ by $\mathrm{ev}(t, n) = \sum_{k=l}^{n} \mathrm{ev}(h, k)$ with $l = \max(\delta, \lambda)$; note that (18) is satisfied. Further, we extend $L : \mathbb{A}'\langle t \rangle \to \mathbb{Z}_{\geq 0}$ with (17) and get $L(t) = l - 1$. Finally, we set $c = \sum_{k=\lambda}^{l-1} H(k) \in \mathbb{K}$. Then we get $b_i = t \quad c$ with $\mathrm{ev}(b_i, n) = \sum_{k=\lambda}^{n} H(k) = \mathrm{ev}(\mathrm{Sum}(\lambda, H), n)$ for all $n \geq L(b_i) = l - 1$.
Product case: If $T_i = \mathrm{Prod}(l, H)$, we take the P-extension $(\mathbb{A}'\langle t \rangle, \sigma)$ of $(\mathbb{A}', \sigma)$ with $\sigma(t) = \sigma(h)t$. In addition we extend ev to $\mathbb{A}'\langle t \rangle$ by $\mathrm{ev}(t, n) = \prod_{k=l}^{n} \mathrm{ev}(h, k)$ with $l = \max(L(h), Z(h), \lambda)$; note that (18) is satisfied. Further, we extend L : $\mathbb{A}\langle t \rangle \to \mathbb{Z}_{\geq 0}$ with (17) and get $L(t) = l - 1$. Thus we can take $b_i = c\,t$ with $c = \prod_{k=\lambda}^{l-1} H(k) \in \mathbb{K}^*$ (the product evaluation is nonzero by assumption of $\quad(\mathbb{G})$) and get $\mathrm{ev}(b_i, n) = \prod_{k=\lambda}^{n} H(k) = \mathrm{ev}(\mathrm{Prod}(\lambda, H), n)$ for all $n \geq L(b_i) = l - 1$.
In all three cases we can follow Lemma 3 and extend the z-function accordingly. After carrying out the steps $i = 1, \ldots, r$ we get a basic APS-extension $(\mathbb{E}, \sigma)$ of $(\mathbb{H}, \sigma)$ together with an evaluation function ev, o-function L and z-function Z (using Lemmas 2 and 3) and $b_1, \ldots, b_r$ such that $T_i(n) = \mathrm{ev}(b_i, n)$ holds for all $1 \leq i \leq r$ and $n \geq L(b_i)$. Finally, let $f_1, \ldots, f_s \in \mathbb{G}$ be all arising elements in A (that do not arise within Prod and Sum). Define $\delta = \max(L(f_1), \ldots, L(f_s)) \in \mathbb{Z}_{\geq 0}$. Then for each $n \in \mathbb{Z}_{\geq 0}$ with $n \geq \delta$ we have that $\mathrm{ev}(A, n)$ can be carried out without catching poles in the second case of (5). Now replace each T_i with $1 \leq i \leq r$ in A by b_i and replace $\oplus, \odot, \owedge$ by $\quad, \cdot$, and $\hat{}$, respectively. This yields $a \in \mathbb{E}$ which we can write in reduced representation. Note that in a some f_k remain and others are combined by putting elements over a common denominator which lies in $\mathbb{K}[x, x_1, \ldots, x_v]$ (or in $\mathbb{K}[x_1, \ldots, x_v]$). Further, some factors of the denominators might cancel. Thus $L(a) \leq \delta$. In particular, $\mathrm{ev}(a, n)$ and $\mathrm{ev}(A, n)$ both do not enter into poles for $n \geq \delta$ and thus by the homomorphic property of the evaluation it follows that $\mathrm{ev}(a, n) = \mathrm{ev}(A, n)$ for all $n \geq \delta$. This completes the induction step.
If $\mathbb{K}$ is computable and one can factorize polynomials over $\mathbb{K}$, then the z- and o-function for $\mathbb{G}$ are computable. Thus all the components of the iterative construction (using Lemmas 2 and 3) are computable. □

As consequence we can establishes a 1-1 correspondence between basic APS-extensions and shift-stable sets whose expressions are in sum-reduced representation.

Corollary 1 *Let* $W = \{T_1, \ldots, T_e\} \subset \quad (\mathbb{G})$ *be in sum-product reduced representation and shift-stable. More precisely, for each* $1 \leq i \leq e$ *the arising sums/products in* T_i *are contained in* $\{T_1, \ldots, T_{i-1}\}$*. Then there is a basic APS-extension* $(\mathbb{E}, \sigma)$ *of* $(\mathbb{G}, \sigma)$ *with* $\mathbb{E} = \mathbb{G}\langle t_1 \rangle \ldots \langle t_e \rangle$ *equipped with an evaluation function* ev *(using Lemmas 2 and 3) such that* $T_i = expr(t_i) \in \quad (\mathbb{G})$ *holds for* $1 \leq i \leq e$.

Proof We can treat the elements $T_1, \ldots, T_e$ following the construction of Lemma 6 iteratively. Let us consider the ith step with $T_i = \mathrm{Sum}(\lambda, H)$ or $T_i = \mathrm{Prod}(\lambda, H)$.

Since the T_i are in sum-product reduced form it follows with Lemma 4 that within the Sum-case (resp. Product-case) we can guarantee $l = \lambda$, i.e, $c = 0$ (resp. $c = 1$). Thus $\text{ev}(t_i, n) = T_i(n)$ for all $n \geq l$ and thus $\text{expr}(t_i) = T_i(n)$. □

In addition, we can provide the following simple proof of Lemma 1.

Proof (of Lemma 1) Let $W = \{T_1, \dots, T_e\} \subseteq W$ be shift-stable and the T_i in sum-product reduced form. Take $F \in \text{SumProd}(W, \mathbb{G})$ and $\lambda \in \mathbb{Z}$. W.l.o.g. we may assume that the T_i are given as in Corollary 1. Thus we can take an APS-extension $(\mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle, \sigma)$ of $(\mathbb{G}, \sigma)$ equipped with an evaluation function ev and o-function L such that $\text{expr}(t_i) = T_i$ for $1 \leq i \leq e$. Then we can take $f \in \mathbb{E}$ with (19) and get $F(n \quad \lambda) = \text{ev}(F, n \quad \lambda) = \text{ev}(\sigma^{\lambda}(t_i), n) = G(n)$ for all $n \in L(f) \quad \max(0, -\lambda)$ with $G(n) := \text{expr}(\sigma^{\lambda}(t_i)) \in \text{SumProd}(W, \mathbb{G})$. Thus W is shift-closed. If $\mathbb{K}$ is computable and one can factor polynomials over $\mathbb{K}$, then one can compute the o-function L and all the above components are computable. □

In short, the naive construction of APS-extensions will not gain any substantial simplification (except a transformation to a sum-product reduced representation). In the next section we will refine this construction further to solve Problem SigmaReduce.

3.2 The embedding into the ring of sequences and *R* -extensions

Let $(\mathbb{A}, \sigma)$ be a difference ring with a subfield $\mathbb{K} \subseteq \mathbb{A}$ that is equipped with an evaluation function $\text{ev} : \mathbb{A} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$. Then ev naturally produces sequences in the commutative ring $\mathbb{K}^{\mathbb{Z}_{\geq 0}}$ with the identity element $\mathbf{1} = (1, 1, 1, \dots)$ with component-wise addition and multiplication. More precisely, we can define the function $\tau \colon \mathbb{A} \to \mathbb{K}^{\mathbb{Z}_{\geq 0}}$ with

$$\tau(f) = (\text{ev}(f, n))_{n \geq 0} = (\text{ev}(f, 0), \text{ev}(f, 1), \text{ev}(f, 2), \dots). \tag{21}$$

Due to (12) and (13) the map τ can be turned to a ring homomorphism by defining the equivalence relation $(f_n)_{n\geq 0} \equiv (g_n)_{n \geq 0}$ with $f_j = g_j$ for all $j \geq \lambda$ for some $\lambda \in \mathbb{Z}_{\geq 0}$; compare [91]. It is easily seen that the set of equivalence classes $[f]$ with $f \in \mathbb{K}^{\mathbb{Z}_{\geq 0}}$ forms with $[f] \quad [g] := [f \quad g]$ and $[f][g] := [fg]$ again a commutative ring with the identity element $[\mathbf{1}]$ which we will denote by $S(\mathbb{K})$. In the following we will simply write f instead of $[f]$. In this setting, $\tau \colon \mathbb{A} \to S(\mathbb{K})$ forms a ring homomorphism. In addition the shift operator $S : S(\mathbb{K}) \to S(\mathbb{K})$ defined by

$$S((a_0, a_1, a_2, \dots)) = (a_1, a_2, a_3, \dots)$$

turns to a ring automorphism. In the following we call $(S(\mathbb{K}), S)$ *also the (difference) ring of sequences over* $\mathbb{K}$. Finally, we observe that property (14) implies that

$$\tau(\sigma(f)) = S(\tau(f)) \tag{22}$$

holds for all $f \in \mathbb{A}$, i.e., τ turns to a difference ring homomorphism. Finally, property (11) implies

$$\tau(c) = \boldsymbol{c} = (c, c, c, \dots) \tag{23}$$

for all $c \in \mathbb{K}$. In the following we call a ring homomorphism $\tau : \mathbb{A} \to S(\mathbb{K})$ with (22) and (23) also a $\mathbb{K}$-*homomorphism*.

We can now link these notions to our construction from above with $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$. Let $(\mathbb{E}, \sigma)$ with $\mathbb{E} = \mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle$ be a basic APS-extension of $(\mathbb{G}, \sigma)$ and take an evaluation function $\mathrm{ev} : \mathbb{E} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ with o-function L. Such a construction can be accomplished by iterative application of Lemmas 2 and 3. Then the function $\tau : \mathbb{E} \to \mathbb{A}$ with (21) for $f \in \mathbb{E}$ yields a $\mathbb{K}$-homomorphism.

If we find two different elements $a, b \in \mathbb{E}$ with $\tau(a) = \tau(b)$, then we find two different reduced sum-product representations $\mathrm{expr}(a)$ and $\mathrm{expr}(b)$ in terms of the sums and products given in $W = \{\mathrm{expr}(t_1), \dots, \mathrm{expr}(t_e)\} \subseteq$ $(\mathbb{G})$ which evaluates to the same sequence. In short, W is not canonical reduced (and thus not σ-reduced) over $\mathbb{G}$. This shows that a solution of Problem SigmaReduce can be only accomplished if τ is injective.

In this context, the set of constants plays a decisive role.

Definition 12 For a difference ring $(\mathbb{A}, \sigma)$ the *set of constants* is defined by

$$\mathrm{const}_\sigma \mathbb{A} = \{c \in \mathbb{A} \mid \sigma(c) = c\}.$$

In general, $\mathrm{const}_\sigma \mathbb{A}$ is a subring of $\mathbb{A}$. If $\mathbb{A}$ is a field, then $\mathrm{const}_\sigma \mathbb{A}$ itself is a field which one also calls the *constant field of* $(\mathbb{A}, \sigma)$.

With this extra notion we can state now the following remarkable property that is based on results from [117]; compare also [125].

Theorem 1 *Let* $(\mathbb{E}, \sigma)$ *be a basic* APS*-extension of a difference field* $(\mathbb{F}, \sigma)$ *with* $\mathbb{K} = \mathrm{const}_\sigma \mathbb{F}$ *and let* τ *be a* $\mathbb{K}$*-homomorphism. Then* τ *is injective iff* $\mathrm{const}_\sigma \mathbb{E} = \mathbb{K}$.

Proof Suppose that $\mathrm{const}_\sigma \mathbb{E} = \mathbb{K}$. By Theorem [117, Thm 3.3] it follows that $(\mathbb{E}, \sigma)$ is simple (i.e., the only difference ideals in $\mathbb{E}$ is $\{0\}$ or $\mathbb{E}$) and thus by [117, Lemma 5.8] we can conclude that τ is injective. Conversely, if τ is injective, it follows by [117, Lemma 5.13] that $\mathrm{const}_\sigma \mathbb{E} = \mathbb{K}$. □

This result gives rise to the following refined definition of PS-field/APS-extensions.

Definition 13 Let $(\mathbb{F}, \sigma)$ be a PS-field extension of $(\mathbb{H}, \sigma)$ as defined in Definition 7. Then this is called a *-field extension* if $\mathrm{const}_\sigma \mathbb{F} = \mathrm{const}_\sigma \mathbb{H}$. The arising P-field and S-field monomials are also called *-field and -field monomials*, respectively. In particular, we call it a *-/ -/ -field extension* if it is built by the corresponding monomials. $(\mathbb{F}, \sigma)$ is called a *-field over* $\mathbb{K}$ if $(\mathbb{F}, \sigma)$ is a -field extension of $(\mathbb{K}, \sigma)$ and $\mathrm{const}_\sigma \mathbb{K} = \mathbb{K}$.

Example 13 As mentioned in Examples 8 and 9, the difference fields $(\mathbb{G}_r, \sigma)$, $(\mathbb{G}_b, \sigma)$ and $(\mathbb{G}_m, \sigma)$ are PS-field extensions of $(\mathbb{K}, \sigma)$. Using the technologies given in Theorems 5 and 10 below one can show that they are all -field extensions. Since $\mathrm{const}_\sigma \mathbb{K} = \mathbb{K}$, they are also -fields over $\mathbb{K}$; compare also [83].

Definition 14 Let $(\mathbb{E},\sigma)$ be an *APS*-extension of $(\mathbb{A},\sigma)$ as defined in Definition 8. Then this is called an *R -extension* if $\text{const}_\sigma\mathbb{E} = \text{const}_\sigma\mathbb{A}$. The arising *A*-monomials are also called *R-monomials*, the *P*-monomials are called a *-monomials* and the *S*-monomials are called *-monomials*. In particular, we call it an *R-/ -/ -/R -/R /- -extension* if it is built by the corresponding monomials.

Example 14 (Cont. of Ex. 10) Consider the difference ring $(\mathbb{Q}(x)[s],\sigma)$ from Example 10. Since $\text{ev} : \mathbb{Q}(x)[s] \to \mathbb{Q}$ defined by (9) and (10) (with $\text{ev}' = \text{ev}$) is an evaluation function of $(\mathbb{Q}(x)[s],\sigma)$ we can construct the $\mathbb{Q}$-homomorphism $\tau : \mathbb{Q}(x)[s] \to \mathbf{S}(\mathbb{Q})$ defined by (21). Since s is a -monomial over $\mathbb{Q}(x)$, we get $\text{const}_\sigma\mathbb{Q}(x)[s] = \mathbb{Q}$. Thus we can apply Theorem 1 and it follows that

$$\tau(\mathbb{Q}(x))[\tau(s)] = \tau(\mathbb{Q}(x))[(\text{ev}(s,n))_{n\geq 0}] = \tau(\mathbb{Q}(x))[(S(n))_{n\geq 0}]$$

with $S = \text{expr}(s) = \text{Sum}(1, \frac{1}{x}) \in \quad (\mathbb{Q}(x))$ is isomorphic to the polynomial ring $\mathbb{Q}(x)[s]$. Further, $(S(n))_{n\geq 0}$ with $S(n) = \sum_{k=1}^{n} \frac{1}{k}$ is transcendental over $\tau(\mathbb{Q}(x))$.

Example 14 generalizes as follows. Suppose that we are given a basic *R* -extension $(\mathbb{E},\sigma)$ of $(\mathbb{G},\sigma)$ with

$$\mathbb{G}[\rho_1]\dots[\rho_l][p_1,p_1^{-1}]\dots[p_u,p_u^{-1}][s_1]\dots[s_r]$$

where the ρ_i are *R*-monomials with $\zeta_i = \frac{\sigma(\rho_i)}{\rho_i} \in \mathcal{R}$ being primitive roots of unity, t_i are -monomials and the s_i are -monomials. In addition, take an evaluation function ev with *o*-function L by iterative applications of Lemmas 2 and 3. Here we may assume that

- $\text{ev}(\rho_i,n) = \zeta_i^n$ for all $1 \leq i \leq l$,
- $\text{ev}(p_i,n) = P_i(n)$ with $\text{expr}(p_i) = P_i \in \quad (\mathbb{G})$ for all $1 \leq i \leq u$, and
- $\text{ev}(s_i,n) = S_i(n)$ with $\text{expr}(s_i) = S_i \in \quad (\mathbb{G})$ for all $1 \leq i \leq r$.

Then $\tau : \mathbb{E} \to S(\mathbb{K})$ with (21) is a $\mathbb{K}$-homomorphism. By Theorem 1 it follows that τ is injective and thus

$$\begin{aligned}\tau(\mathbb{E}) = \tau(\mathbb{G})\ &[\tau(\rho_1)]\dots[\tau(\rho_l)]\\ &\times[\tau(p_1),\tau(p_1)^{-1}]\dots[\tau(p_u),\tau(p_u)^{-1}]\\ &\times[\tau(s_1)]\dots[\tau(s_r)]\\ = \tau(\mathbb{G})\ &[(\zeta_1^n)_{n\geq 0}]\dots[(\zeta_l^n)_{n\geq 0}]\\ &\times[(P_1(n))_{n\geq 0},(\tfrac{1}{P_1(n)})_{n\geq 0}]\dots[(P_u(n))_{n\geq 0},(\tfrac{1}{P_u(n)})_{n\geq 0}]\\ &\times[(S_1(n))_{n\geq 0}]\dots[(S_r(n))_{n\geq 0}]\end{aligned}$$

forms a (Laurent) polynomial ring extension over the ring of sequences $R = \tau(\mathbb{G})[(\zeta_1^n)_{n\geq 0}]\dots[(\zeta_l^n)_{n\geq 0}]$. In particular, we conclude that the sequences

$$(P_1(n))_{n\geq 0}, (\tfrac{1}{P_1(n)})_{n\geq 0}, \dots (P_u(n))_{n\geq 0}, (\tfrac{1}{P_u(n)})_{n\geq 0}, (S_1(n))_{n\geq 0}, \dots, (S_r(n))_{n\geq 0}$$

are, up to the trivial relations $(P_i(n))_{n\geq 0} \cdot (\frac{1}{P_i(n)})_{n\geq 0}$ for $1 \leq i \leq u$, algebraically independent among each other over the ring R.

We are now ready to state the main result of this section that connects SumProd($\mathbb{G}$) with difference ring theory.

Theorem 2 *Let $(\mathbb{E}, \sigma)$ be a basic APS-extension of $(\mathbb{G}, \sigma)$ with $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ and $\mathbb{A} = \mathbb{E}\langle t_1\rangle \dots \langle t_e\rangle$ equipped with an evaluation function* ev *(using Lemmas 2 and 3). Take the $\mathbb{K}$-homomorphism $\tau : \mathbb{E} \to S(\mathbb{K})$ with $\tau(f) = (\mathrm{ev}(f, n))_{n\geq 0}$ and $T_i = expr(t_i) \in$ ($\mathbb{G}$) for $1 \leq i \leq e$. Then the following statements are equivalent.*

1. *$(\mathbb{E}, \sigma)$ is an R -extension of $(\mathbb{G}, \sigma)$.*
2. *τ is a $\mathbb{K}$-isomorphism between $(\mathbb{E}, \sigma)$ and $(\tau(\mathbb{E}), S)$; in particular all sequences generated by the -monomials are algebraically independent over the ring given by the sequences of $\tau(\mathbb{G})$ adjoined with the sequences generated by R-monomials.*
3. *$W = \{T_1, \dots, T_e\}$ is canonical-reduced over $\mathbb{G}$.*
4. *The zero recognition problem is trivial, i.e., for any $F \in$ SumProd$(W, \mathbb{G})$ the following holds: if $\mathrm{ev}(F, n) = 0$ for all $n \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$, then $F = 0$.*

Proof (1) $\Leftrightarrow$ (2) is an immediate consequence of Theorem 1.
(2) $\Rightarrow$ (3): Let $F, F' \in \mathrm{SumProd}(W, \mathbb{G})$ with $F(n) = F'(n)$ for all $n \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$. Replace in F, F' any occurrences of $T_i^{\textcircled{\wedge}} z_i$ for $1 \leq i \leq e$ with $z_i \in \mathbb{Z}$ by $t_i^{z_i}$, $\oplus$ by , and $\odot$ by $\cdot$. This yields $f, f' \in \mathbb{E}$ with $\mathrm{ev}(f, n) = F(n)$ for all $n \geq L(f)$ and $\mathrm{ev}(f', n) = F'(n)$ for all $n \geq L(f')$. Hence $\tau(f) = \tau(f')$. Since τ is injective, $f = f'$. But this implies that F and F' are the same up to trivial permutations of the operands in $\odot$ and $\oplus$. Consequently W is canonical reduced.
(3) $\Rightarrow$ (4): Suppose that W is canonical reduced and take $F \in \mathrm{SumProd}(W, \mathbb{G})$ with $F(n) = 0$ for all $n \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$. Since $\mathrm{ev}(0, n) = 0$ for all $n \geq 0$ and W is canonical reduced, it follows that $F = 0$.
(4) $\Rightarrow$ (1): Suppose that τ is not injective and take $f \in \mathbb{E} \setminus \{0\}$ with $\tau(f) = \mathbf{0}$. By Lemma 5 we can take 0 $F \in \mathrm{SumProd}(\{T_1, \dots, T_e\}, \mathbb{G})$ and $\delta \in \mathbb{Z}_{\geq 0}$ with $\mathrm{ev}(f, n) = F(n) = 0$ for all $n \geq \delta$. Thus statement (4) does not hold. □

In order to derive the equivalences in Theorem 2 we assumed that an *APS*-extension is given. We can relax this assumption if the set W is shift-stable.

Corollary 2 *Let $W = \{T_1, \dots, T_e\} \subset$ ($\mathbb{G}$) be in sum-product reduced representation and shift-stable. More precisely, for each $1 \leq i \leq e$ the arising sums/products in T_i are contained in $\{T_1, \dots, T_{i-1}\}$. Then the two statements are equivalent:*

1. *There is a basic R -extension $(\mathbb{E}, \sigma)$ of $(\mathbb{G}, \sigma)$ with $\mathbb{E} = \mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle$ equipped with an evaluation function* ev *(using Lemmas 2 and 3) with $T_i = expr(t_i) \in$ ($\mathbb{G}$) for $1 \leq i \leq e$.*
2. *$W = \{T_1, \dots, T_e\}$ is σ-reduced over $\mathbb{G}$.*

Proof (1) $\Rightarrow$ (2): By assumption W is sum-product reduced and shift-stable, and by (1) $\Rightarrow$ (3) of Theorem 2 it is canonical-reduced. Thus W is σ-reduced.
(2) $\Rightarrow$ (1): By Corollary 1 we get an *APS*-extension $(\mathbb{E}, \sigma)$ of $(\mathbb{G}, \sigma)$ with $\mathbb{E} = \mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle$ equipped with an evaluation function ev (using Lemmas 2 and 3) with

$T_i = \mathrm{expr}(t_i) \in \quad (\mathbb{G})$ for $1 \leq i \leq e$. Since W is canonical reduced, it follows by (3) $\Rightarrow$ (1) of Theorem 2 that $(\mathbb{E}, \sigma)$ is an R -extension of $(\mathbb{G}, \sigma)$. □

Corollary 2 yields immediately a strategy (actually the only strategy for shift-stable sets) to solve Problem SigmaReduce.

Strategy to solve Problem SigmaReduce

Given: $A_1, \dots, A_u \in \mathrm{SumProd}(\mathbb{G})$ with $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$, i.e., $\mathbb{G} = \mathbb{K}(x, x_1, \dots, x_v)$ or $\mathbb{G} = \mathbb{K}(x_1, \dots, x_v)$.
Find: a σ-reduced set $W = \{T_1, \dots, T_e\} \subset \quad (\mathbb{G}')$ with $B_1 \dots, B_u \in \mathrm{SumProd}(W, \mathbb{G}')$ and $\delta_1, \dots, \delta_u \in \mathbb{Z}_{\geq 0}$ such that $A_i(n) = B_i(n)$ holds for all $n \geq \delta_i$ and $1 \leq i \leq r$.

1. Construct an R -extension $(\mathbb{E}, \sigma)$ of $(\mathbb{G}', \sigma)$ with $\mathbb{E} = \mathbb{G}'\langle t_1\rangle \dots \langle t_e\rangle$ equipped with an evaluation function $\mathrm{ev} : \mathbb{E} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}'$ and o-function L in which $A_1, \dots, A_u$ are modeled by $a_1, \dots, a_u \in \mathbb{E}$. More precisely, for $1 \leq i \leq u$ we compute in addition $\delta_i \in \mathbb{Z}_{\geq 0}$ with $\delta_i \geq L(a_i)$ such that
$$A_i(n) = \mathrm{ev}(a_i, n) \quad \forall n \geq \delta_i. \tag{24}$$
2. Set $W = \{T_1, \dots, T_e\}$ with $T_i := \mathrm{expr}(t_i) \in \quad (\mathbb{G}')$ for $1 \leq i \leq e$.
3. Set $B_i := \mathrm{expr}(a_i) \in \mathrm{SumProd}(W, \mathbb{G}')$ for $1 \leq i \leq u$.
4. Return W, $(B_1, \dots, B_u)$ and $(\delta_1, \dots, \delta_u)$.

What remains open is to enrich this general method with the construction required in step (1). This task will be considered in detail in the next section.

4 The representation problem

In this section we will give an overview of the existing algorithms that assist in the task to solve the open subproblem given in step (1) of our general method SigmaReduce. The resulting machinery can be summarized as follows.

Theorem 3 *Given $A_1, \dots, A_u \in \mathit{SumProd}_1(\mathbb{G})$ with $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ where $\mathbb{K}$ is a rational function field over an algebraic number field. Then one can compute a σ-reduced set $W = \{T_1, \dots, T_e\} \subset \quad_1(\mathbb{G})$ with $B_1 \dots, B_u \in \mathit{SumProd}(W, \mathbb{G})$ and $\delta_1, \dots, \delta_u \in \mathbb{Z}_{\geq 0}$ such that $A_i(n) = B_i(n)$ holds for all $n \geq \delta_i$ and $1 \leq i \leq u$.*

Theorem 4 *Given $A_1, \dots, A_u \in \mathit{SumProd}(\mathbb{K}(x))$ where $\mathbb{K} = \mathcal{A}(y_1, \dots, y_o)$ is a rational function field over an algebraic number field $\mathcal{A}$. Then one can take $\mathbb{K}' = \mathcal{A}'(y_1, \dots, y_o)$ where $\mathcal{A}'$ is an algebraic extension of $\mathcal{A}$ and can compute a*

Here we get $\mathbb{G}' = \mathbb{K}'(x, x_1, \dots, x_v)$ or $\mathbb{G}' = \mathbb{K}'(x_1, \dots, x_v)$ where $\mathbb{K}'$ is a field extension of $\mathbb{K}$; if $A_1, \dots, A_u \in \mathrm{SumProd}_1(\mathbb{G})$, we are in the special case $\mathbb{G} = \mathbb{G}'$.

σ-reduced set $W = \{T_1, \ldots, T_e\} \subset$ $(\mathbb{K}'(x))$ with $B_1 \ldots, B_u \in SumProd(W, \mathbb{G}')$ and $\delta_1, \ldots, \delta_u \in \mathbb{Z}_{\geq 0}$ such that $A_i(n) = B_i(n)$ holds for all $n \geq \delta_i$ and $1 \leq i \leq u$.

Here we will start with the problem to represent products in R -monomials (see Subsection 4.1). More precisely, we will show various tactics that enable one to represent expressions of $\mathrm{Prod}_1(\mathbb{G}_r)$, $\mathrm{Prod}_1(\mathbb{G}_b)$, $\mathrm{Prod}_1(\mathbb{G}_m)$ and $\mathrm{Prod}(\mathbb{G}_r)$. Afterwards, we will consider the problem to represent nested sums over such products (i.e., expressions of $\mathrm{Sum}(\mathbb{G})$, $\mathrm{SumProd}_1(\mathbb{G}_r)$, $\mathrm{SumProd}_1(\mathbb{G}_b)$, $\mathrm{SumProd}_1(\mathbb{G}_m)$ and $\mathrm{SumProd}(\mathbb{G}_r)$) in -monomials (see Subsection 4.2).

Remark 2 `Sigma` can represent fully algorithmically single nested products in R -extensions; in addition, Ocansey's package `NestedProducts` can deal with the case $\mathrm{Prod}(\mathbb{G}_r)$. Expressions from more general domains (e.g., sums and products that arise non trivially in denominators) also work with the function call **`SigmaReduce`** of `Sigma`. But for these cases the underlying summation mechanisms (like given in Lemmas 5 and Lemma 6) are only partially developed and the back translation from the difference ring setting to the term algebra might fail.

In general, it suffices in our proposed construction to compute an R -extension in which a finite set of sums and products are modeled. However, in some important instances it is possible to perform this constructions step wise.

Definition 15 Fix X as one of the term algebras $\mathrm{Prod}_1(\mathbb{G})$, $\mathrm{Prod}(\mathbb{G})$, $\mathrm{Sum}(\mathbb{G})$, $\mathrm{SumProd}_1(\mathbb{G})$, $\mathrm{SumProd}(\mathbb{G})$, and let $\mathcal{D}$ be a subclass of basic R -extensions of $(\mathbb{G}, \sigma)$. Then $\mathcal{D}$ is called *X-extension stable* if for any $(\mathbb{H}, \sigma) \in \mathcal{D}$ and any $A \in X$ one can construct an R -extension $(\mathbb{E}, \sigma)$ of $(\mathbb{H}, \sigma)$ with $(\mathbb{E}, \sigma) \in \mathcal{D}$ and $a \in \mathbb{E}$ such that one can model A with a.

We note that within such an extension stable class of R -extensions one does not have to treat the arising sums and products in one stroke, but one can consider them iteratively. This is in particular interesting, when unforeseen sums and products arise in a later step, that have to be considered in addition. In a nutshell, we will provide a general overview of the existing tools to design basic R -extensions. In particular, we will emphasis the available algorithms to construct extension-stable versions.

4.1 Representation of products in R -extensions

We start with algorithmic tools that enables one to test if a P-extension forms a -extension. Based on these tools we present (without further details) the existing techniques to represent a finite set of products in a R -extension.

4.1.1 Algorithmic tests

In [108, Theorem 9.1] based on Karr's work [65, 66] a general criterion for -field extensions is elaborated. Here we present a more flexible version in the ring setting.

Theorem 5 *Let* $(\mathbb{E},\sigma)$ *be a P-extension of a difference ring* $(\mathbb{H},\sigma)$ *with* $\mathbb{E}=\mathbb{H}\langle t_1\rangle\dots\langle t_d\rangle$ *and* $f_i=\frac{\sigma(t_i)}{t_i}\in\mathbb{H}^*$ *for* $1\leq i\leq d$*. Suppose that*

$$\{g\in\mathbb{H}\setminus\{0\}\mid\sigma(g)=u\,g\text{ for some }u\in\mathbb{H}^*\}\subseteq\mathbb{H}^* \tag{25}$$

holds. Then the following statements are equivalent:

1. $(\mathbb{E},\sigma)$ *is a -extension of* $(\mathbb{H},\sigma)$*, i.e.,* $\text{const}_\sigma\mathbb{E}=\text{const}_\sigma\mathbb{H}$.
2. *There do not exist* $g\in\mathbb{H}\setminus\{0\}$ *and* $(z_1,\dots,z_d)\in\mathbb{Z}^d\setminus\{\mathbf{0}\}$ *with*

$$\sigma(g)=f_1^{z_1}\dots f_d^{z_d}\,g.$$

Proof (1) ⇒ (2): Suppose that there is a $g\in\mathbb{H}\setminus\{0\}$ and $(z_1,\dots,z_d)\in\mathbb{Z}^d\setminus\{\mathbf{0}\}$ with $\sigma(g)=f_1^{z_1}\dots f_d^{z_d}g$. Let i be maximal such that z_i 0. Then we can take $h=g\,f_1^{-z_1}\dots f_{i-1}^{-z_{i-1}}$ and get $\sigma(h)=f_i^{z_i}\,h$. With part (2) of Theorem 2.12 in [115] it follows that $(\mathbb{H}\langle t_1\rangle\dots\langle t_i\rangle,\sigma)$ is not a -extension of $(\mathbb{H}\langle t_1\rangle\dots\langle t_{i-1}\rangle,\sigma)$.
(2) ⇒ (1): Let i with $1\leq i\leq d$ be minimal such that $(\mathbb{H}\langle t_1\rangle\dots\langle t_i\rangle,\sigma)$ is not a -extension of $(\mathbb{H}\langle t_1\rangle\dots\langle t_{i-1}\rangle,\sigma)$. Then $\sigma(g)=\alpha_i^{z_i}\,g$ for some $g\in\mathbb{H}\langle t_1\rangle\dots\langle t_{i-1}\rangle\setminus\{0\}$ and $z_i\in\mathbb{Z}\setminus\{0\}$ by part (2) of Theorem 2.12 in [115]. In particular, with property (25) we can apply Theorem 22 of [115] and it follows that $g=h\,t_1^{-z_1}\dots t_{i-1}^{-z_{i-1}}$ for some $z_i\in\mathbb{Z}$ and $h\in\mathbb{H}^*$. Thus we get $\sigma(h)=\alpha_1^{z_1}\dots\alpha_i^{z_i}\,h$ with z_i 0 which shows statement (1). □

Remark 3 (1) Theorem 10 contains the following special case (see [66] for the field and [115] for the ring case): a P extension $(\mathbb{A}\langle p\rangle,\sigma)$ of $(\mathbb{A},\sigma)$ with $f:=\frac{\sigma(p)}{p}\in\mathbb{A}^*$ is a -extension iff there are no $g\in\mathbb{A}$, $m\in\mathbb{Z}\setminus\{0\}$ with $\sigma(g)=f\,g$.
(2) Often Theorem 5 is applied to the special case that the ground ring $(\mathbb{H},\sigma)$ forms a field. Note that in this particular instance, the assumption (25) trivially holds.

Let $(\mathbb{A},\sigma)$ be a difference ring and $\boldsymbol{f}=(f_1,\dots,f_d)\in(\mathbb{A}^*)^d$. Then we define

$$M(\boldsymbol{f},\mathbb{A}):=\{(m_1,\dots,m_d)\in\mathbb{Z}^d\,|\,\sigma(g)=f_1^{m_1}\dots f_d^{m_d}\,g\text{ for some }g\in\mathbb{A}\setminus\{0\}\};$$

see also [65]. Note that Theorem 5 states that the P-extension $(\mathbb{E},\sigma)$ of the difference ring $(\mathbb{H},\sigma)$ with $\mathbb{E}=\mathbb{H}\langle t_1\rangle\dots\langle t_d\rangle$ and $f_i=\frac{\sigma(t_i)}{t_i}\in\mathbb{H}^*$ for $1\leq i\leq d$ is a -extension if and only if $M(\boldsymbol{f},\mathbb{H})=\{\mathbf{0}\}$. If $\boldsymbol{f}\in([\mathbb{F}^*]_{\mathbb{F}}^{\mathbb{H}})^d$ (which holds for $\mathbb{F}^*$-basic P-extensions), this latter property can be checked by utilizing the following result.

Theorem 6 *Let* $(\mathbb{H},\sigma)$ *be a basic R -extension of a difference field* $(\mathbb{F},\sigma)$ *and* $\boldsymbol{f}\in([\mathbb{F}^*]_{\mathbb{F}}^{\mathbb{H}})^d$*. Then the following holds:*

1. $M(\boldsymbol{f},\mathbb{H})$ *is a* $\mathbb{Z}$*-module over* $\mathbb{Z}^d$.
2. *If one can compute a basis of* $M(\boldsymbol{h},\mathbb{F})$ *for any* $\boldsymbol{h}\in(\mathbb{F}^*)^m$ *with* $m\geq 1$*, then one can compute a basis of* $M(\boldsymbol{f},\mathbb{H})$.

Proof Part (1) follows by Lemma 2.6 and Theorem 2.22 of [115] and part (2) by [115, Theorem 2.23]. □

In other words, we can apply Theorem 5 to test if a basic P-extension over $\mathbb{F}$ is a -extension if one can compute a basis of $M(\boldsymbol{h}, \mathbb{F})$ in a difference field $(\mathbb{F}, \sigma)$. In particular, using the algorithms from [65] this is possible if $(\mathbb{F}, \sigma)$ is a -field over $\mathbb{K}$ where the constant field satisfies certain algorithmic properties.

Definition 16 A field $\mathbb{K}$ is called *σ-computable* if the following holds:

1. One can factorize multivariate polynomials over $\mathbb{K}$;
2. given $(f_1, \dots, f_d) \in (\mathbb{K}^*)^d$ one can compute for $\{(z_1, \dots, z_d) \in \mathbb{Z}^d \mid f_1^{z_1} \dots f_d^{z_d} = 1\}$ a $\mathbb{Z}$-basis;
3. one can decide if $c \in \mathbb{K}$ is an integer.

More precisely, the following holds if $(\mathbb{F}, \sigma)$ is a -field over a σ-computable constant field; special cases are $\mathbb{G}_r$, $\mathbb{G}_b$ or $\mathbb{G}_m$ where $\mathbb{K}$ is σ-computable.

Corollary 3 *Let $(\mathbb{E}, \sigma)$ be a basic R -extension of a -field $(\mathbb{F}, \sigma)$ over $\mathbb{K}$. If $\mathbb{K}$ is σ-computable, one can compute a basis of $M(\boldsymbol{h}, \mathbb{E})$ for any $\boldsymbol{h} \in ([\mathbb{F}^*]_{\mathbb{F}}^{\mathbb{E}})^d$ with $d \geq 1$. This in particular the case, if $\mathbb{K} = \mathcal{A}(y_1, \dots, y_o)$ is a rational function over an algebraic number $\mathcal{A}$.*

Proof If $\mathbb{K}$ is σ-computable, it follows by [65, Theorem 9] that one can compute a basis of $M(\boldsymbol{f}, \mathbb{F})$ for any $\boldsymbol{f} \in (\mathbb{F}^*)^m$ with $m \geq 1$. Thus by part 2 of Theorem 6 one can compute a basis of $M(\boldsymbol{h}, \mathbb{E})$ for any $\boldsymbol{h} \in ([\mathbb{F}^*]_{\mathbb{F}}^{\mathbb{E}})^d$ with $d \geq 1$. In particular, it follows by [103, Thm. 3.5] (based on the algorithm of [59]) that a rational function field over an algebraic number field is σ-computable. □

Remark 4 (1) By [115, Theorem 2.26] Corollary 3 is also valid for $\boldsymbol{f} \in (\{\mathbb{F}^*\}_{\mathbb{F}}^{\mathbb{E}})^d$ in simple R -extension defined over a -field. As elaborated in [115, Sect. 2.3.3] (using ideas of [70]) it holds even in the more general setting that $(\mathbb{F}, \sigma)$ is a -field extension of a difference field $(\mathbb{F}_0, \sigma)$ where all roots of unities in $\mathbb{F}$ are constants and $(\mathbb{F}_0, \sigma)$ is σ-computable; for the definition of these algorithmic properties we refer to [70, Def. 1]. Further aspects can be also found in [21]. In particular, all these properties hold, if $(\mathbb{F}_0, \sigma)$ is a free difference field [69, 70] (covering generic/unspecified sequences X_n) or is built by radical extensions [71] (covering objects like $\sqrt{n}$). For the underlying implementations enhancing `Sigma` we refer to [70, 71].
(2) Within `Sigma` the case of -fields is implemented properly where the constant field is given by a rational function field over the rational numbers. In parts also algebraic numbers work, but here we rely on sub-optimal routines of Mathematica.

4.1.2 Algorithmic representations

In this section we present several algorithms that provide proofs of Theorems 3 and 4 if one restricts to the cases $\text{Prod}_1(\mathbb{G})$ with $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ or $\text{Prod}(\mathbb{G}_r)$, i.e., if one drops expressions where sums arise. More precisely, we will introduce several solutions of step (1) for our method SigmaReduce.

First, we treat the case $\text{Prod}_1(\mathbb{G})$. In this setting (where also sums can arise) single-basic R -extensions, a subclass of basic R -extensions, are sufficient.

Definition 17 An $R\quad$-extension $(\mathbb{E},\sigma)$ of a difference ring $(\mathbb{A},\sigma)$ with $\mathbb{E} = \mathbb{A}\langle t_1\rangle\dots\langle t_e\rangle$ is called *single-basic* if for any R-monomial t_i we have $\frac{\sigma(t_i)}{t_i} \in \text{const}_\sigma\mathbb{A}^*$ and for any P-monomial t_i we have $\frac{\sigma(t_i)}{t_i} \in \mathbb{A}^*$.

We will present the following two main strategies.

• *Optimal product representations.* In [119, Theorem 69] we showed that one can construct $R\quad$-extensions with minimal extension degree and minimal order.

Theorem 7 *Let* $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ *and* $A_1,\dots,A_u \in Prod_1(\mathbb{G})$*. Then there is a single-basic* $R\quad$*-extension* $(\mathbb{E},\sigma)$ *of* $(\mathbb{G},\sigma)$ *with* $\mathbb{E} = \mathbb{G}\langle t_1\rangle\dots\langle t_e\rangle$ *together with an evaluation function* ev *and o-function* L *with the following properties:*

1. $A_1,\dots,A_u$ *are modeled by* $a_1,\dots,a_u \in \mathbb{E}$*, i.e., for all* $1 \le i \le u$ *we have* (24) *for some explicitly given* $\delta_i \in \mathbb{Z}_{\geq 0}$ *with* $\delta_i \geq L(a_i)$.
2. *There is at most one* R*-monomial in* $\mathbb{E}$*. If this is the case, the order* λ *is minimal among all such extensions in which one can model* $a_1,\dots,a_u$*.*
3. *The number of   -monomials in* $\mathbb{E}$ *is minimal among all such extensions in which one can model* $a_1,\dots,a_u$*.*

If the constant field of $(\mathbb{K},\sigma)$ *is a rational function field over an algebraic number field, then the above components are computable.*

Example 15 For the following products in $\text{Prod}_1(\mathbb{Q}[\mathbb{i}](x)$ with the imaginary unit $\mathbb{i}$:

$$A_1 = \text{Prod}\big(1, \tfrac{-13122x(1\ \ x)}{(3\ \ x)^3}\big), \qquad A_2 = \text{Prod}\big(1, \tfrac{26244x^2(2\ \ x)^2}{(3\ \ x)^2}\big),$$
$$A_3 = \text{Prod}\big(1, \tfrac{\mathbb{i}k(2\ \ x)^3}{729(5\ \ x)}\big), \qquad A_4 = \text{Prod}\big(1, \tfrac{-162x(2\ \ x)}{5\ \ x}\big),$$

we compute the alternative expressions $B_1 = \frac{5(1\ \ x)^2(2\ \ x)^5(3\ \ x)^8}{52488(4\ \ x)(5\ \ x)}T_1T_2T_3^{-2}$, $B_2 = \frac{(4\ \ x)^2(5\ \ x)^2}{400}T_2^2$, $B_3 = \frac{2754990144(4\ \ x)^2(5\ \ x)^2}{25(1\ \ x)^4(2\ \ x)^{10}(3\ \ x)^{16}}T_3^3$ and $B_4 = T_2$ in terms of the σ-reduced set $W = \{T_1, T_2, T_3\}$ with

$$T_1 = \text{RPow}(-1), \quad T_2 = \text{Prod}(1, \tfrac{-162x(2\ \ x)}{5\ \ x}), \quad T_3 = \text{Prod}(1, \tfrac{-\mathbb{i}(3\ \ x)^6}{9x(1\ \ x)^2(2\ \ x)(5\ \ x)});$$

internally, T_1 is modeled by an R-monomial of order 2 and T_2, T_3 are modeled by two   -monomials. Details on this construction are given in [119, Ex. 70].

We remark that this optimal representation has one essential drawback: if further products have to be treated in a later situation, the existing difference ring cannot be reused, but a completely new difference ring has to be designed.

• *Extension stable representations for completely factorizable constant fields.* In the following we will follow another approach: instead of computing the smallest ring in which one can model a finite set of single nested products, we design a difference ring where the multiplicands are as small as possible such that the constructed difference rings are $\text{Prod}_1(\mathbb{G})$-extension stable. In order to accomplish this task, we will restrict the constant field $\mathbb{K}$ further as follows.

A *ring* R *is called completely factorizable* if R is a unique factorization domain (UFD) and all units in R are roots of unity. In particular, any element $a \in R$ can be written in the form $a = u\, a_1^{n_1} \dots a_l^{n_l}$ with a root of unity u, $n_1, \dots, n_l \in \mathbb{Z}_{\geq 1}$ and $a_1, \dots, a_l \in R$ being coprime irreducible elements. In addition, a *field* K *is called completely factorizable* if it is the quotient field of completely factorizable ring R. In such a field any element $a \in K$ can be written in the form $a = u\, a_1^{n_1} \dots a_l^{n_l}$ with a root of unity u, $n_1, \dots, n_l \in \mathbb{Z} \setminus \{0\}$ and $a_1, \dots, a_l \in R$ being coprime irreducible elements. We call K *completely factorizable of order* $\lambda \in \mathbb{Z}_{\geq 0}$, if the set of roots of unity is finite and the maximal order is λ. We say that *complete factorizations are computable over such a field* K if for any rational function from $K(x_1, \dots, x_r)$ a complete factorization can be computed.

The following lemma allows to lift the property of completely factorizable rings.

Lemma 7 *If a ring (resp. field)* $\mathbb{A}$ *is completely factorizable, the polynomial ring* $\mathbb{A}[x_1, \dots, x_r]$ *(resp. rat. function field* $\mathbb{A}(x_1, \dots, x_r)$*) is completely factorizable.*

Example 16 The ring $\mathbb{Z}$ and the Gaussian ring $\mathbb{Z}[\mathbb{i}]$ with the roots uf unity $1, -1$ and $1, -1, \mathbb{i}, -\mathbb{i}$, respectively. Thus $\mathbb{Z}$, $\mathbb{Z}[\mathbb{i}]$ and, in particular $\mathbb{Z}[x_1, \dots, x_r]$ and $\mathbb{Z}[\mathbb{i}][x_1, \dots, x_r]$ are completely factorizable rings. Furthermore, their quotient fields $\mathbb{Q}$, $\mathbb{Q}[\mathbb{i}]$, $\mathbb{Q}(x_1, \dots, x_r)$ and $\mathbb{Q}[\mathbb{i}](x_1, \dots, x_r)$ are completely factorizable of order 2 or 4, respectively. In particular, one can compute complete factorizations over $\mathbb{Q}[i]$.

Definition 18 Let $\mathbb{F}$ be the quotient field of a completely factorizable ring R of order λ. A single-basic R -extension $(\mathbb{E}, \sigma)$ of $(\mathbb{F}, \sigma)$ with $\mathbb{E} = \mathbb{F}\langle t_1 \rangle \dots \langle t_e \rangle$ is called *completely factorized* if there is at most one R-monomial ρ with $\frac{\sigma(\rho)}{\rho} \in (\text{const}_\sigma \mathbb{F})^*$ of order λ and for any -monomial t_i we have that $\frac{\sigma(t_i)}{t_i} \in R$ is irreducible.

We are now ready to state the following result implemented within `Sigma`; the case $\mathbb{G}_r$ is covered by [113, Theorem 2]; the extension to $\mathbb{G}_b$ and $\mathbb{G}_m$ is straightforward.

Theorem 8 *Let* $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ *where* $\mathbb{K}$ *is completely factorizable of order* λ*. Then the class of completely factorized* R *-extensions over* $(\mathbb{G}, \sigma)$ *is* $Prod_1(\mathbb{G})$*-extension stable. More precisely, let* $(\mathbb{H}, \sigma)$ *be a completely factorized* R *-extension of* $(\mathbb{G}, \sigma)$ *equipped with an evaluation function* ev *an o-function* L*. Let* $A \in Prod_1(\mathbb{G})$*. Then there is an* R *-extension* $(\mathbb{E}, \sigma)$ *of* $(\mathbb{H}, \sigma)$ *with an extended evaluation function* ev *and o-function* L *with the following properties:*

1. $(\mathbb{E}, \sigma)$ *is a completely factorizable* R *-extension of* $(\mathbb{G}, \sigma)$*.*
2. A *is modeled by* $a \in \mathbb{E}$*, i.e.,* $A(n) = \text{ev}(a, n)$ *for all* $n \geq \delta$ *for some* $\delta \in \mathbb{Z}_{\geq 0}$*.*

If complete factorization over $\mathbb{K}$ *can be computed, all components are computable.*

Example 17 Given the products (15), we can split the multiplicands into irreducible factors and get (after some technical details) the product representations $B_1 = \frac{216 T_1^2 T_2 T_3^8}{(n\;\;1)^2 (n\;\;2)^3 (n\;\;3)^3 T_4}$, $B_2 = \frac{9 T_2^2 T_3^8 T_4^2}{(n\;\;3)^2}$, $B_3 = \frac{15 (n\;\;1)^2 (n\;\;2)^2 T_1^2 T_4^3}{(n\;\;3)(n\;\;4)(n\;\;5) T_3^6}$ and $B_4 = \frac{60 T_1^2 T_2 T_3^4 T_4}{(n\;\;3)(n\;\;4)(n\;\;5)}$ in terms of the σ-reduced set $W = \{T_1, T_2, T_3, T_4\}$ with

$$T_1 = \mathrm{RPow}(\mathbb{i}), \quad T_2 = \mathrm{Prod}(1,2), \quad T_3 = \mathrm{Prod}(1,3), \quad T_4 = \mathrm{Prod}(1,x);$$

internally, T_1 is modeled by an R-monomial of order 4, and T_2, T_3, T_4 are modeled by three -monomials.

It would be interesting to see extension-stable difference ring constructions that work in more general settings. A first step in this direction has been elaborated in [83, Theorem 6.2]. Here a toolbox (implemented within `NestedProducts`) is summarized where one tries to follow the above construction of completely factorized R -extensions as much as possible. In this way, a modification of the existing R -extension will arise only for products whose multiplicands are taken from an algebraic number field.

• *Representation of nested products.* We obtained the first algorithm in [84, Theorem 9] (implemented in `NestedProducts`) to represent products from $\mathrm{Prod}(\mathbb{G}_r)$ fully algorithmically in a basic R -extension. This result can be stated as follows.

Theorem 9 *Let $\mathbb{G} = \mathbb{G}_r = \mathbb{K}(x)$ where $\mathbb{K} = \mathcal{A}(y_1, \dots, y_o)$ with $o \geq 0$ is a rational function field over an algebraic number field $\mathcal{A}$. Then for $A_1, \dots, A_u \in Prod(\mathbb{G})$ one can compute a basic R -extension $(\mathbb{E}, \sigma)$ of $(\mathbb{G}', \sigma)$ with an evaluation function* ev *and o-function L with the following properties:*

1. *The ground field $\mathbb{G}$ is extended to $\mathbb{G}' = \mathbb{K}'(x)$ where $\mathbb{K}' = \mathcal{A}'(y_1, \dots, y_o)$ with $\mathcal{A}'$ being an algebraic field extension of $\mathcal{A}$.*
2. *Within the R -monomials in $(\mathbb{E}, \sigma)$ there is at most one R-monomial.*
3. *$A_1, \dots, A_u$ are modeled by $a_1, \dots, a_u \in \mathbb{E}$, i.e., for all $1 \leq i \leq u$ we have* (24) *for some explicitly given $\delta_i \in \mathbb{Z}_{\geq 0}$ with $\delta_i \geq L(a_i)$.*

Remark 5 Theorem 7 holds also for general ground rings $(\mathbb{G}, \sigma)$ with certain algorithmic properties; see [119]. Fascinating structural properties of mixed hypergeometric products (and related objects within the differential case) are presented in [51]. Further simplification aspects within -fields (e.g., finding products where the degrees of the top most sum or product in the numerator and denominator of a multiplicand are minimal) are elaborated in [25, 103]. In addition, methods to find algebraic relations of sequences built by products are given in [73, 84, 108, 119, 123].

4.2 Representation of sums

4.2.1 Algorithmic tests via (parameterized) telescoping

We will proceed as in the product case. The additive version of Theorem 5, which is nothing else than parameterized telescoping (see Section 5.2), reads as follows.

Theorem 10 ([117, Thm. 7.10]) *Let $(\mathbb{E}, \sigma)$ be an S-extension of a difference ring $(\mathbb{H}, \sigma)$ with $\mathbb{E} = \mathbb{H}\langle t_1 \rangle \dots \langle t_d \rangle$ and $f_i = \sigma(t_i) - t_i \in \mathbb{H}$ for $1 \leq i \leq d$. If $\mathbb{K} := \mathrm{const}_\sigma \mathbb{H}$ is a field, then the following statements are equivalent:*

1. $(\mathbb{E}, \sigma)$ *is a -extension of* $(\mathbb{H}, \sigma)$*, i.e.,* $\mathrm{const}_\sigma \mathbb{E} = \mathrm{const}_\sigma \mathbb{H}$.
2. *There do not exist* $g \in \mathbb{H}$ *and* $(c_1, \dots, c_d) \in \mathbb{K}^d \setminus \{\mathbf{0}\}$ *with*

$$\sigma(g) - g = c_1 f_1 \quad \cdots \quad c_d f_d.$$

Note that Theorem 10 contains the following special case (compare [65] for the field case and [115] for the ring case): an S extension $(\mathbb{A}[s], \sigma)$ of $(\mathbb{A}, \sigma)$ with $f := \sigma(s) - s \in \mathbb{A}$ is a -extension if and only if there is no $g \in \mathbb{A}$ such that the telescoping equation $\sigma(g) - g = f$ holds; this property will be crucial for the construction that establishes Theorem 12 given below.

Let $(\mathbb{A}, \sigma)$ be a difference ring with constant field $\mathbb{K}$, $u \in \mathbb{A} \setminus \{0\}$ and $\boldsymbol{f} = (f_1, \dots, f_d) \in \mathbb{A}^d$. Then following [65] we define the *set of solutions of parameterized first-order linear difference equations*:

$$V_1(u, \boldsymbol{f}, \mathbb{A}) = \{(c_1, \dots, c_d, g) \in \mathbb{K}^d \times \mathbb{A} \mid \sigma(g) - u\,g = c_1 f_1 \quad \cdots \quad c_d f_d\}.$$

With this notion, Theorem 10 can be restated as follows: $(\mathbb{E}, \sigma)$ is a -extension of $(\mathbb{H}, \sigma)$ if and only if $V_1(1, (f_1, \dots, f_d), \mathbb{H}) = \{0\}^d \times \mathbb{K}$. In order to check that this is the case, we can utilize the following theorem.

Theorem 11 *Let* $(\mathbb{H}, \sigma)$ *be a basic R -extension of a difference field* $(\mathbb{F}, \sigma)$ *with constant field* $\mathbb{K}$*,* $u \in [\mathbb{F}^*]_{\mathbb{F}}^{\mathbb{H}}$ *and* $\boldsymbol{f} \in \mathbb{H}^d$*. Then the following holds:*

1. $V_1(u, \boldsymbol{f}, \mathbb{H})$ *is a* $\mathbb{K}$*-vector space of dimensions* $\leq d \quad 1$.
2. *If one can compute a basis of* $M(\boldsymbol{h}, \mathbb{F})$ *for any* $\boldsymbol{h} \in (\mathbb{F}^*)^n$ *and a basis of* $V_1(v, \boldsymbol{h}, \mathbb{F})$ *for any* $v \in \mathbb{F}^*$*,* $\boldsymbol{h} \in \mathbb{F}^n$*, then one can compute a basis of* $V_1(u, \boldsymbol{f}, \mathbb{H})$.

Proof Lemma 2.17 and Thm. 2.22 of [115] give (1); [115, Thm. 2.23]. □

In particular, we can activate this machinery if $(\mathbb{F}, \sigma)$ is a -field over a σ-computable constant field; a special case is, e.g., $\mathbb{F} = \mathbb{G}_m$.

Corollary 4 *Let* $(\mathbb{E}, \sigma)$ *be an R -extension of a -field* $(\mathbb{F}, \sigma)$ *over* $\mathbb{K}$*. If* $\mathbb{K}$ *is* σ*-computable, one can compute a basis of* $V_1(1, \boldsymbol{f}, \mathbb{E})$ *for any* $\boldsymbol{f} \in (\mathbb{E}^*)^d$*. This in particular the case, if* $\mathbb{K}$ *is a rational function over an algebraic number field.*

Proof If $\mathbb{K}$ is σ-computable, it follows by [65] (or [103]) that one can compute a basis of $V_1(u, \boldsymbol{f}, \mathbb{F})$ for any $u \in \mathbb{F}^*$, $\boldsymbol{f} \in (\mathbb{F}^*)^d$. Thus by part 2 of Theorem 11 one can compute a basis of $V_1(1, \boldsymbol{h}, \mathbb{E})$ for any $\boldsymbol{h} \in (\mathbb{E}^*)^d$. In particular, it follows by [103, Thm. 3.5] (based on the algorithm of [59]) that a rational function field over an algebraic number field is σ-computable. □

Remark 6 (1) By [115, Thm. 2.26] Corollary 4 is also valid for $\boldsymbol{f} \in (\{\mathbb{F}^*\}_{\mathbb{F}}^{\mathbb{E}})^d$ in simple R -extensions over a -field. As elaborated in [115, Sect. 2.3.3] it holds even in the more general setting that $(\mathbb{F}, \sigma)$ is a -field extension of a difference field $(\mathbb{F}_0, \sigma)$ which is σ^*-computable (see [70, Def. 1]) and one can solve a basis of $V(u, \boldsymbol{f})$ in $(\mathbb{F}_0, \sigma^k)$ for any $k > 0$, $u \in \mathbb{F}^*$ and $\boldsymbol{f} \in \mathbb{F}_0^m$; see also Remark 4.(1).

If the extension is basic, we only need the case $k = 1$.

4.2.2 Basic representations

The following theorem (based on Theorem 10 and the property that one can solve the telescoping problem (26) given below) enables one to lift the results of $\mathrm{Prod}_1(\mathbb{G})$ and $\mathrm{Prod}(\mathbb{G}_r)$ form Section 4.1 to the cases $\mathrm{Sum}(\mathbb{G})$, $\mathrm{SumProd}_1(\mathbb{G})$ and $\mathrm{SumProd}(\mathbb{G}_r)$.

Theorem 12 *Let $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ and $A_1, \dots, A_u \in$ SumProd$(\mathbb{G})$. Let $(\mathbb{H}, \sigma)$ be a basic R -extension of $(\mathbb{G}, \sigma)$ equipped with an evaluation function* ev *and an o-function L where all arising products in $A_1, \dots, A_u$ can be modeled. Then there is a -extension $(\mathbb{E}, \sigma)$ of $(\mathbb{H}, \sigma)$ with an extended evaluation function* ev *and o-function L such that $a_1, \dots, a_u \in \mathbb{E}$ model $A_1, \dots, A_u$, i.e., for all $1 \le i \le u$ we have* (24) *for some explicitly given $\delta_i \in \mathbb{Z}_{\ge 0}$ with $\delta_i \ge L(a_i)$.*
If $\mathbb{K}$ is σ-computable, and $L : \mathbb{H} \to \mathbb{Z}_{\ge 0}$ and $\mathrm{ev} : \mathbb{H} \times \mathbb{Z}_{\ge 0} \to \mathbb{K}$ *are computable, the above components can be computed.*

Proof This result follows from the construction given in [117, pp. 657–658] which can be summarized as follows. We suppose that we have constructed already a basic R -extension of $(\mathbb{G}, \sigma)$ equipped with an evaluation function ev and an o-function L where all arising products in $A_1, \dots, A_u$ can be modeled. Then we can adapt the construction of Lemma 6 and deal with all arising sums and products arising in the $A_1, \dots, A_u$. Suppose that we have constructed already a -extension $(\mathbb{A}, \sigma)$ of $(\mathbb{G}, \sigma)$ and we are treating now the product or sum T_i. If it is a product, we sort it out in the book keeping step and obtain an element $b_i \in \mathbb{H}^* \subseteq \mathbb{E}^*$ that models T_i by assumption. Otherwise, $T_i = \mathrm{Sum}(\lambda, H)$. By assumption (on the depth of the arising sums) we can construct a -extension $(\mathbb{A}', \sigma)$ of $(\mathbb{A}, \sigma)$ together with an extended evaluation function ev and o-function L such that we can take $h \in \mathbb{A}'$ with $\mathrm{ev}(h, n) = H(n)$ for all $n \ge L(h)$. Now we enter the sum case and perform the following extra test. We check if there is a $g \in \mathbb{A}'$ with

$$\sigma(g) = g \quad f \quad \Leftrightarrow \quad \sigma(g) - g = f \tag{26}$$

for $f := \sigma(h)$. Suppose there is such a g. We define $\delta_i := \max(L(f), L(g), l)$. Then for $b_i := g \quad \sum_{j=l}^{\delta_i} H(j) - \mathrm{ev}(g, \delta_i) \in \mathbb{A}'$ we get $\mathrm{ev}(b_i, n \quad 1) - \mathrm{ev}(b_i, n) = \mathrm{ev}(g, n \quad 1) - \mathrm{ev}(g, n) = H(n \quad 1)$ and $T_i(n \quad 1) = T_i(n) \quad H(n \quad 1)$ for all $n \ge \delta_i$. Since $\mathrm{ev}(b_i, \delta_i) = \sum_{j=l}^{\delta_i} F(j) = \mathrm{ev}(T_i, \delta_i)$, we get $\mathrm{ev}(b_i, n) = \mathrm{ev}(T_i, n)$ for all $n \ge \delta_i$. Otherwise, if there is no such g, we proceed as in the sum-case of Lemma 6: we adjoin the -monomial t to $\mathbb{A}'$ with $\sigma(t) = t \quad f$ with $f = \sigma(h)$ and get the claimed $b_i = t \quad c$ with $c \in \mathbb{K}$ such that $\mathrm{ev}(b_i, n) = \mathrm{ev}(T_i, n)$ holds for all $n \ge L(b_i) = \delta_i$.
Summarizing, we can construct a nested -extension in which the elements from $\mathrm{SumProd}(\mathbb{G})$ can be modeled. If $\mathbb{K}$ is σ-computable, one can decide constructively by Corollary 4 if there exists such a g. Furthermore, if $L : \mathbb{H} \to \mathbb{Z}_{\ge 0}$ and $\mathrm{ev} : \mathbb{H} \times \mathbb{Z}_{\ge 0} \to \mathbb{K}$ are computable also their extensions for $(\mathbb{E}, \sigma)$ are computable by recursion. Consequently, all components are computable. □

We get immediately the following result for $\mathrm{Sum}(\mathbb{G})$-stable extensions.

Corollary 5 *Let $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$. The class of -extensions over $(\mathbb{G}, \sigma)$ is Sum$(\mathbb{G})$-extension stable. More precisely, let $(\mathbb{H}, \sigma)$ be a -extension of $(\mathbb{G}, \sigma)$*

with an evaluation function ev *and o-function L, and let* $A \in \textit{Sum}(\mathbb{G})$*. Then there is a -extension* $(\mathbb{E}, \sigma)$ *of* $(\mathbb{H}, \sigma)$ *with an extended evaluation function* ev *and o-function L together with* $a \in \mathbb{E}$ *and* $\delta \in \mathbb{Z}_{\geq 0}$ *with* $A(n) = \text{ev}(a, n)$ *for all* $n \geq \delta$*.*
If $\mathbb{K}$ *is* σ*-computable, these components can be computed.*

Combining Theorems 8 and 12 we get `Sigma`'s main translation mechanism.

Corollary 6 *Let* $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ *where* $\mathbb{K}$ *is completely factorizable of order* λ*. Then the class of completely factorized R -extensions is* $\textit{SumProd}_1(\mathbb{G})$*-extension stable. More precisely, let* $(\mathbb{H}, \sigma)$ *be a completely factorized R -extension of* $(\mathbb{G}, \sigma)$ *equipped with an evaluation function* ev *an o-function L. Let* $A \in \textit{SumProd}_1(\mathbb{G})$*. Then there is an R -extension* $(\mathbb{E}, \sigma)$ *of* $(\mathbb{H}, \sigma)$ *with an extended evaluation function* ev *and o-function L with the following properties:*

1. $(\mathbb{E}, \sigma)$ *is a completely factorizable R -extension of* $(\mathbb{G}, \sigma)$*.*
2. *A is modeled by* $a \in \mathbb{E}$*, i.e.,* $A(n) = \text{ev}(a, n)$ *for all* $n \geq \delta$ *for some* $\delta \in \mathbb{Z}_{\geq 0}$*.*

If $\mathbb{K}$ *is* σ*-computable and complete factorizations over* $\mathbb{K}$ *can be computed, all the components can be given explicitly.*

Proof We can write $\mathbb{H} = \mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle[s_1, \dots, s_u]$ where the t_i are R -monomials and the s_i are -monomials. Take all products that arise in A. Since $(\mathbb{H}_0, \sigma)$ with $\mathbb{H}_0 = \mathbb{G}\langle t_1\rangle \dots \langle t_e\rangle$ is a completely factorized R -extension of $(\mathbb{G}, \sigma)$, we can apply Theorem 8 and get an R -extension $(\mathbb{H}_1, \sigma)$ of $(\mathbb{H}_0, \sigma)$ with $\mathbb{H}_1 = \mathbb{H}_0\langle p_1\rangle \dots \langle p_v\rangle$ together with an extended evaluation function ev and o-function L such that $(\mathbb{H}_1, \sigma)$ is a completely factorized R -extension of $(\mathbb{G}, \sigma)$ and such that all products in A can be modeled in $\mathbb{H}_1$. By [117, Cor. 6.5] (together with [117, Prop 3.23]) it follows that also $(\mathbb{H}_2, \sigma)$ with $\mathbb{H}_2 = \mathbb{H}\langle p_1\rangle \dots \langle p_v\rangle$ is a -extension of $(\mathbb{H}, \sigma)$. In particular, $(\mathbb{H}_2, \sigma)$ is a completely factorized R -extension of $(\mathbb{G}, \sigma)$ and we can merge the evaluation functions and o-functions to $\text{ev} : \mathbb{H}_2 \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ and $L : \mathbb{H}_2 \to \mathbb{Z}_{\geq 0}$. Finally, we apply Theorem 12 and get a -extension of $(\mathbb{E}, \sigma)$ of $(\mathbb{H}_2, \sigma)$ with an appropriately extended evaluation function ev and o-function L together with $a \in \mathbb{E}$ and $\delta \in \mathbb{Z}_{\geq 0}$ such that $\text{ev}(a, n) = A(n)$ holds for all $n \geq \delta$. By definition $(\mathbb{E}, \sigma)$ is a completely factorized R -extension of $(\mathbb{G}, \sigma)$.
If $\mathbb{K}$ is σ-computable and one can compute complete factorizations over $\mathbb{K}$, Theorems 8 and 12 are constructive and all components can be computed. □

Furthermore, combining Theorems 7 and 12 gives the following result (we omit the optimality properties given in Theorem 7).

Corollary 7 *Let* $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$ *where* $\mathbb{K}$ *is built by a rational function field defined over an algebraic number field. Then for* $A_1, \dots, A_u \in \textit{SumProd}_1(\mathbb{G})$ *there is a single-basic R -extension* $(\mathbb{E}, \sigma)$ *of* $(\mathbb{G}, \sigma)$ *together with an extended evaluation function* $\text{ev} : \mathbb{E} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ *and o-function* $L : \mathbb{E} \to \mathbb{Z}_{\geq 0}$ *with the following properties:* $A_1, \dots, A_u$ *are modeled by* $a_1, \dots, a_u \in \mathbb{E}$*, i.e., for all* $1 \leq i \leq u$ *we have* (24) *for some explicitly given* $\delta_i \in \mathbb{Z}_{\geq 0}$ *with* $\delta_i \geq L(a_i)$*.*

In addition, the applications of Theorems 9 and 12 yield the following statement.

Corollary 8 *Let $\mathbb{G}_r = \mathbb{K}(x)$ with $\mathbb{K} = \mathcal{A}(y_1, \dots, y_o)$ be a rational function field over an algebraic number field $\mathcal{A}$. Then for $A_1, \dots, A_u \in$ SumProd$(\mathbb{G})$ there is a basic R -extension $(\mathbb{E}, \sigma)$ of $(\mathbb{G}'_r, \sigma)$ with an evaluation function* ev : $\mathbb{E} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}'$ *and o-function $L : \mathbb{E} \to \mathbb{Z}_{\geq 0}$ with the following properties:*

1. *The ground field $\mathbb{G}_r$ is extended to $\mathbb{G}'_r = \mathbb{K}'(x)$ where $\mathbb{K}' = \mathcal{A}'(y_1, \dots, y_o)$ with $\mathcal{A}'$ being an algebraic field extension of $\mathcal{A}$.*
2. *Within the R -monomials in $(\mathbb{E}, \sigma)$ there is at most one R-monomial.*
3. *$A_1, \dots, A_u$ are modeled by $a_1, \dots, a_u \in \mathbb{E}$, i.e., for all $1 \leq i \leq u$ we have* (24) *for some explicitly given $\delta_i \in \mathbb{Z}_{\geq 0}$ with $\delta_i \geq L(a_i)$.*

In particular, activating our method **`SigmaReduce`** in combination with Corollaries 6 and 7 establishes Theorems 3 and 3, respectively.

Technical details of the summation package `Sigma`

Remark 7 (1) Within `Sigma` the function call `SigmaReduce` follows the method given on page 26. Note that in this construction the σ-reduced set W is constructed by treating step wise the sums and products that occur in the A_i.
(2) The user can control the σ-reduced set W manually by introducing extra sums and products with the option `Tower`$\to \{S_1, \dots, S_v\}$ that will be parsed before the arsing sums in $A_1, \dots, A_u$ are considered; as an example we refer to In[20] in Example 19.
(3) `Sigma` is tuned for expressions from SumProd$_1(\mathbb{G})$ where the constant field $\mathbb{K}$ is a completely factorizable field. In particular for the case that $\mathbb{K}$ is a rational function field over the rational numbers, the machinery given in Corollary 6 is highly robust. `Sigma` also works partially with rational function fields over algebraic number fields; but here it depends on the stability of the subroutines in Mathematica.
(4) For nested products the machinery of `SigmaReduce` works if the objects can be transformed straightforwardly to R -extensions. For more complicated situations the objects SumProd$(\mathbb{G}_r)$ can be handled fully algorithmically in combination with Ocansey's package `NestedProducts`.

Remark 8 As observed in [36] an algebraic independent basis of certain classes of indefinite nested sums can be obtained by exploiting the underlying quasi-shuffle algebra. In [36] this aspect has been utilized for the class of harmonic sums. Later it has been shown in [16] that the relations in the class of cyclotomic harmonic sums produced by difference ring theory (compare Theorem 1) and by the quasi-shuffle algebra are equivalent. As a consequence, the quasi-shuffle algebra of cyclotomic sums induces a canonical representation. We emphasize that many of the above aspects can be carried over to a summation theory of unspecified sequences [88].

4.2.3 Depth-optimal representations

In [101, 107] we have refined Karr's definition of -field extensions to depth-optimal -field extensions and have developed improved telescoping algorithms therein. In this way, we could provide a general toolbox in [110] that can find representations such that the nesting depth of the arising sums are minimal. As it turns out, the underlying telescoping algorithms can be adapted (and even simplified) for R -extensions. For the specification of the refined representation (without entering into technical details) we need the following definition.

Definition 19 A finite set $W \subset \quad(\mathbb{G})$ is called *depth-optimal* if for any $G \in$ SumProd$(W, \mathbb{G})$ and $G' \in$ SumProd$(\mathbb{G})$ with $G(n) = G'(n)$ for all $n \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$ it follows that $\delta(G) \leq \delta(G')$ holds.

Then combining the results from Section 4.2.2 with the tools from [101, 107, 110] we obtain algorithms that can solve the following problem if $\mathbb{K}$ is σ-computable; for simplicity we skipped the general case SumProd$(\mathbb{G})$. Further technical details concerning the implementation in `Sigma` can be found in Remark 7.

Problem DOS: Depth-optimal SigmaReduce

Given: $A_1, \dots, A_u \in \text{SumProd}_1(\mathbb{G}_m)$.
Find: a finite σ-reduced depth-optimal set $W \subset \quad {}_1(\mathbb{G}_m)$ together with $B_1, \dots, B_u \in \text{SumProd}(W, \mathbb{G}_m)$ and $\delta_1, \dots, \delta_u \in \mathbb{Z}_{\geq 0}$ such that $A_i(n) = B_i(n)$ holds for all $n \geq \delta_i$ and $1 \leq i \leq u$

Example 18 Given the sums $A_1, A_2, A_3 \in \text{Sum}(\mathbb{Q}(x))$ defined by

In[15]:= $$\{\mathbf{A_1}, \mathbf{A_2}, \mathbf{A_3}\} = \left\{\sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^2}\right) \sum_{i=1}^{k} \frac{(-1)^i}{i}}{1 \quad k}, \sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^2}\right) \sum_{i=1}^{k} \frac{(-1)^i}{i}}{2 \quad k}, \sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^2}\right) \sum_{i=1}^{k} \frac{(-1)^i}{i}}{3 \quad k}\right\};$$

we get the alternative expressions $B_1, B_2, B_3 \in \text{SumProd}(W, \mathbb{Q}(x))$ by executing

In[16]:= **{B₁, B₂, B₃} = SigmaReduce[{A₁, A₂, A₃}, n]**

Out[16]= $$\left\{\sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^2}\right) \sum_{i=1}^{k} \frac{(-1)^i}{i}}{1 \quad k}, \sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^2}\right) \sum_{i=1}^{k} \frac{(-1)^i}{i}}{2 \quad k},\right.$$
$$\frac{3}{16} \quad \frac{(-3-2n)(-1)^n}{8(1 \quad n)(2 \quad n)} \quad \frac{(-1)^n}{2(2 \quad n)} \sum_{i=1}^{n} \frac{1}{i^2} \quad \frac{1}{2} \sum_{i=1}^{n} \frac{(-1)^i}{i^2} \quad \frac{-3 \quad 2n \quad 2n^2}{4(1 \quad n)(2 \quad n)} \sum_{i=1}^{n} \frac{(-1)^i}{i}$$
$$\left. - \frac{(1 \quad n)(5 \quad 2n)}{2(2 \quad n)(3 \quad n)} \left(\sum_{i=1}^{n} \frac{1}{i^2}\right) \sum_{i=1}^{n} \frac{(-1)^i}{i} \quad \frac{1}{2} \sum_{i=1}^{n} \frac{\left(\sum_{j=1}^{i} \frac{1}{j^2}\right) \sum_{j=1}^{i} \frac{(-1)^j}{j}}{1 \quad i} \quad \frac{1}{2} \sum_{i=1}^{n} \frac{\left(\sum_{j=1}^{i} \frac{1}{j^2}\right) \sum_{j=1}^{i} \frac{(-1)^j}{j}}{2 \quad i}\right\}$$

with the σ-reduced set

$$W = \left\{ \sum_{k=1}^{n} \frac{1}{k^2}, \sum_{k=1}^{n} \frac{(-1)^k}{k}, \sum_{i=1}^{n} \frac{(-1)^i}{i^2}, \sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^2}\right) \sum_{i=1}^{k} \frac{(-1)^i}{i}}{1 \quad k}, \sum_{k=1}^{n} \frac{\left(\sum_{i=1}^{k} \frac{1}{i^2}\right) \sum_{i=1}^{k} \frac{(-1)^i}{i}}{2 \quad k} \right\}.$$

Note: instead of A_3 (a sum of nesting depth 3) the simpler sum $\sum_{i=1}^{n} \frac{(-1)^i}{i^2}$ (with nesting depth 2) has been introduced automatically.

Remark 9 Further refined -extensions, such as reduced -extensions, have been elaborated in [109] (based on improved telescoping algorithms given in [99, 114]).

5 The summation paradigms

We have explained in detail how sums and products can be modeled automatically within R -extensions. Thus steps 1 and 3 on page (1) are settled and we focus on step 2: We will introduce the summation paradigms in difference rings and fields; further details how these problems are handled in `Sigma` are given below.

5.1 Refined telescoping

As indicated in Section 4.2.2, in particular in Theorem 12, the construction of basic R -extensions for the representation of SumProd($\mathbb{G}$) is based on algorithms that solve the telescoping problem (26). In particular, the quality of the constructed extensions and the used telescoping algorithms are intertwined by each other. As illustrated for instance in Section 4.2.3, the underlying telescoping algorithms could be refined further (using [101, 107, 110]) to compute depth-optimal representations. In the following we will focus on the available telescoping technologies in `Sigma` (based on [99, 101, 105, 107–110, 114]) that enable one to simplify sums further. For simplicity we will focus on sums from $_1(\mathbb{G}_m)$ and skip, e.g., the case $(\mathbb{G}_r)$.

Problem RT: Refined Telescoping

Given: $F \in \text{SumProd}_1(\mathbb{G}_m)$.
Find: $\delta \in \mathbb{Z}_{\geq 0}$ and a σ-reduced set $W = \{T_1, \dots, T_e\} \subset {}_1(\mathbb{G}_m)$ where $\mathtt{d}(T_1) \leq \mathtt{d}(T_2) \leq \dots \leq \mathtt{d}(T_e)$ together with $F', G \in \text{SumProd}(W, \mathbb{G}_m)$ such that for all $k \geq \delta$ we have $F(k) = F'(k)$ and

$$G(k \quad 1) - G(k) = F'(k).$$

- Refinement 1: W is depth-optimal (by using **SimplifyByExt→MinDepth**).
- Refinement 2: In addition, if $\mathtt{d}(G) = \mathtt{d}(F') \quad 1$, then $\mathtt{d}(T_{e-1}) < \mathtt{d}(T_e) = \mathtt{d}(G)$ and $T_e = \text{Sum}(\delta, H)$ with $H \in \text{SumProd}(\{T_1, \dots, T_i\}, \mathbb{G}_m)$ where i with $1 \leq i < e$ is minimal (by using **SimplifyByExt→DepthNumber**).
- Refinement 3: One can compute, among all possible choices with i minimal, H such that also $\deg_{T_i}$ is minimal (by using **SimplifyByExt→DepthNumberDegree**).

Given such G and $\delta \in \mathbb{Z}_{\geq 0}$ for F we obtain the simplification (2) for all $n \geq \delta$.

Example 19 We start with the the following sum:

In[17]:= $\textbf{mySum1} = \sum_{k=1}^{n} \Big(\sum_{j=1}^{k} \frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{k} \frac{(-1)^j}{j}\Big)^2;$

Telescoping without any refinements does not yield a simplification. However, by activating the first refinement with the option **SimplifyByExt→MinDepth** we get

```
In[18]:= SigmaReduce[mySum1, n, SimplifyByExt → MinDepth]
```

Out[18]= $\frac{1}{3}\sum_{i=1}^{n}\frac{(-1)^i}{i^3}\quad (-1)^{1\ \ n}\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\sum_{j=1}^{n}\frac{(-1)^j}{j}\quad (1\ \ \ n)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^2-\frac{1}{3}\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^3$

We illustrate the second refinement with the sum:

In[19]:= **mySum2 =** $\sum_{k=1}^{n}\Big(\sum_{j=1}^{k}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{k}\frac{(-1)^j}{j}\Big)^3$**;**

In[20]:= **SigmaReduce[mySum2, n, SimplifyByExt → DepthNumber, Tower →** $\Big\{\sum_{i=1}^{n}\frac{(-1)^i}{i},\sum_{i=1}^{n}\frac{(-1)^i}{i^2}\Big\}$
SimpleSumRepresentation → False]

Out[20]= $\frac{1}{4}\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)^2-\frac{3}{2}(-1)^n\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^2\quad (1\ \ \ n)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^3$
$-\frac{1}{4}\sum_{i=1}^{n}\Big(\frac{1}{i^4}-\frac{6\Big(\sum_{j=1}^{i}\frac{(-1)^j}{j}\Big)^2}{i^2}\quad\frac{4(-1)^i\Big(\sum_{j=1}^{i}\frac{(-1)^j}{j}\Big)^3}{i}\Big)$

Namely, within the given extension (specified by `Tower`→ $\{\sum_{i=1}^{n}\frac{(-1)^i}{i},\sum_{i=1}^{n}\frac{(-1)^i}{i^2}\}$, compare Remark 7) we find a sum extension which is free of $\sum_{i=1}^{n}\frac{(-1)^i}{i^2}$. Without the option `SimpleSumRepresentation`→`False` further simplifications on the found sum (using in addition partial fraction decomposition) are applied and one gets:

In[21]:= **SigmaReduce[mySum2, n, SimplifyByExt → DepthNumber, Tower →** $\Big\{\sum_{i=1}^{n}\frac{(-1)^i}{i},\sum_{i=1}^{n}\frac{(-1)^i}{i^2}\Big\}$**]**

Out[21]= $-\frac{1}{4}\sum_{i=1}^{n}\frac{1}{i^4}\quad\frac{1}{4}\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)^2-\frac{3}{2}(-1)^n\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^2$
$(1\ \ \ n)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^3\quad\frac{3}{2}\sum_{i=1}^{n}\frac{\Big(\sum_{j=1}^{i}\frac{(-1)^j}{j}\Big)^2}{i^2}-\sum_{i=1}^{n}\frac{(-1)^i\Big(\sum_{j=1}^{i}\frac{(-1)^j}{j}\Big)^3}{i}$

If one changes the order of the extension with the option `Tower`→ $\{\sum_{i=1}^{n}\frac{(-1)^i}{i^2},\sum_{i=1}^{n}\frac{(-1)^i}{i}\}$, no simplification is possible with the option `SimplifyByExt`→`DepthNumber`. However, using the option `SimplifyByExt`→`DepthNumberDegree` one finds a sum extension where in the summand the degree w.r.t. $T=\sum_{i=1}^{n}\frac{(-1)^i}{i}$ is minimal. In this case we find

In[22]:= **SigmaReduce[mySum2, n, SimplifyByExt → DepthNumberDegree, Tower →** $\Big\{\sum_{i=1}^{n}\frac{(-1)^i}{i^2},\sum_{i=1}^{n}\frac{(-1)^i}{i}\Big\}$
SimpleSumRepresentation → False]

Out[22]= $-\frac{3}{2}(-1)^n\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^2\quad(1\ \ \ n)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^3-\frac{1}{4}\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^4$
$\frac{1}{4}\sum_{i=1}^{n}\Big(-\frac{3}{i^4}\quad\frac{2(-1)^i}{i^2}\sum_{j=1}^{i}\frac{(-1)^j}{j^2}\quad\frac{4(-1)^i}{i^3}\sum_{j=1}^{i}\frac{(-1)^j}{j}\Big)$

where in the summand of the found sum the degree w.r.t. T is 1. With the option `SimpleSumRepresentation`→`True` (which is the standard option) this sum is simplified further (by splitting it into atomics by partial fraction decomposition) and we get:

In[23]:= **SigmaReduce[mySum2, n, SimplifyByExt → DepthNumberDegree, Tower →** $\Big\{\sum_{i=1}^{n}\frac{(-1)^i}{i^2},\sum_{i=1}^{n}\frac{(-1)^i}{i}\Big\}$

$$\text{Out[23]=}\ -\frac{1}{2}\sum_{i=1}^{n}\frac{1}{i^4}\quad\frac{1}{4}\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)^2-\frac{3}{2}(-1)^n\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^2$$
$$(1\quad n)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j^2}\Big)\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^3-\frac{1}{4}\Big(\sum_{j=1}^{n}\frac{(-1)^j}{j}\Big)^4\quad\sum_{i=1}^{n}\frac{(-1)^i}{i^3}\sum_{j=1}^{i}\frac{(-1)^j}{j}$$

5.2 Parameterized telescoping (including creative telescoping)

The summation paradigm of telescoping can be generalized as follows.

Problem PT: Parameterized Telescoping

Given: $F_1,\dots,F_d\in\mathrm{SumProd}(\mathbb{G})$ with $\mathbb{G}\in\{\mathbb{G}_r,\mathbb{G}_b,\mathbb{G}_m\}$.
Find: Find, if possible, a suitable σ-reduced finite set $W\subset\quad(\mathbb{G}')$ and $\delta\in\mathbb{Z}_{\geq 0}$ with the following properties; as in Problem SigmaReduce, one might have to extend the constant field $\mathbb{K}$ of $\mathbb{G}$ to $\mathbb{K}'$ yielding $\mathbb{G}'$.

- One can take $F_1',\dots,F_d'\in\mathrm{SumProd}(W,\mathbb{G}')$ such that for all $1\leq i\leq d$ and all $k\geq\delta$ we have $F_i(k)=F_i'(k)$;
- one can take $c_1,\dots,c_d\in\mathbb{K}'$ with $c_1\quad 0$ and $G\in\mathrm{SumProd}(W,G')$ such that for all $k\geq\delta$ we have

$$G(k\quad 1)-G(k)=c_1\,F_1'(k)\quad\cdots\quad c_d\,F_d'(k).\tag{27}$$

Given such $c_1,\dots,c_d\in\mathbb{K}$, G and $\delta\in\mathbb{Z}_{\geq 0}$ for $F_1,\dots,F_d$, we obtain

$$c_1\sum_{k=\delta}^{n}F_1(k)\quad\cdots\quad c_d\sum_{k=\delta}^{n}F_d(k)=G(n\quad 1)-G(\delta)\tag{28}$$

for all $n\geq\delta$. In particular, if one is given a bivariate sequence $F(n,k)$ with $F_i(k)=F(n\quad i-1,k)\in\mathrm{SumProd}(\mathbb{G})$ for $i=1,\dots,d$, equation (27) turns to (3). In particular, the sum relation (28) can can be transformed to the recurrence (4) for the sum $S(n)=\sum_{k=\delta}^{n}F(n,k)$. Summarizing, parameterized telescoping contains creative telescoping [131] as a special case.

A straightforward solution to the above problem can be obtained by the application of Theorem 13. In the context of σ-reduced sets this can be rephrased as follows.

Proposition 1 *Let $W=\{T_1,\dots,T_e\}\subseteq\quad(\mathbb{G})$ be σ-reduced where for each $1\leq i\leq e$ the arising sums and products within in T_i are contained in $\{T_1,\dots,T_{i-1}\}$ and are in sum-product reduced form. Let $F_1',\dots,F_d'\in$ SumProd$(W,\mathbb{G})$. Then one can compute, in case of existence, $(c_1,\dots,c_d)\in\mathbb{K}^d$ with $c_1\quad 0$ together with $G\in$ SumProd$(W,\mathbb{G})$ and $\delta\in\mathbb{Z}_{\geq 0}$ such that* (27) *holds for all $k\geq\delta$.*

Proof By Corollary 2 we get an $R\quad$-extension $(\mathbb{E},\sigma)$ of $(\mathbb{G},\sigma)$ with $\mathbb{E}=\mathbb{G}\langle t_1\rangle\dots,\langle t_e\rangle$ together with an evaluation function ev and o-function L with

$\mathrm{expr}(t_i) = T_i$ for all $1 \leq i \leq e$. In particular, we get $\boldsymbol{f} = (f_1, \dots, f_d) \in \mathbb{E}^d$ with $\mathrm{ev}(f_i, k) = F_i'(k)$ for all $1 \leq i \leq u$ and all $n \geq L(f_i)$. Note that $(c_1, \dots, c_d, G) \in \mathbb{K}^d \times \mathrm{SumProd}(W, \mathbb{G})$ with (27) for all $k \geq \delta$ for some $\delta \in \mathbb{Z}_{\geq 0}$ iff $(c_1, \dots, c_d, g) \in \mathbb{K}^d \times \mathbb{E}$. By Theorem 13 we can compute a basis $V = V_1(1, \boldsymbol{f}, \mathbb{E})$ and can check if there is $(c_1, \dots, c_d, g) \in V$ with $c_1 \quad 0$. If this is not the case, then there is no $(c_1, \dots, c_d, G) \in \mathbb{K}^d \times \mathrm{SumProd}(W, \mathbb{G})$ with $c_1 \quad 0$. Otherwise, we rephrase the result to $(c_1, \dots, c_d, G) \in \mathbb{K}^d \times \mathrm{SumProd}(W, \mathbb{G})$ such that (27) holds for all $k \geq \delta$ with $\delta = \max(L(F_1'), \dots, L(F_u'), L(G))$. □

Remark 10 In Proposition 1 we assume that the input expressions from $\mathrm{SumProd}(\mathbb{G})$ can be rephrased directly to an R -extension. If this is not the case, the representation machinery has to be applied in a preprocessing step. To support this construction, the user can control the σ-reduced set W as outlined in the Remark 7.(2) above. But this should be done with care in order to avoid useless results. If W contains, e.g., $T_j \in \mathrm{Sum}(l, F_1')$, one gets trivially $G = T_j$ and $(c_1, c_2, \dots, c_d) = (1, 0, \dots, 0)$.

Example 20 We activate Proposition 1 to apply Zeilberger's creative telescoping paradigm. Take the summand $F(n, k)$ defined in

In[24]:= $\mathbf{F = \frac{(-1)^k}{k}\binom{n}{k}\sum_{i=1}^{k}\frac{1}{i}\sum_{j=1}^{i}\frac{1}{j \quad n};}$

and define the definite sum

In[25]:= **definiteSum = SigmaSum[F, {k, 1, n}]**

Out[25]= $\sum_{k=1}^{n}\frac{(-1)^k}{k}\binom{n}{k}\sum_{i=1}^{k}\frac{1}{i}\sum_{j=1}^{i}\frac{1}{j \quad n}$

Then we can compute a linear recurrence for `SUM[n]=definiteSum` with the call

In[26]:= **rec = GenerateRecurrence[mySum, n, SimplifyByExt → None]**

Out[26]= $\Big\{(1 \quad n)^3(8 \quad 3n)^2\mathrm{SUM}[n] \quad \big(-1692 - 4306n - 4369n^2 - 2202n^3 - 549n^4 - 54n^5\big)\mathrm{SUM}[n \quad 1]$
$(7 \quad 3n)\big(554 \quad 1072n \quad 764n^2 \quad 237n^3 \quad 27n^4\big)\mathrm{SUM}[n \quad 2]$
$-2(3 \quad n)^2(5 \quad 2n)(5 \quad 3n)^2\mathrm{SUM}[n \quad 3] == \frac{808 \quad 2008n \quad 2007n^2 \quad 1017n^3 \quad 261n^4 \quad 27n^5}{(2 \quad n)^2(3 \quad n)}\Big\}$

Here `Sigma` searches for a solution of (3) with $d = 0, 1, 2, \dots$ and finally computes a solution for $d = 3$. Internally, it takes the shifted versions $F(n \quad i, k)$ with $i = 0, 1, 2, 3$

In[27]:= **FList = {f, (f/.n → n 1), (f/.n → n 2), (f/.n → n 3)};**

and rewrites the expressions in a σ-reduced representation:

In[28]:= **fListRed = SigmaReduce[fList, k]**

Out[28]= $\Big\{\frac{(-1)^k}{k}\binom{n}{k}\sum_{i=1}^{k}\frac{1}{i}\sum_{j=1}^{i}\frac{1}{j \quad n},$
$(-1)^k\binom{n}{k}\Big(\frac{1}{(1 \quad n)(1-k \quad n)(1 \quad k \quad n)} \quad \frac{1}{k(-1 \quad k-n)}\sum_{i=1}^{k}\frac{1}{i \quad n} \quad \frac{(-1-n)}{k(-1 \quad k-n)}\sum_{i=1}^{k}\frac{1}{i}\sum_{j=1}^{i}\frac{1}{n \quad j}\Big), \dots\Big\}$

Here we have printed only the first two entries of the output list. Afterwards it activates Proposition 1 by executing the command

In[29]:= **ParameterizedTelescoping[FListRed, n]**

```
Out[29]= {{0, 0, 0, 0, 1}, {c1, c2, c3, c4, G}}
```

The expressions c_1, c_2, c_3, c_4 and G equal

$$\begin{aligned}
c_1 &= -(1 \quad n)^3(8 \quad 3n)^2,\\
c_2 &= 1692 \quad 4306n \quad 4369n^2 \quad 2202n^3 \quad 549n^4 \quad 54n^5,\\
c_3 &= -(7 \quad 3n)(554 \quad 1072n \quad 764n^2 \quad 237n^3 \quad 27n^4),\\
c_4 &= 2(3 \quad n)^2(5 \quad 2n)(5 \quad 3n)^2,\\
G(n,k) &= (-1)^k \binom{n}{k}\left(Q_1 \sum_{i=1}^{k} \frac{1}{i} \sum_{j=1}^{i} \frac{1}{n \quad j} \quad Q_2 \sum_{i=1}^{k} \frac{1}{i \quad n} \quad Q_3\right)
\end{aligned}$$

for some $Q_1, Q_2, Q_3 \in \mathbb{Q}(n,k)$. Alternatively, `ParameterizedTelescoping[fList,k]` (without `SigmaReduce` as a preprocessing step) could be used. The same result could be produced with `CreativeTelescoping[definiteSum,n,SimplifyByExt→ None]`.

Finally, summing (3) with $d = 3$ over k from 0 to n yields the recurrence given in Out[26]. Note that the verification of the correctness of the solution (c_1, c_2, c_3, c_4, G) of (3) with $d = 4$ can be verified straightforwardly: Since W is σ-reduced, one simply has to plug in the solutions and checks that the left-hand and right-hand sides agree. Thus we have shown rigorously that the definite sum given in In[25] is a solution of Out[26].

In order to introduce refined methods, we need the following definition.

Definition 20 Let $W \subset \quad (\mathbb{G})$ be σ-reduced depth-optimal and $\boldsymbol{F}' = (F'_1, \dots, F'_d)$ from $\mathrm{SumProd}_1(W, \mathbb{G})^d$. W is called *$\boldsymbol{F}'$-one complete* if the following holds: If there is $(c_1, \dots, c_d, G) \in \mathbb{K}^* \times \mathbb{K}^{d-1} \times \mathrm{SumProd}_1(\mathbb{G})$ with $\mathtt{d}(G) \leq \min(\mathtt{d}(F'_1), \dots, \mathtt{d}(F'_d))$ such that (27) holds for all n sufficiently large, then there is $G' \in \mathrm{SumProd}_1(W, \mathbb{G})$ with the same c_i such that (27) holds (G replaced by G') for all n sufficiently large.

Using the techniques from [99, 101, 105, 107–110, 114] the following refined parameterized telescoping techniques are available for the class $\mathrm{SumProd}_1(\mathbb{G}_m)$ over a σ-computable field $\mathbb{K}$; for simplicity we skip more general cases, like $\mathrm{SumProd}(\mathbb{G}_r)$.

Problem RPT: Refined Parameterized Telescoping

Given: $F_1, \dots, F_d \in \mathrm{SumProd}_1(\mathbb{G}_m)$.
Find: $\delta \in \mathbb{Z}_{\geq 0}$ and a depth-optimal σ-reduced set $W = \{T_1, \dots, T_e\} \subset \quad {}_1(\mathbb{G}_m)$ with $\mathtt{d}(T_1) \leq \mathtt{d}(T_2) \leq \dots \leq \mathtt{d}(T_e)$ with the following properties:

- One gets $\boldsymbol{F}' = (F'_1, \dots, F'_d) \in \mathrm{SumProd}(W, \mathbb{G}_m)^d$ such that $F_i(k) = F'_i(k)$ holds for all $1 \leq i \leq d$ and $k \geq \delta$.

In addition, based on the refinements given below, one obtains $(c_1, \dots, c_d, G) \in \mathbb{K}^d \times \mathrm{SumProd}(W, \mathbb{G}_m)$ with $c_1 \quad 1$ such that (27) holds for all $k \geq \delta$.

Since W is depth-optimal, it follows in particular that $\mathtt{d}(G') \leq \mathtt{d}(G)$.

- Refinement 1: W is $\boldsymbol{F}'$-one complete. Further, one can compute (it it exists) such a solution with $\mathrm{d}(G) \leq \mathrm{d}(F_1')$ (by using **SimplifyByExt→MinDepth**).
- Refinement 2: If this is not possible, one gets $\mathrm{d}(G) = \mathrm{d}(F_1') \;\; 1$ with the following extra property. $\mathrm{d}(T_{e-1}) < \mathrm{d}(T_e) = \mathrm{d}(G)$ and $T_e = \mathrm{Sum}(\delta, H)$ with $H \in \mathrm{SumProd}(\{T_1, \ldots, T_i\}, \mathbb{G}_m)$ where i with $1 \leq i < e$ is minimal (by using the option **SimplifyByExt→DepthNumber**).
- Refinement 3: One can compute, among all possible choices with i minimal, H such that also $\deg_{T_i}$ is minimal (by using **SimplifyByExt→DepthNumberDegree**).

For technical details concerning `Sigma` we refer to Remarks 7 and 10 above.

Example 21 While the standard approach finds for the definite sum given in In[25] only a recurrence of order 3, the refined parameterized telescoping toolbox (refinement 1) computes a recurrence of order 1:

In[30]:= **GenerateRecurrence[mySum, n, SimplifyByExt → MinDepth]**

Out[30]= $\Big\{\mathrm{SUM}[n] - \mathrm{SUM}[n\;\;1] == \frac{1}{(1\;\;n)^3} - \frac{1}{2(1\;\;n)^2}\sum_{i=0}^{n}\frac{(-1)^i\binom{n}{i}}{1\;\;i\;\;n}\;\;\frac{1}{1\;\;n}\sum_{i=1}^{n}\frac{(-1)^i\binom{n}{i}\sum_{j=1}^{i}\frac{1}{n\;\;j}}{i}\Big\}$

by introducing in addition the sum $\sum_{i=0}^{n}\frac{(-1)^i\binom{n}{i}}{1\;\;i\;\;n}$. The right-hand side is given by definite sums which are simpler than the input sum. In this situation, they can be simplified further to

$$\frac{1}{(1\quad n)^3} - \frac{1}{2(1\quad n)^2(1\quad 2n)}\frac{1}{\binom{2n}{n}}\quad\frac{1}{1\quad n}\sum_{i=1}^{n}\frac{1}{i^2} - \frac{3}{1\quad n}\sum_{i=1}^{n}\frac{1}{i^2\binom{2i}{i}}$$

in $\mathrm{SumProd}_1(\mathbb{Q}(x))$ by applying again the creative telescoping paradigm plus recurrence solving (which we will introduce in the next subsection).

This refined version turns out to be highly valuable in concrete applications. First, one can discover in many problems the minimal recurrence relation. Sometimes this enables one even to read off hypergeometric series solutions, like, e.g., in [87]. In addition, the calculation of such recurrences with shorter order are more efficient, and the extra time to simplify the more complicated right hand sides is often negligible. In applications from particle physics, like in [10, 12], the standard approach is even out of scope and only our improved methods produced the desired results.

Remark 11 (1) Structural theorems (together with algorithmic versions) that are strongly related to Liouville's theorem of integration [79, 95] can be found in [109]. (2) Based on Theorems 1 and 10 additional aspects on the algebraic independence of indefinite nested sums (related to [62]) are worked out in [108] and [117, Section 7.2]. Namely, if there is no solution of a parameterized telescoping solution (in particular of a creative telescoping solution), then the indefinite sums defined over these parameters are algebraically independent.

5.3 Recurrence solving

Finally, we turn to difference ring algorithms that solve parameterized higher-order linear difference equations. Let $(\mathbb{A},\sigma)$ be a difference ring with constant field $\mathbb{K}$, $\boldsymbol{a}=(a_0,\dots,a_m)\in\mathbb{A}^{m\ \ 1}$ and $\boldsymbol{f}=(f_1,\dots,f_d)\in\mathbb{A}^d$. Then we define [65]

$$V(\boldsymbol{a},\boldsymbol{f},\mathbb{A})=\{(c_1,\dots,c_d,g)\in\mathbb{K}^d\times\mathbb{A}\mid \\ a_m\,\sigma^m(g)\quad\cdots\quad a_1\,\sigma(g)\quad a_0\,g=c_1\,f_1\quad\cdots\quad c_d\,f_d\};$$

note that we have $V((1,-u),\boldsymbol{f},\mathbb{A})=V_1(u,\boldsymbol{f},\mathbb{A})$.

In `Sigma` algorithms are available to solve parameterized linear difference equations that are based on the following Theorem 13.

Theorem 13 *Let $(\mathbb{E},\sigma)$ be a basic R   -extension of a   -field $(\mathbb{F},\sigma)$ over $\mathbb{K}$, $\boldsymbol{a}=(a_0,\dots,a_m)\in\mathbb{F}^{m\ \ 1}$ with $a_0,a_m\in\mathbb{F}^*$ and $\boldsymbol{f}\in\mathbb{E}^d$. Then the following holds:*

1. *$V(\boldsymbol{a},\boldsymbol{f},\mathbb{E})$ is a $\mathbb{K}$-vector space of dimensions $\leq m\quad d$.*
2. *If $\mathbb{K}$ is σ-computable, then one can compute a basis of $V(\boldsymbol{a},\boldsymbol{f},\mathbb{E})$.*

Proof (1) follows by a slight variant of [97, Prop 3.1.1] and [57, Thm. XII (page 272)]. By [22, Theorem 9] (based on [20, 49, 90, 98, 100, 104]) the statement (2) holds for $\boldsymbol{f}\in\mathbb{F}^n$. Thus with [17] statement (2) holds also for $\boldsymbol{f}\in\mathbb{E}^n$. □

In addition, `Sigma` contains a solver that finds all hypergeometric solutions in the setting of   -fields. This result follows by Theorems 9 and 10 of [22], which can be considered as the differential version of Singer's celebrated algorithm [122] that finds Liouvillian solutions of linear differential equations with Liouvillian coefficients.

Theorem 14 *Let $(\mathbb{F},\sigma)$ be a   -field over a σ-computable $\mathbb{K}$. Let $a_0,\dots,a_m\in\mathbb{F}$ with $a_0\,a_m\quad 0$. Then one can compute a P-extension $(\mathbb{E},\sigma)$ of $(\mathbb{F},\sigma)$ with $\mathbb{E}=\mathbb{F}\langle t_1\rangle\dots\langle t_e\rangle$ and $\frac{\sigma(t_i)}{t_i}\in\mathbb{F}^*$ and finite sets $S_i\subset\mathbb{F}^*$ for $1\leq i\leq e$ as follows.*

1. *For any $1\leq i\leq e$ and any $h\in S_i$ it follows that $g=ht_i$ is a solution of*

$$a_m\,\sigma^m(g)\quad\cdots\quad a_1\,\sigma(g)\quad a_0\,g=0. \tag{29}$$

2. *For any difference ring extension $(\mathbb{H},\sigma)$ of $(\mathbb{F},\sigma)$ with $\mathbb{H}=\mathbb{F}\langle p_1\rangle\dots\langle p_u\rangle$ and $\frac{\sigma(p_i)}{p_i}\in\mathbb{F}^*$ and any solution $g\in\mathbb{H}$ of (29) with $\alpha=\frac{\sigma(g)}{g}\in\mathbb{F}^*$ one can take $i\in\{1,\dots,e\}$ with $f_1,\dots,f_l\in S_i$ and $c_1,\dots,c_l\in\mathbb{K}^*$ such that $\frac{\sigma(g')}{g'}=\alpha$ holds for $g'=(c_1\,h_1\quad\cdots\quad c_l\,h_l)t_i$.*

We note that the obtained solver of hypergeometric solutions covers the special cases $\mathbb{G}_r$ (see [90, 92]), $\mathbb{G}_b$ with $v=1$ (see [23]) and $\mathbb{G}_m$ (see [31]).

Remark 12 Theorems 13 and 14 hold in the more general setting that $(\mathbb{F},\sigma)$ is a   -field extension of a difference field $(\mathbb{F}_0,\sigma)$ where certain properties are satisfied (see [22, Def. 7]). In a addition, there is a generalization of Theorem 13 given in [17] (based on [22, 117]) where the a_i (with some extra properties) can be taken from $\mathbb{E}$.

Based on [24, 26] we obtain the following result to find all d'Alembertian solutions, a subclass of Liouvillian solutions [63]. The solver relies on [43, Cor 2.1] and [97, Alg. 4.5.3] and the algorithmic machinery of Theorems 14 and 13.

Problem PLDE: Solving Parameterized Linear Difference Equations

Given: $a_0, \dots, a_m \in \mathbb{G}$ and $F_1, \dots, F_d \in \mathrm{SumProd}_1(\mathbb{G})$ with $\mathbb{G} \in \{\mathbb{G}_r, \mathbb{G}_b, \mathbb{G}_m\}$, i.e., $\mathbb{G} = \mathbb{K}(x, x_1, \dots, x_v)$ (or $\mathbb{G} = \mathbb{K}(x_1, \dots, x_v)$) where $\mathbb{K} = \mathcal{A}(y_1, \dots, y_o)(q_1, \dots, q_v)$ is a rational function over an algebraic number field $\mathcal{A}$.

Find: $\delta \in \mathbb{Z}_{\geq 0}$, a finite σ-reduced set $W \subset \quad _1(\mathbb{G}')$ and $B = \{(c_{i,1}, \dots, c_{i,d}, G_i)\}_{1 \leq i \leq \nu} \subseteq \mathbb{K}^d \times \mathrm{SumProd}(W, \mathbb{G}')$ such that

$$a_m(n)\, G_i(n \quad m) \quad \cdots \quad a_0(n) G_i(n) = c_{i,1} F_1(n) \quad \cdots \quad c_{i,d} F_d(n)$$

holds for all $n \geq \delta$ with $1 \leq i \leq \nu$; here $\mathbb{G}' = \mathbb{K}'(x, x_1, \dots, x_v)$ (or $\mathbb{G}' = \mathbb{K}'(x_1, \dots, x_v)$) with $\mathbb{K}' = \mathcal{A}'(y_1, \dots, y_o)(q_1, \dots, q_v)$ where $\mathcal{A}'$ is an algebraic field extension of $\mathcal{A}$.

In addition, the following properties hold:

1. Completeness: For any $\mathbb{G}'' = \mathbb{K}''(x, x_1, \dots, x_v)$ (or $\mathbb{G}'' = \mathbb{K}''(x_1, \dots, x_v)$) with $\mathbb{K}'' = \mathcal{A}''(y_1, \dots, y_o)(q_1, \dots, q_v)$ where $\mathcal{A}''$ is an algebraic extension of $\mathcal{A}$ and $(c_1, \dots, c_d, G) \in \mathbb{K}^d \times \mathrm{SumProd}_1(\mathbb{G}'')$ with
$$(a_m(n)\, G(n \quad m) \quad \cdots \quad a_0(n) G(n))_{n \geq 0} = (c_1 F_1(n) \quad \cdots \quad c_d F_d(n))_{n \geq 0}$$
there is a $(\kappa_1, \dots, \kappa_\nu) \in (\mathbb{K}'')^\nu$ with
$$(c_1, \dots, c_d) = \kappa_1 (c_{1,1}, \dots, c_{1,d}) \quad \cdots \quad \kappa_\nu (c_{\nu,1}, \dots, c_{\nu,d}),$$
$$(G(n))_{n \geq 0} = (\kappa_1 G_1(n) \quad \cdots \quad \kappa_\nu G_d(n))_{n \geq 0}.$$
2. Linear independence: If there is a $(\kappa_1, \dots, \kappa_\nu) \in (\mathbb{K}')^\nu$ with
$$\kappa_1 (c_{1,1}, \dots, c_{1,d}) \quad \cdots \quad \kappa_\nu (c_{\nu,1}, \dots, c_{\nu,d}) = 0,$$
$$(\kappa_1 G_1(n) \quad \cdots \quad \kappa_\nu G_d(n))_{n \geq 0} = \mathbf{0},$$
then $(\kappa_1, \dots, \kappa_\nu) = \mathbf{0}$.

Remark 13 `Sigma` works also for nested products, i.e., $F_1, \dots, F_d \in \mathrm{SumProd}(\mathbb{G})$, if the F_i can be expressed straightforwardly in an R -extension. Using in addition the package `NestedProducts` this toolbox works also fully algorithmically for the case $\mathrm{SumProd}(\mathbb{G}_r)$.

Example 22 We proceed with the calculations given in Example 20. We apply our solver in `Sigma` to the already computed recurrence Out[26] and get

```
In[31]:= recSol = SolveRecurrence[rec[[1]], SUM[n]]
```

Out[31]= $\Big\{\{0,1\},\{0,\sum_{i=1}^{n}\frac{1}{i}\},\{0,\frac{4}{9}\sum_{i=1}^{n}\frac{i!^2}{i^3(2i)!}\;\;\frac{4}{3}\Big(\sum_{i=1}^{n}\frac{1}{i}\Big)\sum_{i=1}^{n}\frac{i!^2}{i^2(2i)!}-\frac{4}{3}\sum_{i=1}^{n}\frac{i!^2\sum_{j=1}^{i}\frac{1}{j}}{i^2(2i)!}\},\{1,-\sum_{i=1}^{n}\frac{\sum_{j=1}^{i}\frac{1}{j^2}}{i}\}\Big\}$

Remark 14 (1) By default the found solutions are represented in a depth-optimal σ-reduced set W to keep the nesting depth of the solutions as small as possible.
(2) Since all components (i.e., a_i, F_i, G_i) can be represented in the given σ-reduced set W, the correctness of the solutions G_i can be verified by plugging them into the recurrence and checking if the left-hand and right-hand side are equal.
(3) If one finds m linearly independent solutions of the homogeneous version together with a particular solution, the solution space is fully determined. In particular, any sequence, which is a solution of the recurrence, can be represented by SumProd($\mathbb{G}$): simply find a linear combination (which is always possible from a certain point on) such that the evaluation of the expression agrees with the first m initial values.

Example 23 We found three linearly independent solutions of the homogeneous version plus one particular solution of the inhomogeneous recurrence, i.e., we found all solutions. Since also the definite sum given in In[25] is a solution of the recurrence, we can combine the solutions accordingly and get an alternative solution of the input sum. This final computation can be carried out as follows:

In[32]:= **sol = FindLinearCombination[recSol, definiteSum, n, 3]**

Out[32]= $3\Big(\sum_{i=1}^{n}\frac{1}{i}\Big)\sum_{i=1}^{n}\frac{i!^2}{i^2(2i)!}-\sum_{i=1}^{n}\frac{\sum_{j=1}^{i}\frac{1}{j^2}}{i}-3\sum_{i=1}^{n}\frac{i!^2\sum_{j=1}^{i}\frac{1}{j}}{i^2(2i)!}\;\;\sum_{i=1}^{n}\frac{i!^2}{i^3(2i)!}$

Finally, we can rewrite the result in terms of the the central binomial coefficient with

In[33]:= **sol = SigmaReduce[sol, n, Tower → {SigmaBinomial[2n, n]}]**

Out[33]= $3\Big(\sum_{i=1}^{n}\frac{1}{i}\Big)\sum_{i=1}^{n}\frac{1}{i^2\binom{2i}{i}}\;\;\sum_{i=1}^{n}\frac{1}{i^3\binom{2i}{i}}-\sum_{i=1}^{n}\frac{\sum_{j=1}^{i}\frac{1}{j^2}}{i}-3\sum_{i=1}^{n}\frac{\sum_{j=1}^{i}\frac{1}{j}}{i^2\binom{2i}{i}}$

Summarizing we have discovered and proved the identity

$$\sum_{k=1}^{n}\frac{(-1)^k\binom{n}{k}}{k}\sum_{i=1}^{k}\frac{\sum_{j=1}^{i}\frac{1}{j\;\;n}}{i}=3\Big(\sum_{i=1}^{n}\frac{1}{i}\Big)\sum_{i=1}^{n}\frac{1}{i^2\binom{2i}{i}}-\sum_{i=1}^{n}\frac{\sum_{j=1}^{i}\frac{1}{j^2}}{i}-3\sum_{i=1}^{n}\frac{\sum_{j=1}^{i}\frac{1}{j}}{i^2\binom{2i}{i}}\;\;\sum_{i=1}^{n}\frac{1}{i^3\binom{2i}{i}}.$$

Example 24 More generally, using the algorithms from [22] we can solve recurrences where the coefficients are represented within a -field. E.g., for the recurrence

In[34]:= **recFactorial = −F[n 2] (1 n)(8 9n 2n²)n!F[n 1] – 2(1 n)³(3 n)n!²F[n] = 0;**

where the coefficients are taken from $\text{SumProd}_1(\mathbb{Q}(x))$, we can find all its solutions (in this instance, they are again from $\text{SumProd}_1(\mathbb{Q}(x))$) by executing the `Sigma`-call

In[35]:= **SolveRecurrence[recFactorial, F[n]]**

Out[35]= $\Big\{\{0,\prod_{i=1}^{n}i!\},\{0,-2^n n!\prod_{i=1}^{n}i!\;\;\frac{3}{2}\prod_{i=1}^{n}i!\sum_{i=1}^{n}2^i i!\}\Big\}$

6 Application: Evaluation of Feynman integrals

The elaborated summation tools from above contributed to highly non-trivial applications, e.g., in the research areas for combinatorics, number theory and particle physics. Here we emphasize the following striking aspects that are most relevant for the treatment of Feynman integrals.

Multi-summation. In order to support the user for the evaluation of definite multi-sums to expressions in SumProd($\mathbb{G}$), the package `EvaluateMultiSums` [111, 112]

```
In[36]:= << EvaluateMultiSums.m
```

EvaluateMultiSums by Carsten Schneider © RISC-JKU

has been developed to tackle definite sums in one stroke. It uses as backbone `Sigma` with all the available tools introduced above. E.g., by executing

In[37]:= **EvaluateMultiSum**$\left[\sum_{k=1}^{n}\frac{(-1)^k}{k}\binom{n}{k}\sum_{i=1}^{k}\frac{1}{i}\sum_{j=1}^{i}\frac{1}{j\quad n}, \{\}, \{n\}, \{1\}, \{\infty\}\right]$

Out[37]= $3\Big(\sum_{i=1}^{n}\frac{1}{i}\Big)\sum_{i=1}^{n}\frac{i!^2}{i^2(2i)!}-\sum_{i=1}^{n}\frac{\sum_{j=1}^{i}\frac{1}{j^2}}{i}-3\sum_{i=1}^{n}\frac{i!^2\sum_{j=1}^{i}\frac{1}{j}}{i^2(2i)!}\quad\sum_{i=1}^{n}\frac{i!^2}{i^3(2i)!}$

we reproduce the identity given in Example 23. In particular, it can tackle definite multi-sums by zooming from inside to outside and, in case that this is possible, transforming step wise the sums to expressions in SumProd$_1(\mathbb{G})$. In this way we could treat highly complicated massive 3-loop Feynman integrals. More precisely, using techniques described in [39] these integrals can be transformed to several thousands of multiple sums with summands from Prod$_1(\mathbb{G}_r)$. Afterwards, the package **`SumProduction`** [37, 112] is applied. It combines these sums to few (but large) sums tailored for our summation toolbox. Afterwards the command **`EvaluateMultiSum`** can be applied (without any further interaction) to treat the obtained sums. In the course of these calculations, we treated up to seven fold multi-sums [6] or 4-fold sums with up to 1GB of size [10, 12]. In addition, this package helped significantly to solve problems from combinatorics [75, 118, 120].

In addition the difference field/ring approach described in this article has been united with important parts of the holonomic approach [55, 130] in [102]. While its first main application arose in combinatorics [28], this combined toolbox has been improved further in [43] and enabled us to tackle various multi-sums coming from particle physics [2, 4, 11]. In addition, these improved tools have been applied in [121] to complicated multi-sums that arose in the context of irrationality proofs of $\zeta(4)$. We remark further, that also other multi-sum and integral techniques from [1, 6, 39] have been explored; for further technologies see also [45] and the references therein.

Solving coupled systems. Using integration by parts methods [54, 76] one can represent physical expressions in terms of master integrals which can be calculated by solving recursively defined coupled systems of linear differential equations. Most of these master integrals can be represented in terms of power series. Utilizing the techniques from above, this gives rise to two general tactics to compute the physical expressions in terms of known special functions (in case that his is possible).

Uncoupling and solving the underlying recurrences. In the first approach we uncouple iteratively the systems of linear differential equation using Gerhold's package `OreSys` [60] and can reduce the problem to solve scalar linear differential equations of each master integral $I(x) = \sum_{n=0}^{\infty} F(n)x^n$. In a first step, each linear differential equation can be transformed to a linear recurrence. Applying `Sigma`'s recurrence solver in a second step enables one to decide constructively if the coefficient $F(n)$ can be expressed in terms of $\text{SumProd}_1(\mathbb{G})$. If this is possible for each master integral, one can express also the physical expressions in $\text{SumProd}_1(\mathbb{G})$. Using these technologies implemented in the package `SolveCoupledSystem` [5, 13] (using `Sigma`) we could treat highly non-trivial particle physical problems as given in [2, 3, 32, 34, 35]. Note that there are also other methods available [64, 78] that can solve certain classes of systems. Furthermore, in ongoing investigations non-trivial methods are developed to solve the coupled systems directly without the usage of uncoupling methods; see [30, 80, 93] and the literature therein.

The large moment method. The second highly successful approach is based on the technology [42, 44] implemented within the package `SolveCoupledSystem`. It enables one to produce for the master integrals the first coefficients $F(n)$ with $n = 0, \dots, \mu$; so far we entered cases where $\mu = 10.000$ was necessary. Here one does not solve the arsing recurrences as proposed above, but uses them to produce a large number of sequence values; as starting point one needs in addition a few initial values that can be produced by our summation tools or procedures like `Mincer` [77] or `MATAD` [124]. A significant feature of the large moment is that one can avoid complicated function spaces (either nested sums with high weight or new classes, like nested sums over, e.g., elliptic functions [7, 27, 46, 47]) during the calculation. Only in the last step, one combines all the calculations and gets large moments of the physical expressions. Then one can use, e.g., the package `ore_algebra` [68] in Sage to guess recurrences (so far up to order 40) that specify precisely the different color factors. Finally, one can decide algorithmically if the physical problem (or individual color factors) can be represented within the class $\text{SumProd}_1(\mathbb{G}_r)$. In this way we could compute, e.g., the 3-loop splitting functions [3], the polarized 3-loop anomalous dimensions [33] and the massive 2- and 3-loop form factor [8, 41]; for another case study see, e.g., [38].

7 Conclusion

We presented two different layers to treat the class of indefinite nested sums defined over nested products in the context of symbolic summation. First, the term algebra layer $\text{SumProd}(\mathbb{G})$ (covering the rational case $\mathbb{G} = \mathbb{G}_r$, the multibasic case $\mathbb{G} = \mathbb{G}_b$ and the mixed multibasic case $\mathbb{G} = \mathbb{G}_m$) equipped with an evaluation function $\text{ev} : \text{SumProd}(\mathbb{G}) \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ has been introduced. There the user can define, evaluate and manipulate the class of nested sums and products conveniently. In particular, we illustrated how this user interface is implemented within the summation package `Sigma`.

Second, the formal difference ring/field layer has been elaborated. Here the elements of $\mathrm{SumProd}(\mathbb{G})$ are rephrased in a ring $\mathbb{E}$ that is built by polynomial ring extensions. More precisely, the adjoined variables (in some instances factored out by ideals) represent the summation objects with two extra ingredients: a ring automorphism $\sigma : \mathbb{E} \to \mathbb{E}$ that describes the action of the shift operator on the ring elements and an evaluation function $\mathrm{ev} : \mathbb{E} \times \mathbb{Z}_{\geq 0} \to \mathbb{K}$ that allows to evaluate the formal ring elements to sequences. In this formal setting one cannot only develop and implement complicated summation algorithms, but one can set up a summation theory that enables one to embed the formal ring extensions into the ring of sequences (see Theorem 1). One of the secrets why the summation package `Sigma`

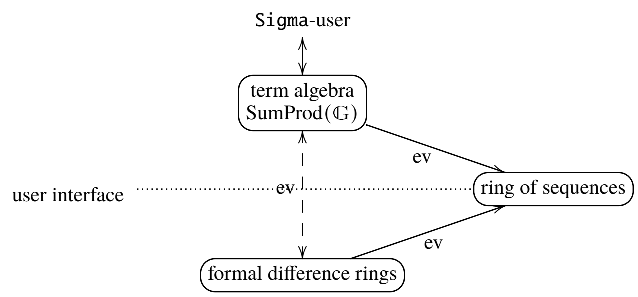

Fig. 1 The symbolic summation framework for difference rings and fields

is successfully used, e.g., within particle physics, combinatorics and number theory, is the smooth interaction between these two different layers: as illustrated in Figure 1 one can represent the objects between the two worlds such that their interpretation with the corresponding evaluation function agrees. In this article, we worked out in detail this algorithmic translation and back translation between the user-friendly term algebra and the complicated difference ring setting. To gain a better understanding of `Sigma`'s capabilities we established a precise input-output specification of the available summation tools using the introduced term algebra language. Special emphasis has been put on the canonical form representation (and their relation to the difference ring theory) for the class $\mathrm{SumProd}(\mathbb{G})$.

References

[1] J. Ablinger. Extensions of the AZ-algorithm and the package MultiIntegrate. In J. Blümlein and C. Schneider, editors, *To appear in Anti-Differentiation and the Calculation of Feynman Amplitudes*, Texts and Monographs in Symbolic Computuation. Springer, 2021. arXiv:2101.11385.

[2] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. The 3-loop pure singlet heavy flavor contributions to the structure function $F_2(x, Q^2)$ and

the anomalous dimension. *Nucl. Phys. B*, 890:48–151, 2014. arXiv:1409.1135.

[3] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. The three-loop splitting functions $P_{qg}^{(2)}$ and $P_{gg}^{(2,N_F)}$. *Nucl. Phys. B*, 922:1–40, 2017. arxiv:1705.01508.

[4] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, M. Round, C. Schneider, and F. Wißbrock. Heavy flavour corrections to polarised and unpolarised deep-inelastic scattering at 3-loop order. In *Proceedings of QCD Evolution 2016*, volume PoS(QCDEV2016)052, pages 1–16, 2016. arXiv:1611.01104.

[5] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, and C. Schneider. Algorithms to solve coupled systems of differential equations in terms of power series. In J. Blümlein, P. Marquard, and T. R. (ed.), editors, *Proc. Loops and Legs in Quantum Field Theory - LL 2016*, volume PoS(LL2016)005, pages 1–15, 2016. arXiv:1608.05376.

[6] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. Calculating three loop ladder and V-topologies for massive operator matrix elements by computer algebra. *Comput. Phys. Comm.*, 202:33–112, 2016. arXiv:1509.08324.

[7] J. Ablinger, J. Blümlein, A. D. Freitas, M. van Hoeij, E. Imamoglu, C. Raab, C.-S. Radu, and C. Schneider. Iterated elliptic and hypergeometric integrals for Feynman diagrams. *J. Math. Phys.*, 59(062305):1–55, 2018. arXiv:1706.01299.

[8] J. Ablinger, J. Blümlein, P. Marquard, N. Rana, and C. Schneider. Heavy quark form factors at three loops in the planar limit. *Phys. Lett. B*, 782:528–532, 2018. arXiv:1804.07313.

[9] J. Ablinger, J. Blümlein, C. G. Raab, and C. Schneider. Iterated binomial sums and their associated iterated integrals. *J. Math. Phys.*, 55(112301):1–57, 2014. arXiv:1407.1822.

[10] J. Ablinger, J. Blümlein, A. De Freitas, A. Goedicke, C. Schneider, and K. Schönwald. The two-mass contribution to the three-loop gluonic operator matrix element $A_{gg,Q}^{(3)}$. *Nucl. Phys. B*, 932:129–240, 2018. arXiv:1804.02226.

[11] J. Ablinger, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, M. Round, C. Schneider, and F. Wissbrock. The transition matrix element $A_{gq}(N)$ of the variable flavor number scheme at $O(\alpha_s^3)$. *Nucl. Phys. B*, 882:263–288, 2014. arXiv:1402.0359.

[12] J. Ablinger, J. Blümlein, A. De Freitas, M. Saragnese, C. Schneider, and K. Schönwald. The three-loop polarized pure singlet operator matrix element with two different masses. *Nuclear Physics B*, 952(114916):1–18, 2020. arXiv:1911.11630.

[13] J. Ablinger, J. Blümlein, A. De Freitas, and C. Schneider. A toolbox to solve coupled systems of differential and difference equations. In N. Glover, D. Maitre, and B. Pecjak, editors, *Proc. of the 13th International Symposium on Radiative Corrections (Applications of Quantum Field Theory to Phenomenology)*, volume PoS(RADCOR2015)060, pages 1–13, 2016. arXiv:1601.01856.

[14] J. Ablinger, J. Blümlein, and C. Schneider. Harmonic sums and polylogarithms generated by cyclotomic polynomials. *J. Math. Phys.*, 52(10):1–52, 2011. arXiv:1007.0375.

[15] J. Ablinger, J. Blümlein, and C. Schneider. Analytic and algorithmic aspects of generalized harmonic sums and polylogarithms. *J. Math. Phys.*, 54(8):1–74, 2013. arXiv:1302.0378.

[16] J. Ablinger and C. Schneider. Algebraic independence of sequences generated by (cyclotomic) harmonic sums. *Annals of Combinatorics*, 22(2):213–244, 2018. arXiv:1510.03692.

[17] J. Ablinger and C. Schneider. Solving linear difference equations with coefficients in rings with idempotent representations, 2021. in preperation.

[18] S. A. Abramov. On the summation of rational functions. *Zh. vychisl. mat. Fiz.*, 11:1071–1074, 1971.

[19] S. A. Abramov. The rational component of the solution of a first-order linear recurrence relation with a rational right-hand side. *U.S.S.R. Comput. Maths. Math. Phys.*, 15:216–221, 1975. Transl. from Zh. vychisl. mat. mat. fiz. 15, pp. 1035–1039, 1975.

[20] 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.

[21] S. A. Abramov and M. Bronstein. Hypergeometric dispersion and the orbit problem. In C. Traverso, editor, *Proc. ISSAC'00*. ACM Press, 2000.

[22] S. A. Abramov, M. Bronstein, M. Petkovšek, and C. Schneider. On rational and hypergeometric solutions of linear ordinary difference equations in *-field extensions. *J. Symb. Comput., in press*, 2021. arXiv:2005.04944.

[23] S. A. Abramov, P. Paule, and M. Petkovšek. q-Hypergeometric solutions of q-difference equations. *Discrete Math.*, 180(1-3):3–22, 1998.

[24] 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, 1994.

[25] S. A. Abramov and M. Petkovšek. Polynomial ring automorphisms, rational (w, σ)-canonical forms, and the assignment problem. *J. Symbolic Comput.*, 45(6):684–708, 2010.

[26] S. A. Abramov and E. V. Zima. D'Alembertian solutions of inhomogeneous linear equations (differential, difference, and some other). In *Proc. ISSAC'96*, pages 232–240. ACM Press, 1996.

[27] L. Adams, C. Bogner, A. Schweitzer, and S. Weinzierl. The kite integral to all orders in terms of elliptic polylogarithms. *J. Math. Phys.*, 57(12):122302, 2016.

[28] 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.

[29] M. Apagodu and D. Zeilberger. Multi-variable Zeilberger and Almkvist-Zeilberger algorithms and the sharpening of Wilf-Zeilberger theory. *Adv. Appl. Math.*, 37:139–152, 2006.

[30] M. Barkatou. On rational solutions of systems for linear differential equations. *J. Symbol. Comput.*, 28:547–567, 1999.

[31] A. Bauer and M. Petkovšek. Multibasic and mixed hypergeometric Gosper-type algorithms. *J. Symbolic Comput.*, 28(4–5):711–736, 1999.

[32] A. Behring, J. Blümlein, G. Falcioni, A. De Freitas, A. von Manteuffel, and C. Schneider. The asymptotic 3-loop heavy flavor corrections to the charged current structure functions $F_L^{W\ -W^-}(x, Q^2)$ and $F_2^{W\ -W^-}(x, Q^2)$. *Phys. Rev. D*, 94(11):1–19, 2016. arXiv:1609.06255.

[33] A. Behring, J. Blümlein, A. De Freitas, A. Goedicke, S. Klein, A. von Manteuffel, C. Schneider, and K. Schönwald. The polarized three-loop anomalous dimensions from on-shell massive operator matrix elements. *Nuclear Physics B*, 948(114753):1–41, 2019. arXiv:1908.03779.

[34] A. Behring, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, and C. Schneider. The $O(\alpha_s^3)$ heavy flavor contributions to the charged current structure function $xF_3(x, Q^2)$ at large momentum transfer. *Phys. Rev. D*, 92(114005):1–19, 2015. arXiv:1508.01449.

[35] A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. The 3-loop non-singlet heavy flavor contributions to the structure function $g_1(x, Q^2)$ at large momentum transfer. *Nuclear Phys. B*, 897:612–644, 2015. arXiv:1504.08217.

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

[37] J. Blümlein, A. Hasselhuhn, and C. Schneider. Evaluation of multi-sums for large scale problems. In *Proceedings of RADCOR 2011*, volume PoS(RADCOR2011)32, pages 1–9, 2012. arXiv:1202.4303.

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

[39] J. Blümlein, S. Klein, C. Schneider, and F. Stan. A symbolic summation approach to Feynman integral calculus. *J. Symbolic Comput.*, 47:1267–1289, 2012. arXiv:1011.2656.

[40] J. Blümlein and S. Kurth. Harmonic sums and Mellin transforms up to two-loop order. *Phys. Rev.*, D60, 1999.

[41] J. Blümlein, P. Marquard, N. Rana, and C. Schneider. The heavy fermion contributions to the massive three loop form factors. *Nuclear Physics B*, 949(114751):1–97, 2019. arXiv:1908.00357.

[42] J. Blümlein, P. Marquard, and C. Schneider. A refined machinery to calculate large moments from coupled systems of linear differential equations. In D. Kosower and M. Cacciari, editors, *14th International Symposium on Radiative Corrections (RADCOR2019)*, POS(RADCOR2019)078, pages 1–13, 2020. arXiv:1912.04390.
[43] 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.
[44] J. Blümlein and C. Schneider. The method of arbitrarily large moments to calculate single scale processes in quantum field theory. *Physics Letters B*, 771:31–36, 2017. arXiv:1701.04614.
[45] J. Blümlein and C. Schneider. Analytic computing methods for precision calculations in quantum field theory. *INTERNATIONAL JOURNAL OF MODERN PHYSICS A (IJMPA)*, 33(1830015):1–35, 2018. arXiv:1809.02889.
[46] J. Blümlein, P. Paule, and C. Schneider, editors. *Elliptic Integrals, Elliptic Functions and Modular Forms in Quantum Field Theory*, Texts & Monographs in Symbolic Computation. Springer, 2019.
[47] J. Broedel, C. Duhr, F. Dulat, and L. Tancredi. Elliptic polylogarithms and iterated integrals on elliptic curves. II. an application to the sunrise integral. *Phys. Rev.*, D97(116009), 2018.
[48] M. Bronstein. *Symbolic Integration I, Transcendental functions*. Springer, Berlin-Heidelberg, 1997.
[49] M. Bronstein. On solutions of linear ordinary difference equations in their coefficient field. *J. Symbolic Comput.*, 29(6):841–877, 2000.
[50] B. Buchberger and R. Loos. Algebraic simplification. In *Computer algebra*, pages 11–43. Springer, Vienna, 1983.
[51] S. Chen, R. Feng, G. Fu, and Z. Li. On the structure of compatible rational functions. In *Proceedings of ISSAC 2011*, pages 91–98, 2011.
[52] S. Chen, M. Jaroschek, M. Kauers, and M. F. Singer. Desingularization explains order-degree curves for ore operators. In M. Kauers, editor, *Proc. of ISSAC'13*, pages 157–164, 2013.
[53] W. Y. C. Chen, Q.-H. Hou, and H.-T. Jin. The Abel-Zeilberger algorithm. *Electron. J. Combin.*, 18(2):Paper 17, 17, 2011.
[54] K. G. Chetyrkin and F. V. Tkachov. Integration by parts: The algorithm to calculate beta functions in 4 loops. *Nucl. Phys. B*, 192:159–204, 1981.
[55] F. Chyzak. An extension of Zeilberger's fast algorithm to general holonomic functions. *Discrete Math.*, 217:115–134, 2000.
[56] F. Chyzak, M. Kauers, and B. Salvy. A non-holonomic systems approach to special function identities. In J. May, editor, *Proceedings of ISSAC'09*, pages 111–118, 2009.
[57] R. M. Cohn. *Difference Algebra*. John Wiley & Sons, 1965.
[58] A. I. Davydychev and M. Yu. Kalmykov. Massive Feynman diagrams and inverse binomial sums. *Nucl. Phys. B*, 699:3–64, 2004.
[59] G. Ge. *Algorithms related to the multiplicative representation of algebraic numbers*. PhD thesis, Univeristy of California at Berkeley, 1993.
[60] S. Gerhold. Uncoupling systems of linear ore operator equations. Master's thesis, RISC, J. Kepler University Linz, 2 2002.
[61] R. W. Gosper. Decision procedures for indefinite hypergeometric summation. *Proc. Nat. Acad. Sci. U.S. A.*, 75:40–42, 1978.
[62] C. Hardouin and M. F. Singer. Differential Galois theory of linear difference equations. *Math. Ann.*, 342(2):333–377, 2008.
[63] P. A. Hendriks and M. F. Singer. Solving difference equations in finite terms. *J. Symbolic Comput.*, 27(3):239–259, 1999.
[64] J. M. Henn. Multiloop integrals in dimensional regularization made simple. *Phys. Rev. Lett.*, 110:251601, 2013.
[65] M. Karr. Summation in finite terms. *J. ACM*, 28:305–350, 1981.

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

[67] M. Kauers. Summation Algorithms for Stirling Number Identities. *Journal of Symbolic Computation*, 42(10):948–970, October 2007.

[68] M. Kauers, M. Jaroschek, and F. Johansson. Ore polynomials in Sage. In J. Gutierrez, J. Schicho, and M. Weimann, editors, *Computer Algebra and Polynomials*, Lecture Notes in Computer Science, pages 105–125, 2014.

[69] M. Kauers and C. Schneider. Application of unspecified sequences in symbolic summation. In J. Dumas, editor, *Proc. ISSAC'06.*, pages 177–183. ACM Press, 2006.

[70] M. Kauers and C. Schneider. Indefinite summation with unspecified summands. *Discrete Math.*, 306(17):2021–2140, 2006.

[71] M. Kauers and C. Schneider. Symbolic summation with radical expressions. In C. Brown, editor, *Proc. ISSAC'07*, pages 219–226, 2007.

[72] M. Kauers and C. Schneider. Automated Proofs for Some Stirling Number Identities. *The Electronic Journal of Combinatorics*, 15(1):1–7, 2008. R2.

[73] M. Kauers and B. Zimmermann. Computing the algebraic relations of c-finite sequences and multisequences. *Journal of Symbolic Computation*, 43(11):787–803, 2008.

[74] C. Koutschan. Creative telescoping for holonomic functions. In C. Schneider and J. Blümlein, editors, *Computer Algebra in Quantum Field Theory: Integration, Summation and Special Functions*, Texts and Monographs in Symbolic Computation, pages 171–194. Springer, 2013. arXiv:1307.4554.

[75] C. Krattenthaler and C. Schneider. Evaluation of binomial double sums involving absolute values. In V. Pillwein and C. Schneider, editors, *Algorithmic Combinatorics: Enumerative Combinatorics, Special Functions and Computer Algebra, in Honour of Peter Paule on his 60th Birthday*, Texts and Monographs in Symbolic Computuation, pages 249–295. Springer, 2020. arXiv:1607.05314.

[76] S. Laporta. High precision calculation of multiloop Feynman integrals by difference equations. *Int. J. Mod. Phys.*, A15:5087–5159, 2000.

[77] S. A. Larin, F. Tkachov, and J. A. M. Vermaseren. The FORM version of Mincer. Technical Report NIKHEF-H-91-18, NIKHEF, Netherlands, 1991.

[78] R. N. Lee. Reducing differential equations for multiloop master integrals. *JHEP*, 04:108, 2015.

[79] J. Liouville. Mémoire sur l'intégration d'une classe de fonctions transcendantes. *J. Reine Angew. Math.*, 13:93–118, 1835.

[80] J. Middeke and C. Schneider. Denominator bounds for systems of recurrence equations using -extensions. In C. Schneider and E. Zima, editors, *Advances in Computer Algebra: In Honour of Sergei Abramov's' 70th Birthday*, volume 226 of *Springer Proceedings in Mathematics & Statistics*, pages 149–173. Springer, 2018. arXiv:1705.00280.

[81] S. O. Moch, P. Uwer, and S. Weinzierl. Nested sums, expansion of transcendental functions, and multiscale multiloop integrals. *J. Math. Phys.*, 6:3363–3386, 2002.

[82] I. Nemes and P. Paule. A canonical form guide to symbolic summation. In A. Miola and M. Temperini, editors, *Advances in the Design of Symbolic Computation Systems*, Texts Monogr. Symbol. Comput., pages 84–110. Springer, Wien-New York, 1997.

[83] E. D. Ocansey and C. Schneider. Representing (q-)hypergeometric products and mixed versions in difference rings. In C. Schneider and E. Zima, editors, *Advances in Computer Algebra. WWCA 2016.*, volume 226 of *Springer Proceedings in Mathematics & Statistics*, pages 175–213. Springer, 2018. arXiv:1705.01368.

[84] E. D. Ocansey and C. Schneider. Representation of hypergeometric products of higher nesting depths in difference rings. RISC Report Series 20-19, Research Institute for Symbolic Computation (RISC), Johannes Kepler University Linz, Schloss Hagenberg, 4232 Hagenberg, Austria, 2020. arXiv:2011.08775.

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

[86] P. Paule and A. Riese. A Mathematica q-analogue of Zeilberger's algorithm based on an algebraically motivated approach to q-hypergeometric telescoping. In M. Ismail and

M. Rahman, editors, *Special Functions, q-Series and Related Topics*, volume 14, pages 179–210. AMS, 1997.

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

[88] P. Paule and C. Schneider. Towards a symbolic summation theory for unspecified sequences. In J. Blümlein, P. Paule, and C. Schneider, editors, *Elliptic Integrals, Elliptic Functions and Modular Forms in Quantum Field Theory*, Texts and Monographs in Symbolic Computation, pages 351–390. Springer, 2019. arXiv:1809.06578.

[89] 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.

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

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

[92] M. van Hoeij. Finite singularities and hypergeometric solutions of linear recurrence equations. *J. Pure Appl. Algebra*, 139(1-3):109–131, 1999.

[93] M. van Hoeij, M. Barkatou, and J. Middeke. A family of denominator bounds for first order linear recurrence systems. Technical Report arXiv:2007.02926, 2020.

[94] R. Risch. The problem of integration in finite terms. *Trans. Amer. Math. Soc.*, 139:167–189, 1969.

[95] M. Rosenlicht. Liouville's theorem on functions with elementary integrals. *Pacific J. Math.*, 24:153–161, 1968.

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

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

[98] C. Schneider. A collection of denominator bounds to solve parameterized linear difference equations in -extensions. *An. Univ. Timişoara Ser. Mat.-Inform.*, 42(2):163–179, 2004. Extended version of Proc. SYNASC'04.

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

[100] C. Schneider. Degree bounds to find polynomial solutions of parameterized linear difference equations in -fields. *Appl. Algebra Engrg. Comm. Comput.*, 16(1):1–32, 2005.

[101] C. Schneider. Finding telescopers with minimal depth for indefinite nested sum and product expressions. In M. Kauers, editor, *Proc. ISSAC'05*, pages 285–292. ACM, 2005.

[102] C. Schneider. A new Sigma approach to multi-summation. *Adv. in Appl. Math.*, 34(4):740–767, 2005.

[103] C. Schneider. Product representations in -fields. *Ann. Comb.*, 9(1):75–99, 2005.

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

[105] C. Schneider. Simplifying sums in -extensions. *J. Algebra Appl.*, 6(3):415–441, 2007.

[106] C. Schneider. Symbolic summation assists combinatorics. *Sém. Lothar. Combin.*, 56:1–36, 2007. Article B56b.

[107] C. Schneider. A refined difference field theory for symbolic summation. *J. Symbolic Comput.*, 43(9):611–644, 2008. arXiv:0808.2543v1.

[108] C. Schneider. Parameterized telescoping proves algebraic independence of sums. *Ann. Comb.*, 14:533–552, 2010. arXiv:0808.2596; for a preliminary version see FPSAC 2007.

[109] C. Schneider. Structural theorems for symbolic summation. *Appl. Algebra Engrg. Comm. Comput.*, 21(1):1–32, 2010.

[110] C. Schneider. A symbolic summation approach to find optimal nested sum representations. In A. Carey, D. Ellwood, S. Paycha, and S. Rosenberg, editors, *Motives, Quantum Field Theory, and Pseudodifferential Operators*, volume 12 of *Clay Mathematics Proceedings*, pages 285–308. Amer. Math. Soc, 2010. arXiv:0808.2543.

[111] C. Schneider. Simplifying multiple sums in difference fields. In C. Schneider and J. Blümlein, editors, *Computer Algebra in Quantum Field Theory: Integration, Summation and Special*

Functions, Texts and Monographs in Symbolic Computation, pages 325–360. Springer, 2013. arXiv:1304.4134.
[112] C. Schneider. Modern summation methods for loop integrals in quantum field theory: The packages Sigma, EvaluateMultiSums and SumProduction. In *Proc. ACAT 2013*, volume 523(012037) of *J. Phys.: Conf. Ser.*, pages 1–17, 2014. arXiv:1310.0160.
[113] C. Schneider. A streamlined difference ring theory: Indefinite nested sums, the alternating sign and the parameterized telescoping problem. In F. Winkler, V. Negru, T. Ida, T. Jebelean, D. Petcu, S. Watt, and D. Zaharie, editors, *Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2014 15th International Symposium*, pages 26–33. IEEE Computer Society, 2014. arXiv:1412.2782.
[114] C. Schneider. Fast algorithms for refined parameterized telescoping in difference fields. In M. W. J. Guitierrez, J. Schicho, editor, *Computer Algebra and Polynomials*, number 8942 in Lecture Notes in Computer Science (LNCS), pages 157–191. Springer, 2015. arXiv:1307.7887.
[115] C. Schneider. A difference ring theory for symbolic summation. *J. Symb. Comput.*, 72:82–127, 2016. arXiv:1408.2776.
[116] C. Schneider. Symbolic summation in difference rings and applications. In M. Rosenkranz, editor, *Proc. ISSAC 2016*, pages 9–12, 2016.
[117] C. Schneider. Summation Theory II: Characterizations of R -extensions and algorithmic aspects. *J. Symb. Comput.*, 80(3):616–664, 2017. arXiv:1603.04285.
[118] C. Schneider. The absent-minded passengers problem: A motivating challenge solved by computer algebra. *Mathematics in Computer Science*, 2020. arXiv:2003.01921.
[119] C. Schneider. Minimal representations and algebraic relations for single nested products. *Programming and Computer Software*, 46(2):133–161, 2020. arXiv:1911.04837.
[120] C. Schneider and R. Sulzgruber. Asymptotic and exact results on the complexity of the Novelli–Pak–Stoyanovskii algorithm. *Electron. J. Combin.*, 24(2):1–33, 2017. #P2.28, arXiv:1606.07597.
[121] C. Schneider and W. Zudilin. A case study for $\zeta(4)$. In A. Bostan and K. Raschel, editors, *To appear in Proceedings of the conference 'Transient Transcendence in Transylvania'*, Proceedings in Mathematics & Statistics. Springer, 2021. arXiv:2004.08158.
[122] M. F. Singer. Liouvillian solutions of linear differential equations with Liouvillian coefficients. *J. Symb. Comput.*, 11(3):251–274, 1991.
[123] M. F. Singer. Algebraic and algorithmic aspects of linear difference equations. In C. Hardouin, J. Sauloy, and M. F. Singer, editors, *Galois theories of linear difference equations: An Introduction*, volume 211 of *Mathematical Surveys and Monographs*. AMS, 2016.
[124] M. Steinhauser. MATAD: A Program package for the computation of MAssive TADpoles. *Comput. Phys. Commun.*, 134:335–364, 2001.
[125] M. van der Put and M. Singer. *Galois theory of difference equations*, volume 1666 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1997.
[126] J. A. M. Vermaseren. Harmonic sums, Mellin transforms and integrals. *Int. J. Mod. Phys.*, A14:2037–2976, 1999.
[127] K. Wegschaider. Computer generated proofs of binomial multi-sum identities. Master's thesis, RISC, J. Kepler University, May 1997.
[128] S. Weinzierl. Expansion around half integer values, binomial sums and inverse binomial sums. *J. Math. Phys.*, 45:2656–2673, 2004.
[129] H. Wilf and D. Zeilberger. An algorithmic proof theory for hypergeometric (ordinary and "q") multisum/integral identities. *Invent. Math.*, 108:575–633, 1992.
[130] D. Zeilberger. A holonomic systems approach to special functions identities. *J. Comput. Appl. Math.*, 32:321–368, 1990.
[131] D. Zeilberger. The method of creative telescoping. *J. Symbolic Comput.*, 11:195–204, 1991.