Computer Generated Proofs of Binomial Multi-Sum Identities

Diplomarbeit
zur Erlangung des akademischen Titels
"Diplom-Ingenieur"
in der Studienrichtung
Technische Mathematik

verfaßt von
Kurt Wegschaider

am
Research Institute for Symbolic Computation,
Institut für Mathematik,
Johannes Kepler Universität Linz

eingereicht bei
Univ.-Doz. Dr. Peter Paule

Mai 1997

Contents

Chapter 1

Introduction

1.1 Computer Generated Proofs of Binomial Multisum Identities

One of the most exciting discoveries in the recent years, due to Doron Zeilberger, was that the problem of proving binomial summation identities like Dixon's identity

$$\sum_k (-1)^k \binom{2n}{k}^3 = (-1)^n \frac{(3n)!}{n!^3}$$

can efficiently be handled by the computer. Since then Zeilberger's fast algorithm has become the standard tool to tackle binomial coefficient identities involving a single summation quantifier. The method is so successful that it is already treated in the book "Concrete Mathematics" ([GKP94]), and that recently the introductory textbook "A=B" ([PWZ96]) was published, which is exclusively devoted to this and similar methods. However, there are many binomial sums that involve more than one summation quantifier, and only a few special examples can be treated iteratively by Zeilberger's fast algorithm. Some examples of multisum identities that can not be proved by single summation techniques are the beautiful identity

$$\sum_k \sum_j \binom{n}{k}\binom{n+k}{k}\binom{k}{j}^3 = \sum_k \binom{n}{k}^2 \binom{n+k}{k}^2, \tag{1.1}$$

the Andrews-Paule sum

$$\sum_{i=0}^{n} \sum_{j=0}^{n} \binom{i+j}{i}^2 \binom{4n-2i-2j}{2n-2i} = (2n+1)\binom{2n}{n},$$

and the following identity due to John Essam

$$\sum_{k_1} \sum_{k_2 \le k_1} \sum_{k_3 \le k_2} (k_1-k_2)(k_1-k_3)(k_2-k_3)\binom{n}{k_1}\binom{n}{k_2}\binom{n}{k_3} = n^2(n-1)\,8^{n-2}\frac{(\frac{3}{2})_{n-2}}{(3)_{n-2}},$$

where $(a)_n = a(a+1)\dots(a+n-1)$. In view of these examples we want to develop an efficient algorithm that solves the following definite summation problem: prove or disprove that for every nonnegative integer n

$$\sum_{k_1}\cdots\sum_{k_r} F(n,k_1,\dots,k_r) \;=\; rhs(n), \tag{1.2}$$

where $F(n,k_1,\dots,k_r)$ and $rhs(n)$ are some binomial functions. It is well known, due to Wilf and Zeilberger ([WZ92a]), that in principle such multisums can be handled with a method called *Sister Celine's technique*. But until now the performance of Sister Celine's technique was only good enough to prove relatively simple examples: the above multisums were beyond the power of the method. It is the goal of this thesis to improve and generalize Sister Celine's technique, such that it constitutes an efficient algorithm. We have implemented a Mathematica procedure for generating proofs — contained in our package MultiSum— that is significantly faster than previous implementations. For instance, with our implementation it is quite easy to compute proofs for the above identities. The main algorithmic achievements of this thesis are a new efficient generalization of Sister Celine's technique (Section 3.5) and the utilization of the relatively unknown theory of P. Verbaeten ([Ver76]). Additionally, we are able to fill some gaps in the theory as it is given in [WZ92a], e.g., by giving a complete proof of the fundamental theorem of hypergeometric summation in [WZ92a] (Corollary 3.3).

A recurrence for the sum. The general method to prove an identity of the form (1.2) is to compute a homogeneous linear polynomial recurrence relation for the multiple sum $f(n) = \sum_{k_1}\cdots\sum_{k_r} F(n,k_1,\dots,k_r)$:

$$a_0(n)f(n) \;+\; a_1(n)f(n-1) \;+\; \cdots + \; a_l(n)f(n-l) \;=\; 0. \tag{1.3}$$

To complete the proof, we only have to check that $rhs(n)$ satisfies this recurrence relation (if $rhs(n)$ is a binomial function simply by plugging in). The identity follows by checking that enough initial values of $f(n)$ and $rhs(n)$ are identical, i.e., we must verify that $f(0) = rhs(0),\dots,f(l-1) = rhs(l-1)$, and $f(n) = rhs(n)$ for those n where $a_0(n)$, the leading polynomial coefficient of the recurrence, vanishes. We can also handle the case that $rhs(n)$ is a sum that satisfies a recurrence relation: it is possible to compute a recurrence relation satisfied by the difference of the two sums and again the identity follows from comparing enough initial values.

As a concrete example, we can prove the identity (1.1) by showing that both sums satisfy the following polynomial recurrence relation (see Chapter 5)

$$n^3\,f(n) \;-\; (2n-1)(17n^2-17n+5)\,f(n-1) \;+\; (n-1)^3\,f(n-2) \;=\; 0.$$

If both sides are equal for $n=0$ and $n=1$, then the identity holds for all n.

Creative telescoping. We obtain recurrences for the sum by using Zeilberger's *creative telescoping*. We require that the summand F has compact support, i.e., that for every n there are only finitely many tuples $(k_1,\dots,k_r) \in \mathbb{Z}^r$ where $F(n,k_1,\dots,k_r)$ is nonzero (a sum with *standard boundary conditions*, see Section 3.3), and that F satisfies a recurrence relation w.r.t. the parameter n of the form

$$a_0(n)F(n,k_1,\ldots,k_r)+\cdots+a_l(n)F(n-l,k_1,\ldots,k_r) \;=\; \sum_{l=1}^{r}\Delta_{k_l}G_l(n,k_1,\ldots,k_r), \qquad (1.4)$$

where $a_i(n)$ are polynomials free of the summation variables, and

$$\Delta_{k_l}G_l(n,k_1,\ldots,k_r) \;=\; G_l(n,k_1,\ldots,k_l+1,\ldots,k_r) \;-\; G_l(n,k_1,\ldots,k_r).$$

It is important that the G_l are always multiples of F, either rational function multiples (in Zeilberger's fast algorithm) or polynomial recurrence operator multiples (in Sister Celine's technique, see Section 3.2), so that also G_l has compact support. Summing the recurrence w.r.t. $k_1,\ldots,k_r$, we observe that the Δ-parts telescope and the boundary values are zero, due to compact support. This yields a homogeneous recurrence relation for $f(n)$.

For the case $r=1$, i.e., the case of single summation, the problem of finding a recurrence of the form (1.4) is successfully solved by Zeilberger's fast algorithm ([Zei90a]) that is based on Gosper's algorithm ([Gos78]) for indefinite hypergeometric summation (see also [PWZ96] or [GKP94]). In the few cases where we make use of Zeilberger's fast algorithm in this thesis, we use the reliable and fast Mathematica implementation due to Paule and Schorn ([PS95]).[1] For multiple summation no equally successful method exists. The problem is much harder to solve, and despite all efforts, the running time of the algorithms can be prohibitively high. There are two essentially different methods to compute the recurrence (1.4): the *hypergeometric method* and the *elimination method*. They differ in the way they compute the recurrence and the class of functions they can treat.

The hypergeometric method. The hypergeometric method operates on a hypergeometric summand $F(n,k_1,\ldots,k_r)$, that means the fractions

$$\frac{F(n-1,k_1,\ldots,k_r)}{F(n,k_1,\ldots,k_r)},\ \frac{F(n,k_1-1,\ldots,k_r)}{F(n,k_1,\ldots,k_r)},\ \ldots,\ \frac{F(n,k_1,\ldots,k_r-1)}{F(n,k_1,\ldots,k_r)}$$

are rational functions in $n,k_1,\ldots,k_r$. The class of hypergeometric functions includes most of the binomial summands we are interested in, e.g., binomial coefficients that are integer linear in the variables $n,k_1,\ldots,k_r$. We are able to check that a hypergeometric function satisfies a recurrence: divide the recurrence by the function and we only have to check that a rational function is identically zero. Therefore a recurrence of the form (1.4) constitutes a *computer generated proof*.

To compute a recurrence of the form (1.4) we use *Sister Celine's technique* or in other words the method of k-free recurrences. Sister Celine's technique was invented by M. C. Fasenmyer ([Fas47], [Fas49]), systematically investigated by P. Verbaeten ([Ver76]), and generalized and used for symbolic summation by H. Wilf and D. Zeilberger ([WZ92a]). The method computes a homogeneous polynomial recurrence relation for the summand with the special property that the polynomial coefficients are free of the summation variables (*k-free recurrence*). Every such

[1] available from `http://www.risc.uni-linz.ac.at/research/combinat/risc/software/`

k-free recurrence can be transformed into the form (1.4). In the next chapters we investigate Sister Celine's technique in detail. In Chapter 2 we prove that for a special class of summands a k-free recurrence always exists, and we investigate Verbaeten's theory (especially the important notion of a P-maximal structureset). In Chapter 3 we investigate the transformation into the form (1.4) and we state an efficient generalization of Sister Celine's technique, which is new. This generalization together with the notion of a P-maximal structureset gives rise to a fast and useful implementation, the Mathematica package MultiSum, which is described in Chapter 4. Chapter 5 contains the proofs of several interesting identities.

An alternative approach is the method proposed by Wilf and Zeilberger in [WZ92a], that looks directly for a recurrence of the form (1.4). But the main computational problems are still unsolved, so in practice this approach is not very successful (see Subsection 3.5.2). Another alternative method is due to L. Yen ([Yen93]), who showed that there exist computable upper bounds for the number of initial values we have to compare to establish the identity. This means we do not have to compute a recurrence at all, but unfortunately the upper bounds are far too large to be useful.

The elimination method. The elimination method operates on *P-finite* functions, i.e., functions that are defined by a set of linear difference-differential equations and finitely many initial conditions. The subject of investigation is the left ideal of operators (differential operators, shift operators, or q-shift operators) that annihilates this function. The research was again stimulated by D. Zeilberger, who showed in [Zei90b] that the subclass of *holonomic* functions is closed w.r.t. definite summation. He also showed that one is able to compute, using elimination in the annihilation ideal of the function, a recurrence of the form (1.4) for any holonomic function. Zeilberger used Sylvester's dialytic elimination to compute this recurrence, a method that is rather slow. The work on such annihilation ideals was extended by F. Chyzak and B. Salvy to δ-finite functions in the general setting of Ore-algebras ([CS97]). In this context elimination is performed by using noncommutative Gröbner bases. These δ-finite functions can be handled with F. Chyzak's ([Chy94]) Maple implementation *Mgfun*.[2] It remains to mention that M. Schorn, in the last chapter of his diploma thesis [Sch95], presents an elimination algorithm that computes recurrences for double sums of hypergeometric functions in incredibly short time.

1.2 Notation and Basic Definitions

In this section we define the mathematical notations and the basic definitions used in the thesis.

Numbers. We frequently use the following sets of numbers: the set of all integers $\mathbb{Z} = \{\dots, -2, -1, 0, 1, 2, \dots\}$, the set of positive integers $\mathbb{N} = \{1, 2, 3, \dots\}$, the set of nonnegative integers $\mathbb{N}_0 = \{0, 1, 2, 3, \dots\}$, the set of negative integers $-\mathbb{N} = \{-1, -2, -3, \dots\}$, the set of real numbers $\mathbb{R}$, the set of complex numbers $\mathbb{C}$, the complex numbers without zero $\mathbb{C}^* = \mathbb{C}\setminus\{0\}$, and the integer interval $[a\,..\,b] = \{i \in \mathbb{Z} \mid a \leq i \leq b\}$.

[2] available from `http://pauillac.inria.fr/algo/libraries/`

Sets. Let S_1 and S_2 be subsets of some set M, and let $l \in M$, and assume we have an addition $+$ on M. Then we can define an addition for sets:

$$\begin{aligned} S_1 + S_2 &:= \{s_1 + s_2 \mid s_1 \in S_1 \text{ and } s_2 \in S_2\}, \\ S_1 + l &:= \{s_1 + l \mid s_1 \in S\}. \end{aligned}$$

Note that $S + \emptyset = \emptyset$ and $l + \emptyset = \emptyset$. The size of a set S, i.e., the number of elements of S, is denoted by $|S|$ (but note that in Section 2.6 $|S|$ has a slightly different meaning: the number of integer lattice points of a bounded subset of $\mathbb{R}^2$).

Vectors. Throughout this work we will use vector notation. A vector, or more exactly a r-vector for some $r \in \mathbb{N}$, is a tuple $(j_1, \ldots, j_r)$ (e.g., of numbers or variables) and is always printed in bold font, e.g., $\mathbf{j}$. The component at position i of $\mathbf{j}$ is denoted by j_i. If a vector has already subscripts, e.g., $\mathbf{j}_p$, the component at position i is denoted by $j_{p,i}$. Let $\mathbf{i} = (i_1, \ldots, i_r)$ and $\mathbf{j} = (j_1, \ldots, j_r)$, we define

$$\begin{aligned} \mathbf{i} \le \mathbf{j} &:\iff \quad i_s \le j_s \quad \text{for all } s \in [1 \,..\, r], \\ \mathbf{i} \cdot \mathbf{j} &:= i_1 j_1 + \cdots + i_r j_r, \\ \mathbf{i}^{\mathbf{j}} &:= i_1^{j_1} \cdots i_r^{j_r}, \\ [\mathbf{i} \,..\, \mathbf{j}] &:= \{\mathbf{l} \in \mathbb{Z}^r \mid \mathbf{i} \le \mathbf{l} \le \mathbf{j}\}. \end{aligned}$$

For the zero-vector we write $\mathbf{0} = (0, \ldots, 0)$. Often we shall use vector notation in a somehow sloppy way, e.g., if $i \in \mathbb{Z}$ and $\mathbf{j} \in \mathbb{Z}^r$ then $(i, \mathbf{j})$ denotes an element of $\mathbb{Z}^{r+1}$, and if n and $\mathbf{k} = (k_1, \ldots, k_r)$ are variables then $F(n, \mathbf{k})$ is the same as $F(n, k_1, \ldots, k_r)$.

Multiple Sums. We also use vector notation for multiple sums. Let $S \subseteq \mathbb{Z}^r$, and let $f(\mathbf{i})$ be a function defined for all $\mathbf{i} \in S$. We define the multiple sum of f over S recursively as

$$\sum_{\mathbf{i} \in S} f(\mathbf{i}) := \sum_{i_1 \in S_1} \sum_{\bar{\mathbf{i}} \in S(i_1)} f(i_1, \bar{\mathbf{i}}),$$

where $S_1 = \{i_1 \in \mathbb{Z} \mid \exists \bar{\mathbf{i}} \in \mathbb{Z}^{r-1} \text{ such that } (i_1, \bar{\mathbf{i}}) \in S\}$ and $S(i_1) = \{\bar{\mathbf{i}} \in \mathbb{Z}^{r-1} \mid (i_1, \bar{\mathbf{i}}) \in S\}$. Occasionally we use the notation

$$\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} f(\mathbf{j}) := \sum_{\mathbf{j} \in [\mathbf{0} \,..\, \mathbf{J}]} f(\mathbf{j}),$$

where $\mathbf{0}, \mathbf{J} \in \mathbb{Z}^r$.

If the summation range S is infinite, then we have to make sure that the sum exists. The sum can be either an analytic object, i.e., a convergent series of a complex valued function, or a formal power series, i.e., the summand has the form $f(\mathbf{i})\mathbf{x}^{\mathbf{i}}$ where $\mathbf{x}$ is a vector of indeterminates and f is a function with values from a ring.

Polynomials. Let $\mathbf{R}$ be a ring. The ring of polynomials over $\mathbf{R}$ in the variable x is denoted by $\mathbf{R}[x]$. The polynomials in several indeterminates $x_1, \ldots, x_r$ are denoted by $\mathbf{R}[x_1, \ldots, x_r]$ (or by $\mathbf{R}[\mathbf{x}]$ or $\mathbf{R}[V]$ where $\mathbf{x} = (x_1, \ldots, x_r)$ is a vector of variables and $V = \{x_1, \ldots, x_r\}$ is a set of variables). The quotient field of $\mathbf{R}[x_1, \ldots, x_r]$, the field of rational functions, is denoted by $\mathbf{R}(x_1, \ldots, x_r)$. The total degree of a polynomial p in the variables $\mathbf{k}$ is denoted by $\deg_{\mathbf{k}}(p)$.

Let $p \in \mathbf{R}[x]$ be a polynomial in x and let $a \in \mathbf{R}$. Then the value of p at a, (the evaluation of p at a) is denoted by $p(a)$. The polynomial p is often also written as $p(x)$, so our notation does not distinguish between a polynomial and the evaluation of a polynomial, but this will not cause any misunderstanding. The evaluation of polynomials in several indeterminates and of rational functions is denoted analogously.

The Gamma Function. The Gamma function $\Gamma(z)$ is defined, following Weierstraß, by the equation

$$\frac{1}{\Gamma(z)} := ze^{\gamma z} \prod_{n=1}^{\infty} \Big((1 + \frac{z}{n}) e^{-\frac{z}{n}} \Big)$$

where $\gamma = \lim_{m\to\infty} (\sum_{k=1}^{m} \frac{1}{k} - \ln m)$ is Euler's constant. The Gamma function is analytic in $\mathbb{C}$ except at the points $z = 0, -1, -2, \ldots$, where it has poles of first order. We will frequently use the following well-know properties of $\Gamma(z)$:

$$\Gamma(n) = (n-1)! \qquad \text{if } n \in \mathbb{N}, \tag{1.5}$$
$$\Gamma(z+1) = z\Gamma(z). \tag{1.6}$$

Falling and Rising Factorials. The Pochhammer symbol $(a)_n$ for $n \in \mathbb{Z}$ and for $a \in \mathbf{K}$, for some field $\mathbf{K}$ of characteristic 0, is defined as

$$(a)_n := \begin{cases} a(a+1)\cdots(a+n-1) & \text{if } n \in \mathbb{N} \\ 1 & \text{if } n = 0 \\ \dfrac{1}{(a-1)(a-2)\cdots(a+n)} & \text{if } n \in -\mathbb{N} \text{ and } a \notin \{1, 2, \ldots, -n\}. \end{cases}$$

The Pochhammer symbol is also known as "rising factorial" and denoted by $a^{\overline{n}}$. We define the falling factorial

$$a^{\underline{n}} := (-1)^n (-a)^{\overline{n}}$$

and see that $a^{\underline{n}} = a(a-1)\ldots(a-n+1)$ if $n \in \mathbb{N}$. The most important and frequently used identities for Pochhammer symbols are

$$(a)_n = \frac{\Gamma(a+n)}{\Gamma(a)} \qquad \text{if } a \in \mathbb{C} \text{ and } a+n \neq 0, -1, -2, \ldots \tag{1.7}$$
$$\frac{(a)_n}{(a)_m} = (a+m)_{n-m}. \tag{1.8}$$

Binomials. The binomial coefficient for $n, k \in \mathbb{C}$ is defined as

$$\binom{n}{k} := \lim_{\epsilon\to 0} \frac{\Gamma(n+1+\epsilon)}{\Gamma(k+1)\Gamma(n-k+1+\epsilon)}$$

which exists, provided that $k \in \mathbb{Z}$ if $n \in -\mathbb{N}$. If $k \in \mathbb{Z}$ this is equivalent to

$$\binom{n}{k} = \begin{cases} \frac{n^{\underline{k}}}{k!} & \text{if } k \geq 0 \\ 0 & \text{if } k < 0. \end{cases}$$

The binomial coefficient satisfies the following basic recurrence relations:

$$\begin{aligned} (n-k)\binom{n}{k} &= n\binom{n-1}{k}, \\ k\binom{n}{k} &= (n-k+1)\binom{n}{k-1}, \\ \binom{n}{k} &= \binom{n-1}{k} + \binom{n-1}{k-1}, \end{aligned}$$

for all n and k in $\mathbb{C}$ such that the binomial coefficients are defined. The first two relations imply that the binomial coefficient is a hypergeometric function, the third one is our first example of a so-called k-free recurrence relation.

Recurrences and Operators. The most important tool for automatic proofs of summation identities are recurrences. We now formally define recurrence relations for functions and a convenient operator notation for recurrences.

Let $F(\mathbf{k}, \mathbf{x}) : D \subseteq \mathbb{C}^{r+s} \to \mathbf{R}$ be a function of the variables $\mathbf{k} = (k_1, \ldots, k_r)$ and $\mathbf{x} = (x_1, \ldots, x_s)$, where $\mathbf{R}$ is some ring. Then F is said to satisfy a *homogeneous linear polynomial recurrence relation in* $\mathbf{k}$ iff there exist a finite nonempty set $S \in \mathbb{Z}^r$ and for every $\mathbf{i} \in S$ a polynomial $a_{\mathbf{i}}(\mathbf{k}, \mathbf{x}) \in \mathbf{R}[\mathbf{k}, \mathbf{x}]$, not all trivial, such that

$$\sum_{\mathbf{i} \in S} a_{\mathbf{i}}(\mathbf{k}, \mathbf{x}) F(\mathbf{k} - \mathbf{i}, \mathbf{x}) = 0$$

for all $(\mathbf{k}, \mathbf{x}) \in D$ with $(\mathbf{k} - \mathbf{i}, \mathbf{x}) \in D$ for all $\mathbf{i} \in S$. The order of the recurrence in the variable k_l is defined as $\max\{i_{1,l} - i_{2,l} \mid \mathbf{i}_1, \mathbf{i}_2 \in S \text{ and } a_{\mathbf{i}_1}, a_{\mathbf{i}_2} \neq 0\}$.

By using operator notation we can handle recurrences in a very elegant way. Let $F(n, k)$ be a function of the variables n and k (we can only use operators if we associate variables to the function). For the variables n, k (which are usually denoted by lower case letters) we define the *forward-shift operators* N, K (which are denoted by the corresponding upper case letter) that operate on F as follows

$$N\, F(n, k) := F(n+1, k) \quad \text{and} \quad K\, F(n, k) := F(n, k+1).$$

Since such operators can be added, multiplied, and even inverted ($N^{-1}\, F(n, k) = F(n-1, k)$) we can write a recurrence relation

$$\sum_{(i,j) \in S} a_{i,j}(n, k) F(n-i, k-j) = 0$$

as the application of the operator $P(n,k,N,K) = \sum_{(i,j)\in S} a_{i,j}(n,k) N^{-i} K^{-j}$ to F:

$$P(n,k,N,K)\,F(n,k) \;=\; 0.$$

It is convenient (and no serious restriction) to consider only those operators where the shift operators have nonnegative exponents, so that such operators are polynomials. We define the *ring of polynomial recurrence operators in n and k* as

$$\mathbf{R}[n,k]\langle N,K\rangle \;:=\; \{\sum_{(i,j)\in S} a_{i,j}(n,k)\,N^i K^j \mid a_{i,j}(n,k)\in \mathbf{R}[n,k] \text{ and } S\subset \mathbb{N}_0^2, |S|<\infty\}.$$

This set is in a natural way equipped with an addition and a noncommutative multiplication, so that $\mathbf{R}[n,k]\langle N,K\rangle$ is a noncommutative ring. The essential noncommutativity relations are

$$n\,N \;=\; N\,(n-1) \quad \text{and} \quad k\,K \;=\; K\,(k-1).$$

Of course we can use any variable and shift operator we want, e.g., if we have a function $F(n,\mathbf{k})$ of the variables $\mathbf{k} = (k_1,\ldots,k_r)$, we denote the ring of recurrence operators by $\mathbf{R}[n,k_1,\ldots.k_r]\langle N,K_1,\ldots,K_r\rangle$ or in vector notation by $\mathbf{R}[n,\mathbf{k}]\langle N,\mathbf{K}\rangle$.

Very often we will use special forms of recurrence operators in $\mathbf{R}[n,\mathbf{k}]\langle N,\mathbf{K}\rangle$. Among the most important are the k-free recurrences (see Chapter 2), whose polynomial coefficients are free of $\mathbf{k}$: we will denote the set of those recurrences by $\mathbf{R}[n]\langle N,\mathbf{K}\rangle$. The other important type of recurrence is the so-called "certificate recurrence" (see Chapter 3), a recurrence involving the *forward-shift difference* or *delta* operators defined as follows

$$\Delta_n \;:=\; (N-1) \quad \text{and} \quad \Delta_{k_l} \;:=\; (K_l-1).$$

Every recurrence involving such delta operators can be written in *normal form*, i.e., as a sum of power products of shift operators with a left polynomial coefficient. If we speak of orders of a certificate recurrence, we mean the orders of the normal form.

Chapter 2

Sister Celine's Technique: k-free Recurrences

"Sister Celine's technique deserves to become more widely used."
Earl D. Rainville ([Rai60])

2.1 Introduction

Sister Celine's technique, or the method of k-free recurrences, provides an elementary method to compute recurrences for multiple sums $\sum_{\mathbf{k}} F(n, \mathbf{k})$. The central concept is that of a k-free recurrence for the hypergeometric summand $F(n, \mathbf{k})$, that is a homogeneous polynomial recurrence relation whose coefficients do not depend on the summation variables $\mathbf{k}$, i.e., a recurrence of the form

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}}(n) F(n-i, \mathbf{k}-\mathbf{j}) \;=\; 0, \tag{2.1}$$

where the $a_{i,\mathbf{j}}(n)$ are polynomials in n and S is a finite set of integer tuples.

A simple example will illustrate how we can find a recurrence for the sum using a k-free recurrence for the summand. Let us prove the trinomial identity

$$(x+y+z)^n \;=\; \sum_{i,j} \binom{n}{j} \binom{j}{i} x^i y^{j-i} z^{n-j}.$$

A recurrence, which is free of i and j, for the summand $F(n, i, j)$ is:

$$xF(n-1, i-1, j-1) + yF(n-1, i, j-1) + zF(n-1, i, j) - F(n, i, j) = 0. \tag{2.2}$$

We (or our computer) can independently check that F satisfies this recurrence: divide the recurrence by $F(n, i, j)$, the fractions like $F(n-1, i-1, j-1)/F(n, i, j)$ are rational functions, and the equation for rational functions can be checked to be zero. We sum this recurrence over all integers i and j and, since the polynomial coefficients do not depend on the summation variables, and since $\sum_{i,j} F(n-1, i-1, j-1) = \sum_{i,j} F(n-1, i, j-1) = \sum_{i,j} F(n-1, i, j)$, we get a recurrence relation for the sum $f(n)$:

$$(x + y + z)f(n-1) - f(n) = 0.$$

It is immediately clear that $(x+y+z)^n$ satisfies this recurrence and, after checking the initial value, the identity is proved. Moreover, the recurrence is of first order, thus allowing us to evaluate the sum without any a priori knowledge of the closed form.

The important observation of Sister Celine, who developed her method to obtain pure recurrence relations for hypergeometric polynomials, was that the problem of finding a recurrence relation for $F(n, \mathbf{k})$ can be reduced to the problem of finding a recurrence relation for the rational functions $F(n-i, \mathbf{k}-\mathbf{j})/F(n, \mathbf{k})$. The latter one is a finite problem and can be solved algorithmically.

Sister Celine's technique

- For a hypergeometric function $F(n, \mathbf{k})$ with values in a field $\mathbf{K}$ and a set S of integer tuples make an Ansatz of the form

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}}(n) F(n-i, \mathbf{k}-\mathbf{j}) \;=\; 0. \tag{2.3}$$

- Divide (2.3) by $F(n, \mathbf{k})$ to get the rational equation

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}}(n) R_{i,\mathbf{j}}(n, \mathbf{k}) \;=\; 0 \tag{2.4}$$

  and clear denominators to get the polynomial equation

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}}(n) p_{i,\mathbf{j}}(n, \mathbf{k}) = 0. \tag{2.5}$$

- Compare the coefficient of every power product $k_1^{l_1} \cdots k_r^{l_r}$ in this equation with zero to get a homogeneous linear equation system for the $a_{i,\mathbf{j}}$ over the field of rational functions $\mathbf{K}(n)$.

- Compute a basis for the vectorspace of solutions of this equation system (i.e., the nullspace of the matrix). Every nontrivial element of this nullspace (after multiplying with a common denominator) yields a k-free recurrence.[1]

[1] Note that two linearly independent solutions of the equation system do not necessarily yield two independent recurrences: they possibly differ only by a shift.

Note that in the next sections we will only consider functions $F(n, \mathbf{k}, \boldsymbol{\alpha}) : D \subseteq \mathbb{C}^m \to \mathbb{C}$ that are hypergeometric in n and $\mathbf{k}$ and possibly contain additional parameters $\boldsymbol{\alpha}$. So the polynomials $a_{i,\mathbf{j}}$ are polynomials in n and $\boldsymbol{\alpha}$ over the complex numbers, and we have to solve an equation system over the field $\mathbb{C}(n, \boldsymbol{\alpha})$.

The remaining task is to find a set of tuples S (a so-called structureset), such that a k-free recurrence exists. In the next sections we will show that for a large class of functions, the proper hypergeometric functions, such a set exists. Furthermore we will show that a special class of structuresets, the P-maximal structuresets, are the most suitable structuresets for this algorithm.

The simple procedure outlined here in this rough form is not a usable algorithmic tool. The problem of finding k-free recurrences is a time and space consuming problem. Some efficient generalizations of Sister Celine's technique are described in Section 3.5. All algorithms in this diploma thesis have been implemented in Mathematica; a description of these implementations is contained in Chapter 4.

Sister Mary Celine Fasenmyer — a short biography can be found in [PWZ96] — is the founder of the subject of finding recurrences for hypergeometric sums ([Fas47], [Fas49]). A whole chapter of Rainville's book on special functions ([Rai60]) is devoted to her method. In the seventies Pierre Verbaeten developed an existence theory for Sister Celine's method, and wrote the first computer programs. His work was done before computer algebra systems were developed to serve as a framework for such programs, and before the connection between hypergeometric functions and binomial summation was well recognized. Unfortunately Verbaeten's results are not well known to the scientific community, due to the fact that his main work [Ver76] is written in Dutch. The only English text [Ver74] is a summary of three pages. A German version can be found in the last chapter of J. Hornegger's diploma thesis [Hor92]. Finally, after the exciting discoveries in the area of symbolic summation, Herbert S. Wilf and Doron Zeilberger used Sister Celine's method to provide elementary existence proofs for Zeilberger's fast single sum algorithm (see [PWZ96]), and generalized her method to the case of several summation variables ([WZ92a]).

This generalization (including the existence theorem) as well as the work of Verbaeten (including the central notion of a P-maximal structureset) are the content of the next sections.

2.2 Proper Hypergeometric Functions

In this section, by following [WZ92a], we will define the set of proper hypergeometric functions, for which we can show that a k-free recurrence relation holds. We define these functions as evaluations of certain hypergeometric terms. Hypergeometric terms are the natural objects to work with: they are easy to define and manipulate and can serve as input to computer programs. But only functions can be summed, so in the following we will clearly distinguish between functions and terms.

The simplest hypergeometric terms are polynomials and factorials expressions like $(n+k)!$ and exponential terms like x^k with x free of k. Note that it is the property of integer linearity in k

and n that makes the factorial hypergeometric. Rational functions, too, are obviously hypergeometric, but they do not necessarily satisfy a k-free recurrence, as Wilf and Zeilberger [WZ92a] pointed out: Even the simple $1/(n^2+k^2)$ does not. Suppose it does satisfy a recurrence of the form (2.1), then in an appropriate neighbourhood of every pole (i, j) all, except one, terms are finite. This one term tends to infinity, because the polynomial coefficient does not depend on k. Thus a recurrence cannot hold in a neighbourhood of this pole.

So it is natural to define the terms as products of polynomials, integer-linear factorials (more general Gamma functions) and exponentials. Additionally we assume that the function does not only depend on the hypergeometric variables n and $\mathbf{k} = (k_1, \ldots, k_r)$, but also on some additional parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_l)$.

Definition 2.1. Let $r \in \mathbb{N}, l \in \mathbb{N}_0$, and let $V = \{n, k_1, \ldots, k_r, \alpha_1, \ldots, \alpha_l\}$ be a set of variables. Let
1. $pp \in \mathbb{N}_0$ and $qq \in \mathbb{N}_0$, and
2. for every $p \in [1\,..\,pp]$ let $a_p \in \mathbb{Z}$, $\mathbf{b}_p \in \mathbb{Z}^r$, and $c_p \in \mathbb{C}[\boldsymbol{\alpha}]$, and
3. for every $q \in [1\,..\,qq]$ let $u_q \in \mathbb{Z}$, $\mathbf{v}_q \in \mathbb{Z}^r$, and $w_q \in \mathbb{C}[\boldsymbol{\alpha}]$, and
4. let $P(n, \mathbf{k}) \in \mathbb{C}[V]$ be a polynomial, and
5. let $x_0, x_1, \ldots, x_r \in \mathbb{C}[\boldsymbol{\alpha}]$ be polynomials.

Then

$$t \;=\; P(n,\mathbf{k}) \frac{\prod_{p=1}^{pp} \Gamma(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)}{\prod_{q=1}^{qq} \Gamma(u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q)} x_0^n x_1^{k_1} \cdots x_r^{k_r}. \tag{2.6}$$

is a *proper hypergeometric term* with hypergeometric variables n and $\mathbf{k}$ and additional parameters $\boldsymbol{\alpha}$. The linear forms $a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p$ are called the *numerator factorial expressions* of t, the linear forms $u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q$ are called the *denominator factorial expressions* of t.

It is possible to give a more general definition of proper hypergeometric terms such that the existence theorem (Theorem 2.19) still holds. The c_p, the w_q, and the x_i may be elements of $\mathbb{C}(\boldsymbol{\alpha})$, and $P(n, \mathbf{k})$ may be a rational function with a denominator that factors completely into integer-linear factors $un + \mathbf{v} \cdot \mathbf{k} + w$. We decided to give a restricted definition to simplify the investigations in the following sections. But note that the restriction is not a serious one. Such more general hypergeometric terms can be transformed into proper hypergeometric terms by introducing new additional unknowns $\bar{\alpha}_i$ to replace the rational functions c_p, w_q, and x_i, and by writing each denominator factor of $P(n, \mathbf{k})$ as the quotient of two Gamma functions: $1/(un + \mathbf{v} \cdot \mathbf{k} + w) = \Gamma(un + \mathbf{v} \cdot \mathbf{k} + w)/\Gamma(un + \mathbf{v} \cdot \mathbf{k} + w + 1)$.

Many important special functions, e.g., the well known Bessel function

$$J_n(z) \;=\; \sum_{r=0}^{\infty} \frac{(-1)^r z^{n+2r}}{2^{n+2r}\, r!\, \Gamma(n+r+1)},$$

are the sum (finite or infinite) of a proper hypergeometric term. The Pochhammer symbol $(a)_n$ is $\Gamma(a+n)/\Gamma(a)$, provided that $a+n$ is neither 0 nor a negative integer. So the summand of the generalized hypergeometric function

$$ {}_pF_q(x) \;=\; \sum_{k=0}^{\infty} \frac{(a_1 n + b_1)_k \cdots (a_p n + b_p)_k}{(u_1 n + v_1)_k \cdots (u_q n + v_q)_k} \frac{x^k}{k!}, $$

if the a_i and u_i are integers, is a proper hypergeometric term.

What can we say about binomial coefficient identities, which we are mainly interested in. Binomial coefficients do not appear in the above definition, but since $\binom{n}{k}$, for $n \notin -\mathbb{N}$, is defined as $\frac{n!}{k!(n-k)!}$, we can write binomial coefficients that are integer-linear in the hypergeometric variables as proper hypergeometric terms. But note that the function $\binom{n}{k}$ is defined for $n \in -\mathbb{N}$ and $k \in \mathbb{Z}$ whereas the function $\frac{n!}{k!(n-k)!}$ is not.

Most hypergeometric terms that we will meet are of a special form, i.e., they are irreducible. They have the nice property that certain cancellations in the rational functions $F(n-i, \mathbf{k}-\mathbf{j})/F(n, \mathbf{k})$ can not happen, so that we are able to give the equation (2.5) explicitly.

Definition 2.2. The proper hypergeometric term t is called *irreducible* iff t does not have a polynomial part, and there do not exist a numerator factorial expression σ and a denominator factorial expression μ of t, such that $\sigma - \mu \in \mathbb{Z}$. If t is not irreducible it is called *reducible*.

Although discrete functions are our main interest, we evaluate the term t also for complex values of n and $\mathbf{k}$. This is not only a natural generalization, it has also the advantage that we can take limits of such functions. This enables us to show that a recurrence holds for certain extensions of proper hypergeometric functions, i.e., for values of n and $\mathbf{k}$ for which the proper hypergeometric function is not well-defined. There is a slight difficulty in this general definition: we have to define the function x^k for complex values of x and k. It is defined as $e^{k\log(x)}$, involving a complex logarithm. Since the logarithm function is a multivalued function we have to agree on a branch of it. It is not so important which branch of the logarithm we actually choose – if k is an integer then $e^{k\log(x)}$ has the same value for every logarithm of x.

We define the complex logarithm function $\log(z) : \mathbb{C}^* \to \mathbb{C}$ as

$$ \log(z) \;=\; \ln(|z|) \;+\; i\arg(z), \tag{2.7} $$

where $\ln(x)$ is the real natural logarithm and $\arg(z)$ is the argument of z with $-\pi < \arg(z) \leq \pi$. This logarithm function is defined on the whole complex plane except zero, and it is not continuous at the negative real numbers. This implies that the proper hypergeometric function, as defined below, is not continuous in the additional parameters $\boldsymbol{\alpha}$ (at least not for all values of $\boldsymbol{\alpha}$). But it is certainly continuous in the hypergeometric variables.

Definition 2.3. Let t be a proper hypergeometric term as in Definition 2.1. Let $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \in \mathbb{C}^{1+r+l}$, and let $\tilde{c}_p$, $\tilde{w}_q$, $\tilde{x}_i$ be the polynomials c_p, w_q, respectively x_i evaluated at $\tilde{\boldsymbol{\alpha}}$. The term t is said to be *well-defined at* $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}})$, iff for every p the number $(a_p\tilde{n} + \mathbf{b}_p \cdot \tilde{\mathbf{k}} + \tilde{c}_p)$ is neither 0 nor a negative integer and iff $\tilde{x}_i$ is nonzero for all $i \in [0\,..\,r]$. The set

$$ D_t \;=\; \{(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \in \mathbb{C}^{1+r+l} \mid t \text{ is well-defined at } (\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}})\} $$

is called the *set of well-defined values of* t. The function $F_t : D_t \to \mathbb{C}$ is defined as the evaluation of t at the well-defined points

$$F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) = P(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \frac{\prod_{p=1}^{pp} \Gamma(a_p \tilde{n} + \mathbf{b}_p \cdot \tilde{\mathbf{k}} + \tilde{c}_p)}{\prod_{q=1}^{qq} \Gamma(u_q \tilde{n} + \mathbf{v}_q \cdot \tilde{\mathbf{k}} + \tilde{w}_q)} e^{\tilde{n} \log(\tilde{x}_0) + \cdots + \tilde{k}_r \log(\tilde{x}_r)},$$

where e^x is the complex exponential function and $\log(x)$ is the complex logarithm defined in (2.7). We call F_t the *proper hypergeometric function* in n and $\mathbf{k}$ of t.

Of course we have to exclude certain values from the evaluation domain: the singularities of the numerator Gamma functions and the zeros of the x_i. But note that we need not exclude points where the argument of a Gamma function in the denominator of the term is a negative integer, because $\frac{1}{\Gamma(x)}$ is an entire function; F_t simply becomes zero at such points.

The hypergeometric function $\binom{n}{k}$ is represented by the proper hypergeometric term $\frac{n!}{k!(n-k)!}$. So $\binom{n}{k}$ is not a proper hypergeometric function for all values of n. We have to exclude the negative integers. This restriction is a big disadvantage, since the existence theorem of the following sections only guarantees a recurrence to hold at well-defined values. Often, however, we want or even need the recurrence to hold at the singular values, and we have to show this by other means (see Section 2.7). We have the same situation for Pochhammer symbols and falling factorials.

Using the functional equation of the Gamma function $\Gamma(a + x) = (x)_a \Gamma(x)$ for an arbitrary integer a, we can give a explicit expression for the fundamental fraction $F(n - i, \mathbf{k} - \mathbf{j})/F(n, \mathbf{k})$.

Definition 2.4. Let t be a proper hypergeometric term in n and $\mathbf{k}$ as in Definition 2.1, and let $(i, \mathbf{j}) \in \mathbb{Z}^{r+1}$. We define

$$R_{t,i,\mathbf{j}} = \frac{P(n - i, \mathbf{k} - \mathbf{j})}{P(n, \mathbf{k})} \frac{\prod_{p=1}^{pp} (a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-i a_p - \mathbf{j} \cdot \mathbf{b}_p}}{\prod_{q=1}^{qq} (u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q)_{-i u_q - \mathbf{j} \cdot \mathbf{v}_q}} x_0^{-i} x_1^{-j_1} \cdots x_r^{-j_r}, \qquad (2.8)$$

as a rational function in the variables $n, \mathbf{k}, \boldsymbol{\alpha}$.

Note that a Pochhammer expression in the numerator actually is part of the denominator iff the integer $-i a_p - \mathbf{j} \cdot \mathbf{b}_p$ is negative. Similarly, a denominator Pochhammer expression can be actually a part of the numerator.

Lemma 2.5. *Let t be a proper hypergeometric term in n and $\mathbf{k}$ and additional parameters $\boldsymbol{\alpha}$. For the proper hypergeometric function F_t of t we have for every point $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \in D_t$*

$$\frac{F_t(\tilde{n} - i, \tilde{\mathbf{k}} - \mathbf{j}, \tilde{\boldsymbol{\alpha}})}{F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}})} = R_{t,i,\mathbf{j}}(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}), \qquad (2.9)$$

if $F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \neq 0$ and $(\tilde{n} - i, \tilde{\mathbf{k}} - \mathbf{j}, \tilde{\boldsymbol{\alpha}}) \in D_t$. Furthermore we can write this equation as a polynomial equation, so that the restriction $F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \neq 0$ becomes superfluous.

Proof. First note that the denominator of the rational function (2.8) is automatically nonzero if $F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \neq 0$. Using the fundamental functional equation (1.6) of the Gamma function one sees that for every p and every q we have

$$\begin{aligned} \frac{\Gamma(a_p(\tilde{n}-i) + \mathbf{b}_p \cdot (\tilde{\mathbf{k}} - \mathbf{j}) + c_p(\tilde{\boldsymbol{\alpha}}))}{\Gamma(a_p \tilde{n} + \mathbf{b}_p \cdot \tilde{\mathbf{k}} + c_p(\tilde{\boldsymbol{\alpha}}))} &= (a_p \tilde{n} + \mathbf{b}_p \cdot \tilde{\mathbf{k}} + c_p(\tilde{\boldsymbol{\alpha}}))_{-ia_p - \mathbf{j}\cdot\mathbf{b}_p}, \\ \frac{\Gamma(u_q \tilde{n} + \mathbf{v_q} \cdot \tilde{\mathbf{k}} + w_q(\tilde{\boldsymbol{\alpha}}))}{\Gamma(u_q(\tilde{n}-i) + \mathbf{v_q} \cdot (\tilde{\mathbf{k}} - \mathbf{j}) + w_q(\tilde{\boldsymbol{\alpha}}))} &= \frac{1}{(u_q \tilde{n} + \mathbf{v}_q \cdot \tilde{\mathbf{k}} + w_p(\tilde{\boldsymbol{\alpha}}))_{-iu_q - \mathbf{j}\cdot\mathbf{v}_q}}, \end{aligned}$$

which proves the claim. □

As a simple consequence we note that every proper hypergeometric function is hypergeometric. Next we define the fundamental concepts structureset and k-free recurrence.

Definition 2.6. Let t be a proper hypergeometric term with m hypergeometric variables. A *structureset* for t is a finite, nonempty set $S \subseteq \mathbb{Z}^m$.

Definition 2.7. Let t be a proper hypergeometric term in n and $\mathbf{k}$ with additional parameters $\boldsymbol{\alpha}$. F_t satisfies a *recurrence free of* $\mathbf{k}$ iff there exist a structureset S for t and polynomials $a_{i,\mathbf{j}}(n, \boldsymbol{\alpha}) \in \mathbb{C}[n, \boldsymbol{\alpha}]$, not all zero, such that for every $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \in D_t$ with $(\tilde{n} - i, \tilde{\mathbf{k}} - \mathbf{j}, \tilde{\boldsymbol{\alpha}}) \in D_t$ for all $(i, \mathbf{j}) \in S$

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}}(\tilde{n}, \tilde{\boldsymbol{\alpha}}) F_t(\tilde{n} - i, \tilde{\mathbf{k}} - \mathbf{j}, \tilde{\boldsymbol{\alpha}}) = 0$$

holds. The structureset S is called a *structure* for F_t, iff F_t satisfies a recurrence free of $\mathbf{k}$ on S.

Although we defined k-free recurrences only for proper hypergeometric function, we will also use this concept for arbitrary functions. For the sake of readability, we will often drop the parameters $\boldsymbol{\alpha}$ from $a_{i,\mathbf{j}}$ and F_t.

Sister Celine's technique computes polynomials $a_{i,\mathbf{j}}$ such that $\sum a_{i,\mathbf{j}} R_{t,i,\mathbf{j}}$, which is an element of a field of rational functions, is identically zero. But in order to prove identities we need a k-free recurrence $\sum a_{i,\mathbf{j}} F_t(n-i, \mathbf{k}-\mathbf{j}) = 0$; so we have to investigate the relationship between the former and the latter equation. Unfortunately, in previous publications on the subject (e.g., in [WZ92a]) no clear distinction has been made between the these two equations. Whereas it is easy to show that a k-free recurrence for F_t yields one for the rational functions (since the rational function identity vanishes for enough values of n and $\mathbf{k}$), the converse is not so simple. Of course we can simply plug in values for n and $\mathbf{k}$ into the rational functions, but only if none of the denominators evaluates to zero. Fortunately the fact that $F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \neq 0$ implies that $R_{t,i,\mathbf{j}}$ is well-defined at $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}})$ settles this problem for proper hypergeometric functions. The following theorem shows this and thus guarantees us that we only have to find a recurrence satisfied by the $R_{t,i,\mathbf{j}}$ to get a recurrence for F_t. However, if we consider extensions of proper hypergeometric functions (e.g., binomials) this proof fails, and we have to use other methods (see Section 2.7).

Theorem 2.8. *If, for a proper hypergeometric term t in the variables n and $\mathbf{k}$ with additional parameters $\boldsymbol{\alpha}$, there exists a structureset S and polynomials $a_{i,\mathbf{j}}(n, \boldsymbol{\alpha}) \in \mathbb{C}[n, \boldsymbol{\alpha}]$ such that*

$$\sum_{(i,\mathbf{j}) \in S} a_{i,j}(n, \boldsymbol{\alpha}) R_{t,i,\mathbf{j}} = 0 \tag{2.10}$$

then also

$$\sum_{(i,\mathbf{j}) \in S} a_{i,j}(\tilde{n}, \tilde{\boldsymbol{\alpha}}) F_t(\tilde{n} - i, \tilde{\mathbf{k}} - \mathbf{j}, \tilde{\boldsymbol{\alpha}}) = 0 \tag{2.11}$$

for every $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \in D_t$ with $(\tilde{n} - i, \tilde{\mathbf{k}} - \mathbf{j}, \tilde{\boldsymbol{\alpha}}) \in D_t$ for all $(i, \mathbf{j}) \in S$.

Proof. Let $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \in D_t$ be a point for which we want to show the recurrence to hold. We distinguish three cases.

1. If $F_t(\tilde{n} - i, \tilde{\mathbf{k}} - \mathbf{j}, \tilde{\boldsymbol{\alpha}})$ is zero for every $(i, \mathbf{j}) \in S$ then (2.11) trivially holds.
2. If $F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) \neq 0$ then (2.11) follows from (2.9) by multiplying (2.10) with $F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}})$.
3. If $F_t(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}}) = 0$ and $F_t(\tilde{n} - \tilde{i}, \tilde{\mathbf{k}} - \tilde{\mathbf{j}}, \tilde{\boldsymbol{\alpha}}) \neq 0$ for some $(\tilde{i}, \tilde{\mathbf{j}}) \in S$ then we multiply the rational equation (2.10) with the rational function $\frac{F_t(n,\mathbf{k},\boldsymbol{\alpha})}{F_t(n-\tilde{i},\mathbf{k}-\tilde{\mathbf{j}},\boldsymbol{\alpha})}$ and we get the rational equation

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}} \frac{F_t(n - i, \mathbf{k} - \mathbf{j}, \boldsymbol{\alpha})}{F_t(n - \tilde{i}, \mathbf{k} - \tilde{\mathbf{j}}, \boldsymbol{\alpha})} \;=\; 0$$

   which, analogous to case 2, can be evaluated at $(\tilde{n}, \tilde{\mathbf{k}}, \tilde{\boldsymbol{\alpha}})$ to yield (2.11).

□

A consequence of Theorem 2.8 is that we cannot only add two recurrences or multiply a recurrence with a polynomial, but we are also allowed to divide a recurrence by a polynomial. More exactly this means that if

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}}(n) F_t(n - i, \mathbf{k} - \mathbf{j}) \;=\; 0$$

and if some $p(n)$ divides all $a_{i,\mathbf{j}}(n)$ then also

$$\sum_{(i,\mathbf{j}) \in S} \frac{a_{i,\mathbf{j}}(n)}{p(n)} F_t(n - i, \mathbf{k} - \mathbf{j}) \;=\; 0,$$

since this holds for the recurrence of rational functions. But note that after manipulations of the k-free recurrence, such as taking limits (see Section 2.7) or transforming into a certificate recurrence (see Section 3.2), such a division is not allowed any more.

2.3 The Associated Polynomial

Given a proper hypergeometric term t, we want to find a structure S for F_t. We have seen that it suffices to solve the rational equation

$$\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n) R_{t,i,\mathbf{j}} \ = 0 \tag{2.12}$$

for polynomials $a_{i,\mathbf{j}}(n)$ not depending on $\mathbf{k}$. To this end, we multiply the equation with the least common multiple of the denominators of the rational functions and cancel the greatest common divisor of the numerators of the $R_{t,i,\mathbf{j}}$, and get a polynomial equation $\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n)p_{t,i,\mathbf{j}} \ = 0$. In this section we will investigate these manipulations for an arbitrary proper hypergeometric term. Following Verbaeten [Ver76], we are able to find a common denominator explicitly, and also some factors that can be cancelled from the numerators. The resulting polynomial equation will be called the associated polynomial.

To make the $R_{t,i,\mathbf{j}}$ polynomial, we have to multiply with parts of the Pochhammer expressions of (2.8). Those points $(i,\mathbf{j}) \in S$ where the integers $-ia_p - \mathbf{j}\cdot\mathbf{b}_p$ (and $-iu_q - \mathbf{j}\cdot\mathbf{v}_q$) are minimal (respectively maximal) play an outstanding role. It is convenient to give them a name.

Definition 2.9. Let t be a proper hypergeometric term as in Definition 2.1 and S a structureset for t. Let $p \in [1\,..\,pp]$ and $q \in [1\,..\,qq]$. A point $(I,\mathbf{J}) \in S$ is called a *numerator boundary point* and denoted by $(I_{num,p}, \mathbf{J}_{num,p})$, iff

$$Ia_p + \mathbf{J}\cdot\mathbf{b}_p \ \geq \ ia_p + \mathbf{j}\cdot\mathbf{b}_p \qquad \text{for all } (i,\mathbf{j}) \in S.$$

Similarly, $(I,\mathbf{J}) \in S$ is called a *denominator boundary point* and denoted by $(I_{den,q}, \mathbf{J}_{den,q})$, iff

$$Iu_q + \mathbf{J}\cdot\mathbf{v}_q \ \leq \ iu_q + \mathbf{j}\cdot\mathbf{v}_q \qquad \text{for all } (i,\mathbf{j}) \in S.$$

The points $(I_{num,p}, \mathbf{J}_{num,p})$ and $(I_{den,q}, \mathbf{J}_{den,q})$ are called boundary points, because as the extremal points of linear functions they are on the "boundary" of S (more exactly, on the boundary of the convex hull of S). It is possible that there are several boundary points for a p (or a q), but this is of no importance.

Definition 2.10. Let t be a proper hypergeometric term as in Definition 2.1, and let S be a structureset for t. The polynomial $P_{t,S}$

$$P_{t,S} \ = \ \sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n)\ P(n-i,\mathbf{k}-\mathbf{j})\ x_0^{\tilde{i}-i} x_1^{\tilde{j}_1-j_1}\cdots x_r^{\tilde{j}_r-j_r}$$
$$\frac{\prod_{p=1}^{pp}(a_p n + \mathbf{b}_p\cdot\mathbf{k} + c_p - I_{num,p}a_p - \mathbf{J}_{num,p}\cdot\mathbf{b}_p)_{(I_{num,p}-i)a_p + (\mathbf{J}_{num,p}-\mathbf{j})\cdot\mathbf{b}_p}}{\prod_{q=1}^{qq}(u_q n + \mathbf{v}_q\cdot\mathbf{k} + w_q - I_{den,q}u_q - \mathbf{J}_{den,q}\cdot\mathbf{v}_q)_{(I_{den,q}-i)u_q + (\mathbf{J}_{den,q}-\mathbf{j})\cdot\mathbf{v}_q}}, \tag{2.13}$$

where $\tilde{i} = \max_{(i,\mathbf{j})\in S} i$ and $\tilde{j}_s = \max_{(i,\mathbf{j})\in S} j_s$, is called the *associated polynomial* of t and S.

It is easy to see that (2.13) is indeed a polynomial in the variables n, $\mathbf{k}$, $\boldsymbol{\alpha}$, and also in our unknowns $a_{i,\mathbf{j}}(n)$ (linear in the latter).

Theorem 2.11. *Let t be a proper hypergeometric term and let S be a structureset for t. The rational equation $\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n)R_{t,i,\mathbf{j}} = 0$ is equivalent to the polynomial equation $P_{t,S} = 0$.*

Proof. In order to transform the rational equation into a polynomial equation we have to multiply with a common denominator of the $R_{t,i,\mathbf{j}}$. Furthermore we cancel common numerator factors. First multiply every $R_{t,i,\mathbf{j}}$ with $P(n,\mathbf{k})$ and $x_0^{\bar{i}} x_1^{\bar{j}_1} \cdots x_r^{\bar{j}_r}$. The remaining rational functions to handle are now the Pochhammer expressions.

What factors of $(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-ia_p - \mathbf{j}\cdot\mathbf{b}_p}$ are in the common denominator of the $R_{t,i,\mathbf{j}}$, or are in the numerator of every $R_{t,i,\mathbf{j}}$? We distinguish two cases according to the sign of the minimal value of the $-ia_p - \mathbf{j}\cdot\mathbf{b}_p$:

- If $-I_{num,p}a_p - \mathbf{J}_{num,p}\cdot\mathbf{b}_p \geq 0$ then every $(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-ia_p - \mathbf{j}\cdot\mathbf{b}_p}$ is in the numerator of $R_{t,i,\mathbf{j}}$. And all these numerator polynomials have the greatest common factor $(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-I_{num,p}a_p - \mathbf{J}_{num,p}\cdot\mathbf{b}_p}$, which we thus can cancel.

- If $-I_{num,p}a_p - \mathbf{J}_{num,p}\cdot\mathbf{b}_p < 0$ then some of the $(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-ia_p - \mathbf{j}\cdot\mathbf{b}_p}$ are part of the denominator of $R_{t,i,\mathbf{j}}$. The least common multiple of these polynomials in the denominator is $1/(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-I_{num,p}a_p - \mathbf{J}_{num,p}\cdot\mathbf{b}_p}$. We multiply each $R_{t,i,\mathbf{j}}$ with this common denominator.

In both cases we multiply $R_{t,i,\mathbf{j}}$ with $1/(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-I_{num,p}a_p - \mathbf{J}_{num,p}\cdot\mathbf{b}_p}$ and the remaining factor of $(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)_{-ia_p - \mathbf{j}\cdot\mathbf{b}_p}$ in $R_{t,i,\mathbf{j}}$ is

$$(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p - I_{num,p}a_p - \mathbf{J}_{num,p}\cdot\mathbf{b}_p)_{(I_{num,p}-i)a_p + (\mathbf{J}_{num,p}-\mathbf{j})\cdot\mathbf{b}_p}.$$

Similarly we can find out with what the Pochhamer expressions $1/(u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q)_{-iu_q - \mathbf{j}\cdot\mathbf{v}_q}$ have to be multiplied (or what has to be canceled) to make them polynomial.

- If $-I_{den,q}u_q - \mathbf{J}_{den,q}\cdot\mathbf{v}_q < 0$ then there are only numerator factors. These numerator factors have the greatest common divisor $1/(u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q)_{-I_{den,q}u_q - \mathbf{J}_{den,q}\cdot\mathbf{v}_q}$, which we thus cancel.

- If $-I_{den,q}u_q - \mathbf{J}_{den,q}\cdot\mathbf{v}_q \geq 0$, the common denominator is equal to $(u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q)_{-I_{den,q}u_q - \mathbf{J}_{den,q}\cdot\mathbf{v}_q}$. We multiply all $R_{t,i,\mathbf{j}}$ with it.

In both cases the remaining expression of $1/(u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q)_{-iu_q - \mathbf{j}\cdot\mathbf{v}_q}$ is equal to the polynomial

$$\frac{1}{(u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q - I_{den,q}u_q - \mathbf{J}_{den,q}\cdot\mathbf{v}_q)_{(I_{den,q}-i)u_q + (\mathbf{J}_{den,q}-\mathbf{j})\cdot\mathbf{v}_q}}.$$

After these manipulations the rational equation is a polynomial equation and equals (2.13). $\square$

We can easily show that we cannot cancel any further factor from the polynomial equation if t is irreducible.

Corollary 2.12. *Let t be an irreducible proper hypergeometric term, S a structureset for t, and $P_{t,S} = \sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n) p_{i,\mathbf{j}}(n,\mathbf{k})$. There does not exist a nonconstant polynomial in n and $\mathbf{k}$ that divides all $p_{i,\mathbf{j}}(n,\mathbf{k})$.*

Proof. Let σ be a numerator factorial expression, and $(I_\sigma, \mathbf{J}_\sigma)$ its boundary point. Every other numerator factorial expression σ_1 with $\sigma - \sigma_1 \in \mathbb{Z}$ has the same boundary point. It follows from (2.13), since there is no denominator factorial expression μ with $\sigma - \mu \in \mathbb{Z}$, that no factor of the form $\sigma + l$ with $l \in \mathbb{Z}$ divides $p_{I_\sigma,\mathbf{J}_\sigma}$. The same holds for every denominator factorial expression. Since t has no polynomial part, every $p_{i,\mathbf{j}}$ consists only of integer shifted factorial expressions, and there cannot be a nonconstant divisor. □

This means that for irreducible terms the equation $P_{t,S} = 0$ cannot be simplified further and is actually the polynomial equation that the algorithm has to solve. Especially the degree formula given below is exact. But note that even for reducible terms it is not very likely that any further cancellation takes place.

We can now determine the degree of the associated polynomial. The total degree of $P_{t,S}$ in $\mathbf{k}$ is the maximal degree of each of the summands. And the degree of a summand is the sum over the subscripts of the Pochhammer expressions in (2.13) (of course we drop the factors that are free of $\mathbf{k}$, i.e., where $\mathbf{b}_p = \mathbf{0}$, $\mathbf{v}_q = \mathbf{0}$), plus the degree of the polynomial part. This means

$$\begin{aligned}
\deg_{\mathbf{k}}(P_{t,S}) &= \deg_{\mathbf{k}}(P(n,\mathbf{k})) \\
&+ \sum_{\substack{p=1\\ \mathbf{b}_p\neq\mathbf{0}}}^{pp} (I_{num,p} a_p + \mathbf{J}_{num,p}\cdot\mathbf{b}_p) - \sum_{\substack{q=1\\ \mathbf{v}_q\neq\mathbf{0}}}^{qq} (I_{den,q} u_q + \mathbf{J}_{den,q}\cdot\mathbf{v}_q) \\
&+ \max_{(i,\mathbf{j})\in S}\Bigg(i\Big(-\sum_{\substack{p=1\\ \mathbf{b}_p\neq\mathbf{0}}}^{pp} a_p + \sum_{\substack{q=1\\ \mathbf{v}_q\neq\mathbf{0}}}^{qq} u_q\Big) + \mathbf{j}\cdot\Big(-\sum_{p=1}^{pp}\mathbf{b}_p + \sum_{q=1}^{qq}\mathbf{v}_q\Big)\Bigg)
\end{aligned} \tag{2.14}$$

Analogous to the definition of the numerator and denominator boundary points, let us define the third type of boundary points.

Definition 2.13. Let t be a proper hypergeometric term as in Definition 2.1, and let S be a structureset for t. A point $(I,\mathbf{J}) \in S$ is called a *difference boundary point* and denoted by $(I_{diff}, \mathbf{J}_{diff})$, iff

$$IA_t + \mathbf{J}\cdot\mathbf{B}_t \le iA_t + \mathbf{j}\cdot\mathbf{B}_t \quad \text{for all } (i,\mathbf{j}) \in S$$

where

$$A_t = \sum_{\substack{p=1\\ \mathbf{b}_p\neq\mathbf{0}}}^{pp} a_p - \sum_{\substack{q=1\\ \mathbf{v}_q\neq\mathbf{0}}}^{qq} u_q \quad\text{and}\quad \mathbf{B}_t = \sum_{p=1}^{pp}\mathbf{b}_p - \sum_{q=1}^{qq}\mathbf{v}_q.$$

$(A_t, \mathbf{B}_t)$ is called the *factorial difference*.

Summarizing we have found

Theorem 2.14. *Let t be a proper hypergeometric term as in Definition 2.1, and let S be a structureset for t. For the total degree of $P_{t,S}$ in the variables $\mathbf{k}$ we have*

$$\begin{aligned}\deg_{\mathbf{k}}(P_{t,S}) &= \deg_{\mathbf{k}}(P(n,\mathbf{k})) \\ &+ \sum_{\substack{p=1\\ \mathbf{b}_p\neq 0}}^{pp} (I_{num,p}a_p + \mathbf{J}_{num,p}\cdot \mathbf{b}_p) - \sum_{\substack{q=1\\ \mathbf{v}_q\neq 0}}^{qq} (I_{den,q}u_q + \mathbf{J}_{den,q}\cdot \mathbf{v}_q) \\ &- I_{\text{diff}}A_t - \mathbf{J}_{\text{diff}}\cdot \mathbf{B}_t.\end{aligned} \tag{2.15}$$

We see that the degree formula only depends on the boundary points of a structureset, and not on any other property of it. This will be of some importance later.

If we compare the coefficients of all power products $k_1^{m_1}\cdots k_r^{m_r}$ in $P_{t,S}$, we get a homogeneous linear equation system for the $a_{i,\mathbf{j}}$. This has a nontrivial solution if the number of unknowns (i.e., the size of the structureset S) exceeds the number of equations. The number of equations equals the number of power products $k_1^{m_1}\cdots k_r^{m_r}$ — an upper bound for this number can be given using the total degree of $P_{t,S}$ in $k_1,\ldots,k_r$. It is well-known that the number of power products of r variables of total degree less or equal d is $\binom{d+r}{r}$. Thus we have

Theorem 2.15. *Let t be a proper hypergeometric term in n and $\mathbf{k}=(k_1,\ldots,k_r)$ and let S be a structureset for t of size $|S|$. If*

$$\binom{deg_{\mathbf{k}}(P_{t,S})+r}{r} < |S|$$

then S is a structure for F_t.

In case of only one summation variable this reduces to

Corollary 2.16. *Let t be a proper hypergeometric term in n and k (i.e., $r=1$) and let S be a structureset for t of size $|S|$. If*

$$deg_k(P_{t,S})+1 < |S|$$

then S is a structure for F_t.

2.4 Existence of Recurrences

After the preliminary work of the last sections, we are ready to prove the existence of k-free recurrences for every proper hypergeometric function. This existence theorem, given by Wilf and Zeilberger in [WZ92a], uses a very simple type of structuresets — the rectangular shaped sets $S_{I,\mathbf{J}}$.

Definition 2.17. Let $I \in \mathbb{N}_0$ and $\mathbf{J} \in \mathbb{N}_0^r$. We define $S_{I,\mathbf{J}}$ to be the structureset $\{(i,\mathbf{j}) \in \mathbb{Z}^{r+1} \mid i \in [0\,..\,I] \text{ and } \mathbf{j} \in [\mathbf{0}\,..\,\mathbf{J}]\}$.

Furthermore let us introduce a useful notation.

Definition 2.18. Let $x \in \mathbb{R}$ and $\mathbf{x} \in \mathbb{R}^m$. Then x^+ is defined as $\max(0,x)$ and $\mathbf{x}^+$ as $(x_1^+, \ldots, x_m^+)$.

Theorem 2.19. *([WZ92a]) Let t be a proper hypergeometric term as in Definition 2.1. The proper hypergeometric function $F_t(n,\mathbf{k})$ satisfies a recurrence in n and $\mathbf{k}$ that is free of $\mathbf{k}$. Moreover there exists a recurrence whose order in n is less or equal*

$$\left\lceil \frac{1}{r!}\left(\mathbf{1} \cdot \Big(\sum_{p=1}^{pp} \mathbf{b}_p^+ + \sum_{q=1}^{qq} (-\mathbf{v}_q)^+ + (-\mathbf{B}_t)^+\Big)\right)^r \right\rceil.$$

Proof. We show that there exist $I \in \mathbb{N}_0$ and $\mathbf{J} \in \mathbb{N}_0^r$ such that $S_{I,\mathbf{J}}$ is a structure for t. The boundary points of this structureset are easily determined. $I_{num,p}$ is equal to I if $a_p \geq 0$ and 0 otherwise. The same for component s of $\mathbf{J}_{num,p}$: $J_{num,p,s}$ is equal to J_s if $b_{p,s} \geq 0$ and 0 otherwise. Similarly the denominator boundary points and the difference boundary point: $I_{den,q}$ ($J_{den,q,s}, I_{diff}, J_{diff,s}$) is equal to I (respectively J_s, I, J_s) if $u_q \leq 0$ (respectively $v_{q,s} \leq 0, A_t \leq 0, B_{t,s} \leq 0$) and 0 otherwise. Using the x^+ notation $I_{num,p}a_p$ can be written as Ia_p^+ and the degree formula becomes

$$deg_{\mathbf{k}}(P(n,\mathbf{k})) \; + \; I\,\Big(\sum_{\substack{p=1\\ \mathbf{b}_p \neq \mathbf{0}}}^{pp} a_p^+ + \sum_{\substack{q=1\\ \mathbf{v}_q \neq \mathbf{0}}}^{qq} (-u_q)^+ + (-A_t)^+\Big) \; +$$

$$\mathbf{J} \;\cdot \Big(\sum_{p=1}^{pp} \mathbf{b}_p^+ + \sum_{q=1}^{qq} (-\mathbf{v}_q)^+ + (-\mathbf{B}_t)^+\Big),$$

which is linear in I and $\mathbf{J}$. We now assume, w.l.o.g., that all entries of $\mathbf{J}$ are equal to an integer J. We write the above degree as $\beta I + \gamma J + \delta$ with suitably defined β, γ, δ. The size of our structureset is $(I+1)(J+1)^r$, so with Theorem 2.15, we only have to show that there exist I, J such that

$$\binom{\beta I + \gamma J + \delta + r}{r} < (I+1)(J+1)^r. \tag{2.16}$$

The binomial coefficient on the l.h.s. of (2.16) is a polynomial in J of degree r with leading coefficient $\frac{\gamma^r}{r!}$. If we choose $I = \lceil \frac{\gamma^r}{r!} \rceil$ then the r.h.s. of (2.16) is a polynomial in J of degree r with leading coefficient $\lceil \frac{\gamma^r}{r!} \rceil + 1$, so the r.h.s. of (2.16) becomes eventually, for large J, larger than the l.h.s.

So a recurrence exists on $S_{I,\mathbf{J}}$ if we choose an integer I such that

$$I \geq \frac{1}{r!}\left(\mathbf{1} \cdot \Big(\sum_{p=1}^{pp} \mathbf{b}_p^+ + \sum_{q=1}^{qq} (-\mathbf{v}_q)^+ + (-\mathbf{B}_t)^+\Big)\right)^r$$

and if $\mathbf{J}$ is large enough. □

The given bound for the order of the recurrence in the main variable is, although slightly better than those given in [WZ92a], not very useful for calculations with multisums. For nearly every proper hypergeometric function with more than one summation variable there exists a structure such that the number of unknowns is smaller than the number of equations. Therefore there usually exist recurrences whose order is much smaller than the predicted one.

For the case of one summation variable, however, the bound for the summation order in n is exact for many examples, and, moreover it is possible to give a bound for the recurrence order in k as well.

Corollary 2.20. *([Wil91]) Let t be a proper hypergeometric term as in Definition 2.1 in the variables n and k (i.e., $r = 1$). There exists a recurrence free of k for F_t of order (I, J) where*

$$\begin{aligned} I &= \sum_{p=1}^{pp} b_p^+ + \sum_{q=1}^{qq} (-v_q)^+ + (-B_t)^+ \\ J &= 1 + \deg_k(P(n,k)) + I\Big(\sum_{\substack{p=1\\ b_p\neq 0}}^{pp} a_p^+ + \sum_{\substack{q=1\\ v_q\neq 0}}^{qq} (-u_q)^+ + (-A_t)^+ - 1\Big) \end{aligned}$$

Proof. With the notations of the above proof, the inequality we have to solve is $\beta I + \gamma J + \delta + 1 < (I+1)(J+1)$. It is easy to see that the above formulas are a solution to this inequality. $\square$

Now we already able to complete the basic algorithm to find recurrences.

Sister Celine's technique with the Wilf-Zeilberger method.

- Start with small values of $I, \mathbf{J}$, and try to find a recurrence for F_t on $S_{I,\mathbf{J}}$.
- If this is not successful, repeat it with higher values of I and $\mathbf{J}$ until a recurrence is found.

The existence theorem guarantees us that this procedure eventually stops. But it soon turns out that the Wilf-Zeilberger method is not very efficient. A much better method (using P-maximal structuresets) can be derived from Verbaeten's work.

2.5 P-maximal Structuresets

In this section we investigate a special kind of structuresets, the P-maximal structuresets. They were introduced by Verbaeten ([Ver76]) and are the optimal structuresets for Sister Celine's technique. To see that the shape of the structuresets has a great influence on the success of Sister Celine's technique, let us consider a concrete example.

To prove Dixon's famous identity

$$\sum_k (-1)^k \binom{2n}{k}^3 = (-1)^n \frac{(3n)!}{n!^3}$$

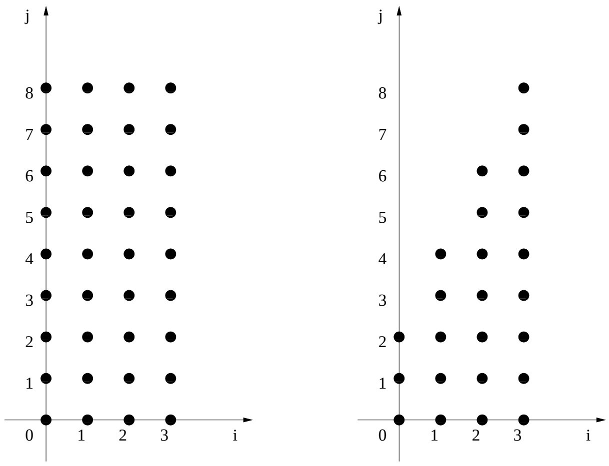

Figure 2.1: The structures $S_{3,8}$ and S_{Dixon} for $(-1)^k\binom{2n}{k}^3$.

with our method we have to find a k-free recurrence for the summand. Of course the method of k-free recurrences is not a serious rival for Zeilberger's fast algorithm: for instance, the Paule-Schorn implementation finds a first order recurrence for the sum within seconds. But we present a single sum example since the main point — that P-maximal structuresets are superior — can be illustrated more easily.

Corollary 2.20 tells us that a recurrence exists for the structureset $S_{3,16}$. This turns out to be too big: the smallest structureset $S_{I,J}$ for which a recurrence exists is $S_{3,8}$, still a rather large set. Geometrically we can interpret this structureset as a part of the two dimensional integer lattice (see Figure 2.1).

The k-free recurrence on $S_{3,8}$ is impressively large:

$(-2+n)^2(-5+2n)^2(-30+315n-1234n^2+2040n^3-1472n^4+384n^5)F(n-3,k-8)+$
$2(-2+n)^2(-5+2n)^2(-48+438n-1499n^2+2262n^3-1564n^4+408n^5)F(n-3,k-7)+$
$4(-2+n)^2(-5+2n)^2(294-2889n+10667n^2-16986n^3+12052n^4-3144n^5)F(n-3,k-6)+$
$2(-2+n)^2(-5+2n)^2(-1680+16650n-61973n^2+99210n^3-70564n^4+18408n^5)F(n-3,k-5)+$
$10(-2+n)^2(-5+2n)^2(462-4587n+17102n^2-27408n^3+19504n^4-5088n^5)F(n-3,k-4)+$
$2(-2+n)^2(-5+2n)^2(-1680+16650n-61973n^2+99210n^3-70564n^4+18408n^5)F(n-3,k-3)+$
$4(-2+n)^2(-5+2n)^2(294-2889n+10667n^2-16986n^3+12052n^4-3144n^5)F(n-3,k-2)+$
$2(-2+n)^2(-5+2n)^2(-48+438n-1499n^2+2262n^3-1564n^4+408n^5)F(n-3,k-1)+$
$(-2+n)^2(-5+2n)^2(-30+315n-1234n^2+2040n^3-1472n^4+384n^5)F(n-3,k)+$
$(4620-60777n+323377n^2-877489n^3+1368286n^4-1301332n^5+768088n^6-274720n^7+54528n^8-4608n^9)F(n-$
$2,k-6)+$
$6(54560-678836n+3449671n^2-9051747n^3+13777595n^4-12881718n^5+7516424n^6-2669792n^7+528240n^8-$
$44640n^9)F(n-2,k-5)+$
$3(1111220-13537307n+67552480n^2-174914274n^3+263743652n^4-244991544n^5+142333856n^6-50424608n^7+$
$9964992n^8-842112n^9)F(n-2,k-4)+$
$2(3901920-47144952n+233568020n^2-601630463n^3+903866249n^4-837509378n^5+485772632n^6-$

$171925088n^7 + 33960720n^8 - 2869920n^9)F(n-2,k-3) +$
$3(1111220 - 13537307n + 67552480n^2 - 174914274n^3 + 263743652n^4 - 244991544n^5 + 142333856n^6 - 50424608n^7 + 9964992n^8 - 842112n^9)F(n-2,k-2) +$
$6(54560 - 678836n + 3449671n^2 - 9051747n^3 + 13777595n^4 - 12881718n^5 + 7516424n^6 - 2669792n^7 + 528240n^8 - 44640n^9)F(n-2,k-1) +$
$(4620 - 60777n + 323377n^2 - 877489n^3 + 1368286n^4 - 1301332n^5 + 768088n^6 - 274720n^7 + 54528n^8 - 4608n^9)F(n-2,k) +$
$(-1620 + 23877n - 145122n^2 + 450785n^3 - 801882n^4 + 864164n^5 - 573048n^6 + 228128n^7 - 49920n^8 + 4608n^9)F(n-1,k-4) +$
$2(23760 - 273456n + 1309143n^2 - 3432274n^3 + 5460939n^4 - 5490946n^5 + 3498672n^6 - 1364896n^7 + 296400n^8 - 27360n^9)F(n-1,k-3) +$
$6(82980 - 1012233n + 5179098n^2 - 14316173n^3 + 23624922n^4 - 24292164n^5 + 15663016n^6 - 6141184n^7 + 1335360n^8 - 123264n^9)F(n-1,k-2) +$
$2(23760 - 273456n + 1309143n^2 - 3432274n^3 + 5460939n^4 - 5490946n^5 + 3498672n^6 - 1364896n^7 + 296400n^8 - 27360n^9)F(n-1,k-1) +$
$(-1620 + 23877n - 145122n^2 + 450785n^3 - 801882n^4 + 864164n^5 - 573048n^6 + 228128n^7 - 49920n^8 + 4608n^9)F(n-1,k) +$
$n^2(-1+2n)^2(5475 - 16711n + 20026n^2 - 11768n^3 + 3392n^4 - 384n^5)F(n,k-2) +$
$2n^2(-1+2n)^2(22644 - 68651n + 81827n^2 - 47902n^3 + 13780n^4 - 1560n^5)F(n,k-1) +$
$n^2(-1+2n)^2(5475 - 16711n + 20026n^2 - 11768n^3 + 3392n^4 - 384n^5)F(n,k) \;=\; 0.$

It was computed by the Mathematica program `FindRecurrence`, which is described in Chapter 4. The resulting recurrence for the sum is a second order recurrence and equals

$$b_2(n)\,\mathrm{SUM}(n-2) \;+\; b_1(n)\,\mathrm{SUM}(n-1) \;+\; b_0(n)\,\mathrm{SUM(n)} \;=\; 0$$

where

$$\begin{aligned} b_2(n) &= 9(3n-5)(3n-4)(6n-11)(6n-7)(36n^3-48n^2+17n-2) \\ b_1(n) &= 6(1296n^7-8208n^6+21096n^5-28428n^4+21577n^3-9168n^2+2013n-180) \\ b_0(n) &= n^2(2n-1)^2(36n^3-156n^2+221n-103). \end{aligned}$$

If we investigate the k-free recurrence we see that a couple of elements of $S_{3,8}$ do not occur in the recurrence: their polynomial coefficients $a_{i,j}(n)$ are zero. Deleting those elements from $S_{3,8}$, we obtain the smaller structure S_{Dixon}, which is shown on the right of Figure 2.1. It is an enormous waste of computation time to take $S_{3,8}$ as a structure if the smaller S_{Dixon} suffices: solving a 25×24 equation system in 76 seconds compared to solving a 43×36 system in 3117 seconds.

But is there a systematic way to find such more complex, i.e., non-rectangular, structuresets? Yes, there is. A closer look at the degree formula for the associated polynomial tells us that certain structuresets, the P-maximal structuresets, are optimal — in the sense that for any larger structureset the degree of the associated polynomial and therefore the number of equations is higher. And, most important, we can construct P-maximal structuresets: given an arbitrary structureset we can find the smallest P-maximal structureset containing it (*Verbaeten completion*).

To define P-maximal structuresets we need the notion of structure functions and structure hyperplanes. Structure functions are linear functions $SF(i, \mathbf{j})$, which we introduce to simplify the degree formula for the associated polynomial.

Definition 2.21. Let t be a proper hypergeometric term in the variables n and $\mathbf{k} = (k_1, \ldots, k_r)$, and let $g \in \mathbb{Z}$ and $\mathbf{h} \in \mathbb{Z}^r$ with $gcd(g, \mathbf{h}) = 1$. The function $SF(i, \mathbf{j}) = gi + \mathbf{h} \cdot \mathbf{j}$ defined on $\mathbb{R}^{r+1}$ is called a *structure function* of t, iff at least one of the following conditions is satisfied:

1. there is a numerator factorial expression $a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p$ of t with $\mathbf{b}_p \neq \mathbf{0}$ such that $g = a_p / gcd(a_p, \mathbf{b}_p)$ and $\mathbf{h} = \mathbf{b}_p / gcd(a_p, \mathbf{b}_p)$;
2. there is a denominator factorial expression $u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q$ of t with $\mathbf{v}_q \neq \mathbf{0}$ such that $g = -u_q / gcd(u_q, \mathbf{v}_q)$ and $\mathbf{h} = -\mathbf{v}_q / gcd(u_q, \mathbf{v}_q)$;
3. for the factorial difference $(A_t, \mathbf{B}_t)$ of t, we have $g = -A_t / gcd(A_t, \mathbf{B}_t)$ and $\mathbf{h} = -\mathbf{B}_t / gcd(A_t, \mathbf{B}_t)$.

We say that a structure function SF *corresponds* to a $a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p$ (or to $u_q n + \mathbf{v}_q \cdot \mathbf{k} + w_q$ or to $(A_t, \mathbf{B}_t)$) iff the conditions of 1 (or 2 or 3) are satisfied. Note that SF may correspond to several expressions.
We define the *multiplicity* of a structure function SF as

$$\omega_{t,SF} \;=\; \sum gcd(a_p, \mathbf{b}_p) \;+\; \sum gcd(u_q, \mathbf{v}_q) \;+\; gcd(A_t, \mathbf{B}_t),$$

where the sum ranges over the expressions to that SF corresponds.
The set of all structure functions of t is denoted by $\mathcal{SF}_t$.

We use the concept of structure functions to unify the notion of boundary points. It is easy to see that if $(I, \mathbf{J})$ is a boundary point — either numerator or denominator or difference — then for the structure function SF corresponding to the defining term of the boundary point, $SF(I, \mathbf{J}) \geq SF(i, \mathbf{j})$ for all $(i, \mathbf{j})$ from the structureset. So the boundary points can now be defined as

Definition 2.22. Let t be a proper hypergeometric term and S a structureset for t, and let $SF \in \mathcal{SF}_t$. A point $(I, \mathbf{J}) \in S$ is called a boundary point of SF and S iff

$$SF(I, \mathbf{J}) \;\geq\; SF(i, \mathbf{j}) \qquad \text{for all } (i, \mathbf{j}) \in S.$$

Such a point will be denoted by denoted by $(I_{SF,S}, \mathbf{J}_{SF,S})$. The hyperplane $H_{SF,S} = \{(i, \mathbf{j}) \in \mathbb{R}^{r+1} \mid SF(i, \mathbf{j}) = SF(I_{SF,S}, \mathbf{J}_{SF,S})\}$ is called the *structure hyperplane* of SF and S.

For single summation terms, i.e., r =1, we usually call a structure hyperplane a structureline. The simplest property of a structure function SF is that it is constant on $H_{SF,S}$ and also on every hyperplane parallel to $H_{SF,S}$. So, if $(I_{SF,S}, \mathbf{J}_{SF,S})$ is a boundary point of SF and a structure set S, there may be several other points in S that can serve as boundary point for SF — all the points in S that lie on the hyperplane $H_{SF,S}$.

We use this notation to give a more compact version of the degree formula, making it more transparent that the degree is just the sum over the maxima of linear functions plus a constant.

Lemma 2.23. *Let t be a proper hypergeometric term in n and $\mathbf{k}$ with polynomial part $P(n,\mathbf{k})$, and let S be a structureset for t. The degree of the associated polynomial $P_{t,S}$ can now be written as*

$$\deg_{\mathbf{k}}(P(n,\mathbf{k})) \;+\; \sum_{\mathrm{SF}\in\mathcal{SF}_t} \omega_{t,\mathrm{SF}}\ \mathrm{SF}(I_{\mathrm{SF},S},\mathbf{J}_{\mathrm{SF},S}).$$

Given a term t and a structureset S, we see that — since the degree formula for the associated polynomial only depends on the boundary points — we can include all the integer lattice points $(i,\mathbf{j})$ that satisfy $SF(i,\mathbf{j}) \le SF(I_{SF,S},\mathbf{J}_{SF,S})$ for every structure function SF to the structureset without increasing the degree of the associated polynomial. Geometrically this completion process corresponds to taking the intersection of all the integer lattice points "below" the hyperplanes parallel to H_{SF} going through the corresponding boundary points — the integer lattice points of a convex set.

Definition 2.24. Let t be a proper hypergeometric term and S be a structureset. S is called *P-maximal* iff there does not exist a structureset S_1 such that $S \subsetneq S_1$ and $\deg(P_{t,S}) = \deg(P_{t,S_1})$.

It is immediately clear that following holds.

Lemma 2.25. *Let t be a proper hypergeometric term and S be a structureset. S is P-maximal iff*

$$S \;=\; \{(i,\mathbf{j})\in\mathbb{Z}^{r+1} \mid \mathrm{SF}(i,\mathbf{j}) \le \mathrm{SF}(I_{\mathrm{SF},S},\mathbf{J}_{\mathrm{SF},S}) \quad \textit{for every } \mathrm{SF}\in\mathcal{SF}_t\}.$$

Note that for trivial terms, like $(n-k)!$, it may happen that a P-maximal set is infinite, so that a P-maximal structureset does not exist (a structureset is by definition finite). This case is algorithmically simple to handle (see below). But usually P-maximal sets are finite.

As an example, let us compute and draw the boundary points and structure lines for the Dixon term

$$(-1)^k\binom{2n}{k}^3 \;=\; (-1)^k\frac{(2n)!^3}{k!^3(2n-k)!^3}.$$

The structure functions of this term are

$$\begin{aligned} SF_1(i,j) &= -j \\ SF_2(i,j) &= -2i+j \\ SF_3(i,j) &= i \end{aligned}$$

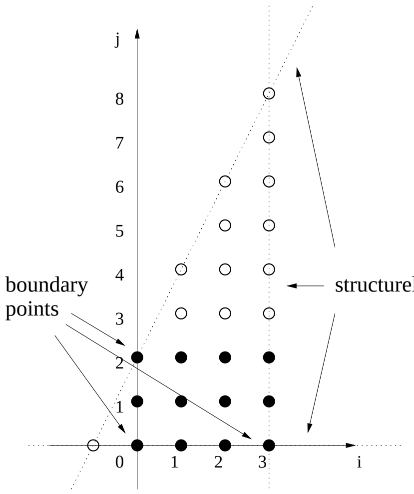

Figure 2.2: The smallest P-maximal structureset of $(-1)^k\binom{2n}{k}^3$ containing $S_{3,2}$.

where the structure functions one and two originate from the denominator factorials, and the third structure function from the factorial difference.

Let us take $S_{3,2}$ as structureset; a set of boundary points for this structureset and the three structure functions is $\{(0,0),(0,2),(3,0)\}$. (Note that, e.g., $\{(1,0),(0,2),(3,1)\}$ would be another set of boundary points.) Figure 2.2 shows the structureset and the structurelines through the boundary points.

In Figure 2.2 the points of the structureset $S_{3,2}$ are the black points, whereas the white points are those points that make the set P-maximal. They can be added to the structureset without increasing the degree of $P_{t,S_{3,2}}$. It turns out that this enlarged structureset is — up to the additional point $(-1,0)$ — the structureset S_{Dixon} (see Figure 2.1) on which a recurrence already exists. Starting with a small structureset we were able to compute nearly exactly the structureset which allows to find a solution much faster.

It was not by chance that the smallest structure for Dixon's summand is nearly a P-maximal structureset. The following theorem tells us that for irreducible terms with one summation variable the convex hull of a structure containing no superfluous points, always contains a nontrivial piece of every structureline.

Theorem 2.26. *Let t be an irreducible proper hypergeometric term in n and k, let S be a structureset for t, and let $a_{i,j}(n)$ be polynomials, all nonzero, such that the associated polynomial $P_{t,S} = \sum_{(i,j)\in S} a_{i,j}(n)p_{i,j}(n,k) = 0$. Let SL be a structureline of t and S. Then there exist at least two points in $S \cap SL$.*

Proof. If SL corresponds to a (numerator or denominator) factorial expression $(an + bk + c)$ then, with $(I, J) \in SL$ a boundary point, the following divisibility relations for $\sigma = (an + bk+$

$c - aI - bJ$) hold:

$$\begin{aligned} \sigma \not| \; p_{i,j} & \quad \text{for every } (i,j) \in S \cap SL, \\ \sigma \mid p_{i,j} & \quad \text{for every } (i,j) \in S \setminus SL. \end{aligned}$$

These divisibility relations hold because t is irreducible and because of formula (2.13) for the associated polynomial. If we set $\sigma = 0$, that is substituting $k = \frac{a}{b}n + \frac{c}{b}$ (remember that b is nonzero according to the definition of a structure function) in the polynomial equation $\sum_{(i,j)\in S} a_{i,j}(n)p_{i,j}(n,k) = 0$, then it becomes $\sum_{(i,j)\in S\cap SL} a_{i,j}(n)p_{i,j}(n, \frac{a}{b}n + \frac{c}{b}) = 0$. Every summand is a nonzero polynomial, so the equation can only hold if there are at least two elements in $S \cap SL$.

If SL corresponds to the factorial difference then the degree formula (2.14) implies that for every $(i,j) \in S \setminus SL$ and every $(I,J) \in S \cap SL$ we have $\deg_k(p_{i,j}) < \deg_k(p_{I,J})$. We compare the highest coefficient of k in the associated polynomial with zero and get an equation $\sum_{(i,j)\in S\cap SL} c_{i,j}(n)a_{i,j}(n) = 0$, where not all $c_{i,j}(n)$ are zero. So again S must contain at least two elements of SL. □

The method of computing the smallest P-maximal structureset containing a given structureset of course also works for examples with r summation variables, involving structure hyperplanes rather than structurelines. We use this *Verbaeten completion* of a structureset to improve the Wilf-Zeilberger method for computing a k-free recurrence; informally this method is described as follows.

Sister Celine's technique with Verbaeten completion.

- Compute for the term t and the structureset $S_{I,\mathbf{J}}$ (the rectangular structuresets) the structure functions and the boundary points.
- Compute the smallest P-maximal structureset S containing $S_{I,\mathbf{J}}$ (the Verbaeten completion). If the P-maximal set is infinite, take only a finite part of this set, with a size such that the condition for the existence of a nontrivial nullspace is fulfilled (i.e., the number of equations is less than the number of unknowns).
- Try to find a recurrence for F_t on this new structureset S. If no recurrence was found increase I and $\mathbf{J}$ and try again.

For irreducible proper hypergeometric terms it is clear what the Verbaeten completion technique means for the size of the equation system: the fact that the degree of the associated polynomial remains constant, guarantees us that the number of equations is the same as before, whereas the number of unknows in the equation system has increased. For reducible terms, however, we can not make such a clear statement, since it can happen that, due to some unpredictable cancellations in the associated polynomial, the number of equations increases. But this increase is small compared to the increase in the number of unknowns. Anyway, in both cases (irreducible as well as reducible) Verbaeten completion has the following advantages: the equation system that has to be solved to find a solution is much smaller, and the number of unsuccessful tries we have to make to find this system, too, is smaller.

Sister Celine's technique with Verbaeten completion usually is, as empirical results show, about 10 to 100 times faster than Sister Celine's technique working only with rectangular structuresets $S_{I,\mathbf{j}}$ suggested by Wilf and Zeilberger[2]. In the example of Dixon's sum the speed-up factor is about 30. To put it somehow stronger: computational experience shows that only with Verbaeten completion k-free recurrences can be computed in reasonable time.

2.6 Verbaeten's Existence Theorem

In his PhD thesis [Ver76] P. Verbaeten gives a proof for the existence of k-free recurrence relations for proper hypergeometric terms with only one summation variable and without polynomial part by constructing a structure from a special P-maximal structureset: the minimal structure. Since this construction uses to a great part arguments from plane geometry, this theory is not, at least not in a straightforward way, generalizeable to the multivariate case. This means that the following existence theory does not contribute anything to algorithms for the multivariate case, and is only included for the sake of completeness. Thus we proceed rather short and sketchy. For this condensed description we follow the elegant presentation in J. Hornegger's diploma thesis [Hor92]. Hornegger and his advisor V. Strehl considerably simplified Verbaeten's construction (e.g., by using the relation (2.19) and Theorem 2.27).

First we introduce some notations used throughout this subsection. A *polygon* in $\mathbb{R}^2$ is the convex hull of a finite subset of $\mathbb{R}^2$. Let S be a bounded subset of $\mathbb{R}^2$. With ∂S and $\mathring{S}$ we denote as usual the boundary respectively the interior of S. $|S|$ denotes the number of integer lattice points in S. $\mathcal{A}(S)$ is defined as the area of S.

In this section let t be a proper hypergeometric term in the variables n and k without polynomial part. Let $SF_i(x,y) = \alpha_i x + \beta_i y$ for $1 \le i \le s$ be the structure functions of t and ω_i their multiplicity. For every structure function we define the *structurevector* $SV_i = \omega_i \binom{-\beta_i}{\alpha_i}$. The structurevectors are such that the corresponding structure function is constant along the vector and decreases on the left side of it (viewing in direction of the vector). So, if we sort the structurevectors with respect to the angles they enclose with the nonnegative real axis[3], we can form a convex polygon generated in circular manner by the structurevectors (see Figure 2.3). Every corner of this polygon has the formula $\sum_{i=1}^{l} \omega_i \binom{-\beta_i}{\alpha_i}$ for some l, so the corners are integer lattice points. By definition of the factorial difference this vectors really form a (closed) polygon, i.e., the sum of all the vectors is zero. The set of integer lattice points of this polygon (including the points on the boundary) is a P-maximal structureset. (With one exception: the minimal structure of terms without finite P-maximal sets is not P-maximal. But in this case the enclosing polygon is degenerate; it is only a line.)

This set of integer lattice points is called the *minimal structure*[4] and denoted by M. Note that there are exactly $\omega_i + 1$ integer lattice points on the edge SV_i of M.

---

[2]But note that if the structuresets $S_{I,\mathbf{J}}$ are already P-maximal, then there is no speed up at all.
[3]In the following, w.l.o.g., we assume that they are already sorted.
[4]Note that the minimal structure is not necessarily a structure, but only a structureset.

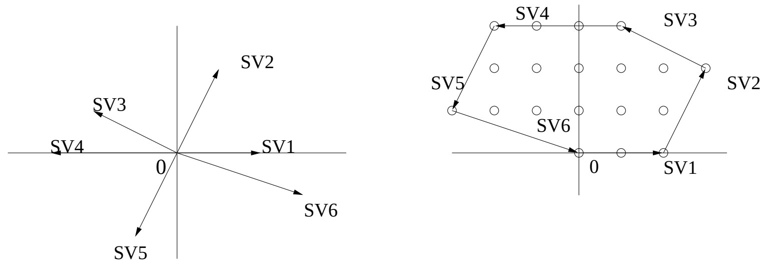

Figure 2.3: The structurevectors and how they form the minimal structure M.

More generally, we easily see that every P-maximal structureset is bounded by multiples of the structurevectors, i.e., is (possibly after a shift in the plane) contained in a polygon with corners $\sum_{i=1}^{l} \lambda_i SV_i$, where the λ_i are positive real numbers. Of course the total sum of the vectors must be zero to make sure that the sequence of vectors form a polygon. Note that the corners of such polygons are not necessarily integer lattice points, but it is tacitly, and w.l.o.g., assumed that at least one integer lattice point is on every edge of the polygon. We denote such a structureset in the following with M_λ.

The boundary point of M_λ for the structure function SF_i is one of the integer lattice points on the corresponding structure vector. So the degree of the associated polynomial in k, now denoted by $\deg(M_\lambda)$ is

$$\deg(M_\lambda) = \sum_{i=1}^{s} \omega_i SF_i\left(\sum_{j=1}^{i-1} \lambda_j SV_j\right) = \sum_{1\le j<i\le s} \lambda_j \omega_i \omega_j \left(\alpha_j \beta_i - \alpha_i \beta_j\right). \tag{2.17}$$

The area of the polygon, that contains the structureset M_λ is denoted by $\mathcal{A}\left(M_\lambda\right)$ and can be found by triangulation. [5]

$$\begin{aligned} \mathcal{A}\left(M_\lambda\right) &= \frac{1}{2}\sum_{l=1}^{s-1} \begin{vmatrix} -\sum_{i=1}^{l} \lambda_i \omega_i \beta_i & -\sum_{j=1}^{l+1} \lambda_j \omega_j \beta_j \\ \sum_{i=1}^{l} \lambda_i \omega_i \alpha_i & \sum_{j=1}^{l+1} \lambda_j \omega_j \alpha_j \end{vmatrix} \\ &= \frac{1}{2}\sum_{l=1}^{s-1}\sum_{i=1}^{l} \lambda_i \lambda_{l+1} \omega_i \omega_{l+1} \left(\alpha_i \beta_{l+1} - \alpha_{l+1}\beta_i\right) \\ &= \frac{1}{2}\sum_{1\le i<l\le s} \lambda_i \lambda_l \omega_i \omega_l \left(\alpha_i \beta_l - \alpha_l \beta_i\right) \end{aligned} \tag{2.18}$$

So the area can be expressed in similar terms as the degree formula, and immediately we get for the minimal structure M (all λ_i equal to 1) the beautiful relationship

$$\deg(M) = 2\mathcal{A}\left(M\right). \tag{2.19}$$

[5]The attentive reader may have noticed that the following sum should only run up to $s-2$ to express the area, but the last determinant is zero, and we add it to make the resulting formula look nicer.

To continue with the construction, we need to know the number of integer lattice points inside a polygon. The next theorem establishes a relationship between this number and the area of the polygon, in the case that the polygon has integer lattice corners.

Theorem 2.27. *([Hor92]) Let P be a polygon whose corners are integer lattice points. Then we have the following formula*

$$2\mathcal{A}(P) \;=\; |P| + |\overset{\circ}{P}| - 2 \;=\; 2|P| - |\partial P| - 2.$$

Proof. The second equation is trivial, so we just prove the first one. We split the proof into two parts, first showing that the formula holds for a polygon A if it holds for two subpolygons, and in the second step showing that the formula holds for the smallest such subpolygons (special triangles).

1. Let A_1 and A_2 be two polygons that are disjoint except a common connected piece of boundary $\partial A_1 \cap \partial A_2$ with integer lattice points as corner on each end: P_1 respectively P_2. Then the polygon $A = A_1 \cup A_2$ has this common boundary as inner points with the exception of P_1 and P_2 that remain boundary points. Our claim is that if the above formula holds for both A_1 and A_2 then it holds for the union A:

$$\begin{aligned}
2\mathcal{A}(A_1 \cup A_2) \;&=\; 2\mathcal{A}(A_1) + 2\mathcal{A}(A_2) \\
&= |A_1| + |\overset{\circ}{A}_1| - 2 \;+\; |A_2| + |\overset{\circ}{A}_2| - 2 \\
&= |A_1| + |A_2| - |\partial A_1 \cap \partial A_2| \;+\; |\overset{\circ}{A}_1| + |\overset{\circ}{A}_2| + |\partial A_1 \cap \partial A_2| - 2 \;-2 \\
&= |A| + |\overset{\circ}{A}| - 2.
\end{aligned}$$

2. We triangulate the given polygon P and, if there is an integer lattice point in or on the triangle other then the corners, we split up these triangles again into triangles until we are left with triangles with integer grid points as corners that have no other integer lattice points in or on them. It suffices to prove the claim for these triangles, i.e., to show that their area is $\frac{1}{2}$.

Let, w.l.o.g., $(0,0)$, $a_1 = (x_1, y_1)$, and $a_2 = (x_2, y_2)$ be the corners of such a triangle. The condition that there are no other grid points in the triangle is equivalent to the condition that the vectors a_1 and a_2 generate the whole integer lattice grid $\mathbb{Z}^2$, i.e., $\mathbb{Z}^2 = \{na_1 + ma_2 \mid n, m \in \mathbb{Z}\}$. So the transformation matrix $A = \left(\begin{smallmatrix} x_1 & x_2 \\ y_1 & y_2 \end{smallmatrix}\right)$ maps $\mathbb{Z}^2$ into itself. It follows that the inverse transformation, too, is an integer matrix. The determinant $\det A = x_1 y_2 - y_1 x_2$ is therefore 1 or -1. Since the area of the triangle is half the absolute value of this determinant the theorem is proved. $\square$

The *inner points* of the structureset M_λ, denoted by $\overset{\circ}{M}_\lambda$, are defined as the integer lattice points of the interior of the enclosing structurevector polygon. We are now able to formulate the first existence theorems.

Theorem 2.28. *If the minimal structure M has no inner points, then M is already a structure for t.*

Proof. By using Theorem 2.27 and equation (2.19) we see that the criterion for the existence of a nontrivial solution (Corollary 2.16) is satisfied: $\deg(M) = 2\mathcal{A}(M) = |M| - 2$. □

If there are inner points in M then it is usually not a structure. From now on we assume that M contains inner points. We can find a P-maximal structure, if we double the length of the sides of M. Let all λ_i be equal to 2, and call the resulting structureset M_2.

Theorem 2.29. *M_2 is a structure for t.*

Proof. Again a nontrivial solution exists because of

$$\deg(M_2) \;= \mathcal{A}(M_2) \;=\; \frac{1}{2}\left(|M_2| + |\mathring{M}_2| - 2\right) \;<\; |M_2| - 1,$$

and Corollary 2.16. □

Although we have found a structure for t, we can achieve more. It turns out that the set M_2 is usually far too large (see, e.g., Figure 2.4). The reason is that $|M_2|$ is a very large upper bound for for the number of inner points of M_2.

But we are able to find a smaller structure, if we drop the condition of P-maximality. The idea is that we elongate the minimal structure in the direction of the summation variable.

Theorem 2.30. *([Ver76]) Let M be the minimal structure of t, and let $S = \{(0, j) \mid j \in [0\,..\,|\mathring{M}|]\}$. Then $M_S = M + S = \{(i, j_1 + j_2) \mid (i, j_1) \in M \text{and } j_2 \in [0\,..\,|\mathring{M}|]\}$ is a structure for t.*

Proof. Let $SV_i = \binom{-\beta_i}{\alpha_i}$ bet the structurevectors of t, let $M_{S,P}$ and S_P be the smallest P-maximal structuresets containing M_S respectively S, and let $\lambda_i SV_i$ be the vectors that generate the polygon containing S_P. It is easy to see that the polygon of $M_{S,P}$ is generated by the vectors $(\lambda_i + 1)\,\mathrm{SV}_i$ and that its area $\mathcal{A}(M_{S,P})$ is $\mathcal{A}(M_S) + \mathcal{A}(S_P)$.

The degree of M_S is, since the structure functions are linear, easily determined.

$$\begin{aligned}
\deg(M + S) \;&=\; \sum_{SF \in \mathcal{SF}} \max_{(i, j+j_0) \in M_S} SF(i, j + j_0) \\
&=\; \sum_{SF \in \mathcal{SF}} \big(\max_{(i,j) \in M} SF(i, j) + \max_{(0,j_0) \in S} SF(0, j_0)\big) \;=\; \deg(M) + \deg(S)
\end{aligned}$$

The area of M_S is equal the area of $M_{S,P}$ minus the area of S_P. With the abbreviation $\delta_{i,j} = \omega_i\omega_j(\alpha_i\beta_j - \alpha_j\beta_i)$ we get by using (2.18)

$$\begin{aligned}\mathcal{A}(M_S) &= \mathcal{A}(M_{S,P}) - \mathcal{A}(S_P)\\ &= \frac{1}{2}\sum_{i<j}(\lambda_i+1)(\lambda_j+1)\delta_{i,j} \;-\; \frac{1}{2}\sum_{i<j}\lambda_i\lambda_j\delta_{i,j}\\ &= \sum_{i<j}\lambda_i\delta_{i,j} \;+\; \frac{1}{2}\sum_{i<j}\delta_{i,j} \;=\; \deg(S) \;+\; \frac{1}{2}\deg(M).\end{aligned}$$

Above we used that $\sum_{i<j}\lambda_i\delta_{i,j} = \sum_{i<j}\lambda_j\delta_{i,j}$, which holds because $\delta_{i,j} = -\delta_{j,i}$ and $\sum_i \delta_{i,j} = \sum_j \delta_{i,j} = 0$.
With

$$\begin{aligned}|\partial M_S| &= |\partial M| + 2(|S|-1) \;=\; |\partial M| + 2|\mathring{M}| \;=\; 2\mathcal{A}(M)+2\\ &= \deg(M)+2\end{aligned}$$

we have, since M_S has only integer lattice corners,

$$\begin{aligned}\deg(M_S) &= \deg(M)+\deg(S) \;=\; \mathcal{A}(M_S) + \frac{1}{2}\deg(M)\\ &= |M_S| - \frac{1}{2}|\partial M_S| - 1 + \frac{1}{2}\deg(M) \;=\; |M_S| - 2\end{aligned}$$

which, by Corollary 2.16, completes the proof. □

This structure is smaller than M_2 (see the example below), especially the order of the resulting recurrence in the main variable is smaller. This order is just the maximal first coordinate of the minimal structure, i.e., the sum over the positive values of the first components of the structure vectors:

$$\sum_{p=1}^{pp} b_p^+ \;+\; \sum_{q=1}^{qq}(-v_q)^+ \;+\; \Big(-\sum_{p=1}^{pp} b_p + \sum_{q=1}^{qq} v_q\Big)^+.$$

This is exactly the degree bound given in Corollary 2.20.

As an example let us compute the minimal structure M, the structure M_2 and the structure M_S for the summand $(-1)^k\frac{(2n)!^3}{k!^3(2n-k)!^3}$ of Dixon's sum (Figure 2.4). The structurevectors (computed from the structure functions given above) are (already sorted) $\binom{1}{0}$, $\binom{0}{1}$, and $\binom{-1}{-2}$. The multiplicities of the structure functions are 3, 6, and 3. From this we can compute M and M_2. To compute M_S we count the inner points of M: there are four. Figure 2.4 shows that M is too small to be a structure, and that M_2 is too large to be useful. We also see that the structure M_S is smaller than the structure $S_{3,16}$, which is predicted by Corollary 2.20. The size of the equation system for M_S is 31×32. This is even smaller then the size of the system for $S_{3,8}$, which is the smallest structure of the form $S_{I,J}$. It follows that, at least for this special example, Verbaeten's existence theory is superior to the Wilf-Zeilberger theory.

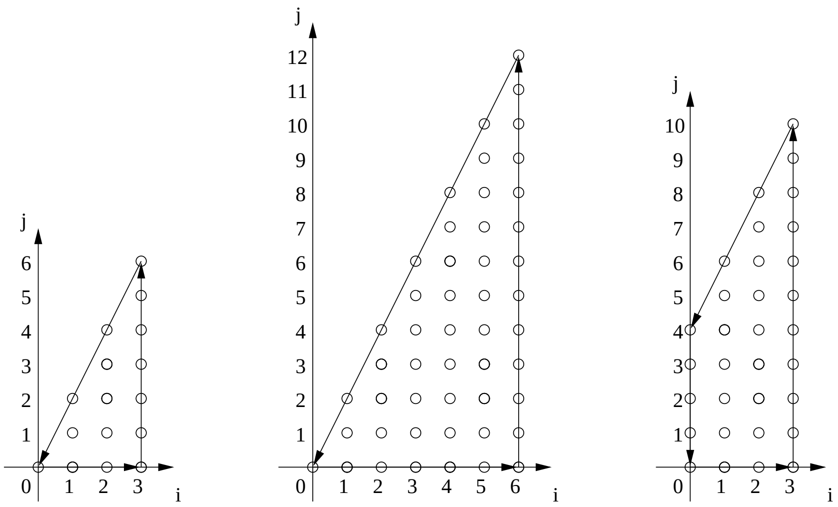

Figure 2.4: M, M_2, and M_S for $(-1)^k\binom{2n}{k}^3$.

Both existence theorems, Theorem 2.19 as well as Theorem 2.30, use the size of the equation system as the criterion for the existence of a nontrivial nullspace (Theorem 2.15). It is clear that in this way we cannot find S_{Dixon}, which the smallest structure for Dixon's summand (see Figure 2.1), a priori. The corresponding equation system has size 25×24: it has a nontrivial solution although the number of equations is larger than the number of unknowns. Not only that this happens almost always for terms with several summation variables, the size of the equation systems such that the number of unknowns exceeds the number of equations is incredibly large — much larger than the equation systems that actually have nontrivial solutions. This implies that an existence theory that predicts structures for summands with several summation variables is of little algorithmic value. It is far more efficient just to start with small structuresets and increase them in case no solution has been found. So we did not make a serious attempt to generalize Verbaeten's existence theory to the multivariate case.

2.7 Recurrences for Binomial Terms

We already mentioned that there is subtle detail in the relationship between the binomial function $\binom{n}{k}$ and its interpretation as a proper hypergeometric term $\frac{\Gamma(n+1)}{\Gamma(k+1)\Gamma(n-k+1)}$, namely that the term is not well-defined for $n \in -\mathbb{N}$. Therefore the k-free recurrence does not necessarily hold for these critical values. Why is it interesting to know whether the recurrence holds for such values or not?

- It's natural: The k-free recurrence for $\frac{n!}{k!(n-k)!}$ that we find with Sister Celine's technique is $F(n,k) - F(n-1,k) - F(n-1,k-1) = 0$, which is also the basic recurrence for the binomial coefficient. So the question for the larger region of validity of a such found recurrence is a natural one.

- It's necessary: Many summands of multivariate sums contain functions like $\binom{i+j}{i}$ where i and j are summation variables. If we sum w.r.t. $i, j \in [0\,..\,n]$ we have to sum the recurrence over a slightly larger domain to get standard boundary conditions (see Section 3.3). It is therefore necessary to know that the recurrence holds for $i + j < 0$.

We would like to show that a binomial function satisfies a given recurrence at the singular values. But this is not always possible. Take the trivial example $\binom{n}{n}$. The corresponding proper hypergeometric term is $\frac{n!}{n!}$ and the simplest recurrence found is $F_t(n) - F_t(n-1) = 0$, which does not hold for the binomial coefficient for $n = 0$. Another function, where exactly the same happens, is $\binom{n}{k}\binom{k}{n}$. The reason for the failure is obvious: some of the factorials in the proper hypergeometric term cancel.

But the problem of singular values arises also in examples that are not so trivial. Consider the following three sums.

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\binom{i+j}{i}^2\binom{2n-i-j}{n-i}^2 \tag{2.20}$$

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\binom{i+j}{j}^2\binom{2n-i-j}{n-j}^2 \tag{2.21}$$

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\binom{i+j}{i}\binom{i+j}{j}\binom{2n-i-j}{n-i}\binom{2n-i-j}{n-j} \tag{2.22}$$

The three sums are equal, since the summands are the same within the summation range. But outside of the summation range the summands are three different functions. All three summands have the same factorial interpretation:

$$\frac{(i+j)!^2(2n-i-j)!^2}{i!^2 j!^2 (n-i)!^2 (n-j)!^2}.$$

We compute a recurrence for the proper hypergeometric function, and in order to get a recurrence for one of the above sums we have to know that the summands satisfies this recurrence outside of the summation range. For which of the summands does the recurrence hold?

We have the following general problem. Given a (not necessarily proper) hypergeometric function $F(n, \mathbf{k})$ and the rational functions $R_{i,\mathbf{j}} = F(n-i, \mathbf{k}-\mathbf{j})/F(n, \mathbf{k})$, show that the recurrence relation $\sum_{i,\mathbf{j}} a_{i,\mathbf{j}} R_{i,\mathbf{j}} = 0$ implies the recurrence relation $\sum_{i,\mathbf{j}} a_{i,\mathbf{j}} F(n-i, \mathbf{k}-\mathbf{j}) = 0$. The argument we used in Theorem 2.8 to prove this for proper hypergeometric functions cannot be applied here: it is no longer true that $F(n, \mathbf{k}) \neq 0$ implies that every $R_{i,\mathbf{j}}$ is well-defined. Just consider the binomial coefficients $\binom{0}{0}$ and $\binom{-1}{0}$. Of course one could find those critical values and (if there are only finitely many) check that the recurrence holds for them by plugging in and evaluating, but we would prefer a more general and less tedious method.

In the following we present several ways of extending — not necessarily k-free — recurrences to singular values. We will take advantage of the more general definition of term evaluation. A proper hypergeometric function $F_t(n, \mathbf{k})$ is defined for almost all complex values of n and $\mathbf{k}$ and a recurrence found by Sister Celine's technique holds for all these values.

2.7.1 Binomials as Polynomials

Recall that for every fixed integer k the binomial coefficient $\binom{n}{k}$ is a polynomial in n of degree at most k. Therefore any polynomial recurrence like

$$\binom{n}{k} - \binom{n-1}{k} - \binom{n-1}{k-1},$$

for fixed integer k, is a polynomial in n. If this polynomial is zero for enough values of n, then it is identically zero. We already know that for every fixed integer k the Pascal triangle recurrence holds for all $n \in \mathbb{C} \setminus \{-1, -2, \ldots\}$, so the polynomial is zero and therefore the recurrence also holds for $n \in \{-1, -2, \ldots\}$. This is called a proof by *polynomial argument.*

The polynomial argument also works for the summand of (2.20), which for fixed integers n and i is a polynomial in j, and of (2.21), a polynomial in i for fixed integers n and j. But it fails for (2.22), as it fails for all binomial terms that do not have a variable in the upper argument of every binomial coefficient that does not occur in any lower argument.

2.7.2 Limits of Proper Hypergeometric Functions

Taking limits of proper hypergeometric functions is the most frequently used method. It is based on the fact that every proper hypergeometric function $F_t(n, \mathbf{k})$ is continuous in D_t.

Now let $F(n, \mathbf{k})$ be an extension of $F_t(n, \mathbf{k})$, i.e. let F be a function defined on $D \supset D_t$ such that F and F_t coincide on D_t. Suppose we have the recurrence

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}} F_t(n-i, \mathbf{k}-\mathbf{j}) \;=\; 0$$

for F_t and we want to show that it also holds for F at the value $(n, \mathbf{k}) \in D$ with $(n, \mathbf{k}) \notin D_t$.

For this purpose we take a sequence $(n_l, \mathbf{k}_l)_{l \in \mathbb{N}}$ such that $(n_l, \mathbf{k}_l) \in D_t$ and $(n_l - i, \mathbf{k}_l - \mathbf{j}) \in D_t$ for all l and for all $(i, \mathbf{j})$ and such that $\lim_{l \to \infty}(n_l, \mathbf{k}_l) = (n, \mathbf{k})$. If $\lim_{l \to \infty} F_t(n_l - i, \mathbf{k}_l - \mathbf{j}) = F(n - i, \mathbf{k} - \mathbf{j})$ for all $(i, \mathbf{j})$, it follows from the continuity of the polynomial coefficients $a_{i,\mathbf{j}}$ that

$$\sum_{(i,\mathbf{j}) \in S} a_{i,\mathbf{j}} F(n-i, \mathbf{k}-\mathbf{j}) \;=\; 0.$$

We have quite a freedom of how to take the limit, we even may take iterated limits. Note that depending on how the limit is taken we possibly get different results, e.g., for $(n, i, j) \in \mathbb{Z}^3$ we have

$$\begin{aligned}
\lim_{\tilde{i} \to i} \frac{(\tilde{i}+j)!^2(2n-\tilde{i}-j)!^2}{\tilde{i}!^2 j!^2 (n-\tilde{i})!^2 (n-j)!^2} &= \binom{i+j}{j}^2 \binom{2n-i-j}{n-j}^2 \\
\lim_{\tilde{j} \to j} \frac{(i+\tilde{j})!^2(2n-i-\tilde{j})!^2}{i!^2 \tilde{j}!^2 (n-i)!^2 (n-\tilde{j})!^2} &= \binom{i+j}{i}^2 \binom{2n-i-j}{n-i}^2
\end{aligned}$$

The method of taking limits again fails for the summand of (2.22), and we were incapable of finding a general method that shows that a recurrence for the factorial term also holds for this summand. It is interesting that the recurrence we found via Sister Celine's technique holds for the summand of (2.22), as we have shown by plugging in the critical values. It is a open question, if there exists some other general method that is able to establish this.

As a by-product note that we are able to give a different proof of Theorem 2.8 by simply taking a limit whenever one of the rational functions $R_{t,i,\mathbf{j}}$ is not defined.

2.7.3 Introducing a New Variable

We now show that with a simple trick — introducing a new variable into the term and then taking a limit — we can always overcome the problem of singularities.

It is possible to get a recurrence for the function $\binom{n}{n}$ that holds for all values $n \in \mathbb{C}$ in the following way. First find a recurrence for the more general function $\binom{n+\epsilon}{n}$: the recurrence we get is $nF(n) = (n+\epsilon)F(n-1)$. This recurrence holds for all values of $n \in \mathbb{C}$ and for all nonzero and sufficiently small $\epsilon \in \mathbb{C}$. If we let $\epsilon \to 0$ we get for every $n \in \mathbb{C}$ the recurrence $nF(n) = nF(n-1)$ for $\binom{n}{n}$.

As another example take $\frac{(i+j+\epsilon)!}{i!j!}$. Every recurrence for this function yields in the limit $\epsilon \to 0$ a recurrence for the function

$$F(i,j) = \begin{cases} 0 & \text{if } i \text{ or } j \text{ is a negative integer} \\ \binom{i+j}{i} & \text{otherwise.} \end{cases}$$

In this way we compute a recurrence that holds for the summand $F(n,i,j)$ of (2.22). First find a recurrence for

$$\tilde{F}(n,i,j,\epsilon) = \frac{(i+j+\epsilon)!^2(2n-i-j+\epsilon)!^2}{i!^2 j!^2 (n-i)!^2 (n-j)!^2}.$$

Since this recurrence holds for all integer values of n,i,j if ϵ is not an integer, and since $\lim_{\epsilon\to 0} \tilde{F}(n,i,j,\epsilon) = F(n,i,j)$, we get a recurrence for $F(n,i,j)$.

This method is general enough to find recurrences for every binomial summand. Unfortunately a new variable usually increases the computation time a lot, so this method should be used with care. However, it is interesting to note that with a modification of this method we are able to transform sums with certain nonstandard boundary conditions into sums with standard boundary conditions (see Subsection 3.4).

2.8 Summary

We give a summary of the main results of this chapter.

- In Section 2.1 we defined Sister Celine's technique, which is a method to compute a k-free recurrence

$$\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n)F(n-i,\mathbf{k}-\mathbf{j}) = 0 \tag{2.23}$$

  for a hypergeometric function F.

- In Section 2.2 we gave some basic definitions: proper hypergeometric functions (F_t), structuresets (S), and k-free recurrences.

- In Section 2.3 we investigated the central step of Sister Celine's technique: the transformation of the Ansatz (2.23) into a polynomial equation $\sum a_{i,\mathbf{j}}(n)p_{i,\mathbf{j}}(n,\mathbf{k})$, the so-called associated polynomial. Using the concept of boundary points, we were able to give an explicit formula for the total degree in $\mathbf{k}$ of the associated polynomial.

- In Section 2.4 we proved that every proper hypergeometric function satisfies a k-free recurrence relation.

- In Section 2.5 we showed that P-maximal structuresets and Verbaeten completion improve the performance of Sister Celine's technique significantly.

- Section 2.6 is devoted to Verbaeten's existence theory for the special case of a single summation variable.

- In Section 2.7 we investigated the relationship between proper hypergeometric functions and binomial functions.

Chapter 3

Sister Celine's Technique: Summation, Certification, and Generalizations

3.1 Introduction

At the beginning of Chapter 2 we already gave an example of how to compute a recurrence for a multiple sum $\sum_{\mathbf{k}} F(n,\mathbf{k})$ from a k-free recurrence for the summand $F(n,\mathbf{k})$: by summing over the k-free recurrence. In this chapter we will investigate this in more detail. In particular, we will show that we can always transform a k-free recurrence into a *certificate recurrence*, i.e., a recurrence (in operator notation) of the form

$$S(n,N) + \sum_{l=1}^{r} \Delta_{k_l} S_l \tag{3.1}$$

where the S_l themselves are polynomial recurrence operators. This certificate recurrence has the appropriate form for summation: it immediately yields a recurrence for the multiple sum. Moreover, the certificate recurrence *certifies* that the sum satisfies this recurrence, since we can easily verify that a hypergeometric function F is annihilated by (3.1). The certificate recurrence is therefore a computer generated proof.

Different from our approach that uses only polynomial recurrence operators, Wilf and Zeilberger ([WZ92a], [WZ92b]) use rational functions for certification, i.e., they present the recurrence operator for the summand in the form

$$S(n,N) + \sum_{l=1}^{r} \Delta_{k_l} \frac{p_l(n,\mathbf{k})}{q_l(n,\mathbf{k})}$$

where the p_l and q_l are polynomials (in this context interpreted as multiplication operators). This rational function approach is more general than the polynomial recurrence operator approach, but this is not necessarily an advantage. We do not only think that certificate recurrences are simpler (we do not have to bother about the singularities of the rational functions), but that they are also algorithmically favourable.

The main point is that contrary to the univariate case, there does not exist an effective algorithm to compute the rational function certificates. The algorithm presented in [WZ92a] does not solve the main problem — guessing the denominator polynomials q_l (see Subsection 3.5.2) — satisfactorily. But in Section 3.5 we describe a generalization of Sister Celine's technique, which automatically finds certificate recurrences, and this algorithm turns out to be very efficient.

From now on we use the operator notation introduced in Section 1.2 for recurrences. Let $F(n, \mathbf{k})$ be a function that satisfies a k-free recurrence relation

$$\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n)\, F(n-i, \mathbf{k}-\mathbf{j}) \;=\; 0.$$

As usual we denote by N, K_i the forward-shift operators in the variables n, respectively k_i. We can write the recurrence in operator form

$$\Big(\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n) N^{-i}\mathbf{K}^{-\mathbf{j}}\Big)\; F(n, \mathbf{k}) \;=\; 0. \tag{3.2}$$

To get rid of shift operators with negative exponents we multiply (3.2) with $N^{-I}\mathbf{K}^{-\mathbf{J}}$, where $I = \min_{(i,\mathbf{j})\in S} -i$ and $J_s = \min_{(i,\mathbf{j})\in S} -j_s$. Thus we have

$$P(n, N, \mathbf{K})\, F(n, \mathbf{k}) \;=\; 0.$$

where P is an element of $\mathbf{R}[n]\langle N, \mathbf{K}\rangle$ ($\mathbf{R}$ is the ring over which the polynomials $a_{i,\mathbf{j}}$ are defined — for proper hypergeometric functions this is $\mathbb{C}[\boldsymbol{\alpha}]$).

Note that in a large part of the next sections the given summand does not need to be (proper) hypergeometric any more. It usually suffices that $F(n, \mathbf{k})$ satisfies a k-free recurrence relation.

3.2 Certificate Recurrences

The recurrence (2.2) for the trinomial function from the beginning of Chapter 2 equals[1]

$$(x \,+\, yI \,+\, zIJ \,-\, NIJ)\, F(n, i, j) \;=\; 0.$$

Let us see how we can transform this recurrence into a certificate recurrence. First divide the recurrence operator by $(J-1)$ (the division is commutative since the recurrence is free of j) such that the recurrence operator equals

$$x + yI + zI - NI \;+\; (J-1)\,(zI - NI).$$

[1]Here I is the shift operator in the variable i, not the identity operator, which will be written as unit 1.

The remainder of this division is free of J and we divide it by $(I-1)$ and get the recurrence operator in certificate form:

$$x+y+z-N \;+\; (I-1)\,(y+z-N) \;+\; (J-1)\,(zI-NI). \tag{3.3}$$

This transformation is not unique: dividing first by $(I-1)$ and then by $(J-1)$ gives different delta parts. But the principal part, here $x+y+z-N$, is always the same.

This certificate recurrence has the appropriate form for summation: if we sum then the certificate parts telescope and, if the summation bounds are naturally induced, the principal part of the certificate recurrence yields a homogeneous recurrence relation for the sum (see Section 3.3). So the advantages of certificate recurrences are that they are as easily verifiable as k-free recurrences and, more important, that we can directly read off a recurrence for the sum from the principal part.

Definition 3.1. Let $n, \mathbf{k} = (k_1, \ldots, k_r)$ be variables, $N, \mathbf{K} = (K_1, \ldots, K_r)$ the forward-shift operators in these variables, and $\mathbf{R}$ a ring. A *certificate recurrence* operator over $\mathbf{R}$ in n with delta parts in $\mathbf{k}$ is an element of $\mathbf{R}[n,\mathbf{k}]\langle N, \mathbf{K}\rangle$ of the form

$$P(n,N) \;+\; \sum_{i=1}^{r} (K_i - 1)\, S_i(n, \mathbf{k}, N, \mathbf{K}),$$

where $S_i \in \mathbf{R}[n,\mathbf{k}]\langle N, \mathbf{K}\rangle$, and $P \in \mathbf{R}[n]\langle N\rangle$. We call P the *principal part* of the certificate recurrence. A certificate recurrence operator is *nontrivial* iff its principal part does not equal zero.

It is important to keep in mind that the delta parts may contain the summation variables. If we expand the delta parts in a certificate recurrence we get an ordinary (but not k-free) recurrence operator. Of course many properties of k-free recurrences, like the relationship between a recurrence for the $R_{t,i,\mathbf{j}}$ and a recurrence for F_t or the possibility of taking limits to prove that a recurrence holds for an extended function, are equally valid for certificate recurrences.

Let us now give a proof that such a certificate form can always be computed from a k-free recurrence. This is done as in the example above by dividing the recurrence operator first by (K_1-1), the remainder of this division by (K_2-1), and so on. The remainder of the last division is free of all K_l and forms the principal part. The problem is that a division by (K_l-1) can yield a trivial remainder. This problem was neglected in [WZ92a], but we can overcome this by a simple "noncommutative trick". The key is the behaviour of an operator of the form $(K-1)S$ by left-multiplication with k:

$$\begin{aligned} k(K-1)S \;&=\; kKS \;-\; kS \;=\; K(k-1)S \;-\; kS \;-\; (k-1)S \;+\; (k-1)S \\ &=\; (K-1)(k-1)S \;-\; S. \end{aligned}$$

This can be generalized and completes the proof of the main theorem of [WZ92a].

Theorem 3.2. *Let $F(n, \mathbf{k})$, $\mathbf{k} = (k_1, \ldots, k_r)$, be a function that is annihilated by a recurrence operator $P \in \mathbf{R}[n]\langle N, \mathbf{K}\rangle$, free of $\mathbf{k}$. Then F is also annihilated by a nontrivial certificate recurrence operator in n with delta parts in $\mathbf{k}$, and it can be constructed from P by successive divisions by $(K_l - 1)$, l from 1 to r.*

Proof. We show that for every $l \in [1 \,..\, r+1]$ we can find $S_j \in \mathbf{R}[n, \mathbf{k}]\langle N, K_j, \ldots, K_r\rangle$ for every $j \in [1 \,..\, l-1]$ and a nontrivial $R_l \in \mathbf{R}[n]\langle N, K_l, \ldots, K_r\rangle$ such that

$$P_l = \sum_{j=1}^{l-1} (K_j - 1)\, S_j(n, \mathbf{k}, N, K_j, \ldots, K_r) \;+\; R_l(n, N, K_l, \ldots, K_r). \tag{3.4}$$

annihilates F. The theorem follows from this since P_{r+1} is the desired certificate recurrence.

We prove the claim inductively: For $l = 1$ take $P_1 = R_1 = P$.

Assume that for some $l \in [1 \,..\, r]$ we have an recurrence operator P_l of the form (3.4) that annihilates F. We show that by division of R_l by $(K_l - 1)$ we can construct the recurrence operator P_{l+1}.

Let $S \in \mathbf{R}[n]\langle N, K_l, \ldots, K_r\rangle$ and $i \in \mathbb{N}_0$ be such that $R_l = (K_l - 1)^i\, S$ and S is not divisible by $(K_l - 1)$. If $i > 0$ then we multiply R_l with $k_l^{\underline{i}}$, the i-th falling factorial of k_l, from the left and get with the rules of noncommutative multiplication and the abbreviation $\bar{S}$ for $(K_l - 1)^{i-1}\, S$,

$$\begin{aligned}
k_l^{\underline{i}} R_l \;&=\; k_l^{\underline{i}} K_l \bar{S} \;-\; k_l^{\underline{i}} \bar{S} \;=\; K_l (k_l - 1)^{\underline{i}} \bar{S} \;-\; (k_l - 1)^{\underline{i}} \bar{S} \;+\; (k_l - 1)^{\underline{i}} \bar{S} \;-\; k_l^{\underline{i}} \bar{S} \\
&=\; (K_l - 1)(k_l - 1)^{\underline{i}} \bar{S} \;-\; (k_l^{\underline{i}} - (k_l - 1)^{\underline{i}}) \bar{S} \\
&=\; (K_l - 1)(k_l - 1)^{\underline{i}} \bar{S} \;-\; i (k_l - 1)^{\underline{i-1}} \bar{S}.
\end{aligned}$$

The last term can be reduced according to the same scheme, and finally we get, including the case $i = 0$,

$$k_l^{\underline{i}} R_l \;=\; (K_l - 1)\, \Big(\sum_{m=1}^{i} (-1)^{m-1} i^{\underline{m-1}} (k_l - m)^{\underline{i-(m-1)}} (K_l - 1)^{i-m} S \Big) \;+\; (-1)^i i!\, S.$$

Since S is not divisible by $(K_l - 1)$ we can write $(-1)^i i!\, S \;=\; (K_l - 1)\, \bar{S}_l \;+\; R_{l+1}$, where $\bar{S}_l \in \mathbf{R}[n]\langle N, K_l, \ldots, K_r\rangle$ and $R_{l+1} \in \mathbf{R}[n]\langle N, K_{l+1}, \ldots, K_r\rangle$ is nontrivial.

Multiplying the whole recurrence P_l with $k_l^{\underline{i}}$ from the left we therefore get the recurrence

$$\begin{aligned}
&\sum_{j=1}^{l-1} (K_j - 1)\, k_l^{\underline{i}} S_j \\
&+\; (K_l - 1)\, \Big(\bar{S}_l \;+\; \sum_{m=1}^{i} (-1)^{m-1} i^{\underline{m-1}} (k_l - m)^{\underline{i-(m-1)}} (K_l - 1)^{i-m} S \Big) \\
&+\; R_{l+1}(n, N, K_{l+1}, \ldots, K_r)
\end{aligned}$$

that annihilates F. This is the recurrence operator P_{l+1} of the form (3.4). $\square$

Corollary 3.3. *(The fundamental theorem of hypergeometric summation, [WZ92a].) Every proper hypergeometric function $F_t(n, \mathbf{k})$ satisfies a nontrivial certificate recurrence in n with delta parts in $\mathbf{k}$.*

Note that for terms with only one summation variable k it is possible to settle the problem of a possibly vanishing remainder in a different way. In [GKP94] and [PWZ96] it is proved, assuming that F is hypergeometric (this is not required in the above proof), that in this case a k-free recurrence for F with lower order in K exists. A generalization of this argument to the multivariate case fails. Furthermore note that the case that a division yields a trivial remainder is a hypothetical one: this never happened in any example we considered.

Algorithmically the division of a recurrence by a factor $(K-1)$ is just a sequence of additions, since

$$\sum_{j=0}^{J} a_j K^j \;=\; (K-1)\Big(\sum_{j=0}^{J-1}\big(\sum_{i=j+1}^{J} a_i\big)K^j\Big) \;+\; \sum_{j=0}^{J} a_j.$$

Under the assumption that no remainder vanishes, the principal part of the certificate recurrence we get from the k-free recurrence $\sum a_{i,\mathbf{j}} N^i \mathbf{K}^{\mathbf{j}}$ equals $\sum_i (\sum_{\mathbf{j}} a_{i,\mathbf{j}}) N^i$. We see that in this case the principal part is independent of the order in which the divisions are performed.

It is also possible to consider k-free recurrence operators of the more general form $\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n) N^{-i} \mathbf{K}^{-\mathbf{j}}$ with an arbitrary structureset set S. Such a recurrence can analogously be transformed into a principal part plus delta parts, where the principal part is now of the form $\sum_i b_i(n) N^{-i} \mathbf{K}^{-\mathbf{J}(i)}$ with $J_l(i) = \max_{(i,\mathbf{j})\in S} j_l$.

Assuming that no remainder vanishes, the certificate recurrences computed from k-free recurrences have the property that the polynomial coefficients of the recurrence operators in the delta parts are again free of the summation variables. This is a superfluous property since only the principal part of a certificate recurrence has to be free of the summation variables. It is likely that an algorithm, which looks for certificate recurrences with delta parts containing the summation variables, finds much simpler recurrences. Indeed, although there is no direct existence theory for this kind of recurrences, such an algorithm turns out to be very efficient (see Section 3.5).

3.3 Summation with Standard Boundary Conditions

In this section we deal with summation of functions with standard boundary conditions. The summation range of such a sum is the support of the function and outside of this support there are enough values for which the function is defined. So we can sum the recurrence over a larger domain and get a homogeneous recurrence relation for the sum.

As an example, consider again the trinomial sum $f(n) = \sum_i \sum_j \binom{n}{j}\binom{j}{i} x^i y^{j-i} z^{n-j}$. We already know that the summand $F(n, i, j)$ is annihilated by the certificate recurrence operator (3.3). The fundamental property of certificate recurrences and of summation with standard boundary conditions is that if we sum the recurrence over a domain that is somewhat larger than the support of the function, then the delta parts telescope and the boundary values are not in the support and vanish. Let us apply the recurrence operator (3.3) to F and sum for i and j from $-\infty$ to ∞:

$$\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}\Big((x+y+z)F(n,i,j)-F(n+1,i,j)$$
$$+(I-1)\big((y+z)F(n,i,j)-F(n+1,i,j)\big)$$
$$+(J-1)\big(zF(n,i+1,j)-F(n+1,i+1,j)\big)\Big) \;=\; 0. \quad (3.5)$$

The $(I-1)$ part telescopes and since $\lim_{i\to\infty}F(n,i,j)=\lim_{i\to-\infty}F(n,i,j)=0$ for all $n\in\mathbb{N}_0$ and $j\in\mathbb{Z}$ we get

$$\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}(I-1)\big((y+z)F(n,i,j)-F(n+1,i,j)$$
$$=\sum_{j=-\infty}^{\infty}\big(\lim_{i\to\infty}((y+z)F(n,i,j)-F(n+1,i,j))-\lim_{i\to-\infty}((y+z)F(n,i,j)-F(n+1,i,j))\big) \;=\; 0,$$

and similarly the sum of the $(J-1)$ part vanishes. Thus (3.5) yields a homogeneous recurrence relation for the sum: $(x+y+z)f(n)-f(n+1)=0$.

We now define the functions that allow summation with standard boundary conditions: those are called the admissible functions. This important notion was introduced and investigated by Wilf and Zeilberger in [WZ92a] (although we here give a slightly more general definition). We will here and in the following assume that F is only a function of n and $\mathbf{k}$ and does not involve any additional parameter. It is easily seen that the following definitions and theorems can be generalized in a straightforward way to functions $F(n,\mathbf{k},\boldsymbol{\alpha})$.

Definition 3.4. Let $F(n,\mathbf{k})$ be a function defined on $D\subseteq\mathbb{Z}^{r+1}$, let $P(n,\mathbf{k},N,\mathbf{K})\in\mathbf{R}[n,\mathbf{k}]\langle N,\mathbf{K}\rangle$ be a recurrence operator, and let $I\in\mathbb{N}_0$ and $\mathbf{J}\in\mathbb{N}_0^r$ be the orders of P in n respectively $\mathbf{k}$.

- The set

$$\mathcal{N}_F := \{n\in\mathbb{Z}\mid \forall i\in[0\,..\,I]\ \exists\mathbf{k}\text{ such that }(n+i,\mathbf{k})\in D\}$$

is called the *range* of n. For every $n\in\mathcal{N}_F$ we define

$$\mathrm{Supp}_F(n)=\{\mathbf{k}\mid(n,\mathbf{k})\in D\text{ and }F(n,\mathbf{k})\neq 0\}$$
$$\mathrm{Summ}_{F,I,\mathbf{J}}(n)=\{\mathbf{k}\mid(n+i,\mathbf{k}+\mathbf{j})\in D\text{ for all }i\in[0\,..\,I]\text{ and }\mathbf{j}\in[\mathbf{0}\,..\,\mathbf{J}]\},$$

the *support*, respectively the *summation range* of F.

- The function F is called *summable* w.r.t. the recurrence P iff

– for all $n \in \mathcal{N}_F$ we have $n+1 \in \mathcal{N}_F$, and

– for all $n \in \mathcal{N}_F$ all the sums in

$$\sum_{\mathbf{k}\in \mathrm{Summ}_{F,I,\mathbf{J}}(n)} P(n,\mathbf{k},N,\mathbf{K})\, F(n,\mathbf{k})$$

exist.

- The function F is called *admissible* w.r.t. the recurrence P iff for all $n \in \mathcal{N}_F$ and for all $i \in [0\,..\,I]$ and $\mathbf{j} \in [\mathbf{0}\,..\,\mathbf{J}]$ we have

$$\mathrm{Supp}_F(n+i) - \mathbf{j} \subseteq \mathrm{Summ}_{F,I,\mathbf{J}}(n).$$

Note that the summation range $\mathrm{Summ}_{F,I,\mathbf{J}}(n)$ is defined such that we formally can form the sum $\sum_{\mathbf{k}\in \mathrm{Summ}_{F,I,\mathbf{J}}(n)} P(n,\mathbf{k},N,\mathbf{K})\, F(n,\mathbf{k})$, and the summability condition assures us that this sum actually exists. The summability condition is only necessary if the support of F is infinite. If F is summable w.r.t. a k-free recurrence P this merely means that for all $n \in \mathcal{N}_F$ the sum $\sum_{\mathbf{k}\in \mathrm{Supp}_F(n)} F(n,\mathbf{k})$ exists. The most important condition is admissibility: it guarantees us a large enough zone of zeros around the support of F.

It is easy to see that the most important case, i.e., an everywhere defined function with compact support, is summable and admissible w.r.t. to every recurrence. But the definition is more general, infinite convergent sums and infinite formal power series are included as well as the definition for standard summation given in [WZ92a].

Nearly all the interesting identities, which we prove in Chapter 5, involve sums with standard boundary conditions. For example, the summands of the sums

$$\sum_k \sum_j \binom{n}{k}\binom{n+k}{k}\binom{k}{j}^3, \quad \text{and} \quad \sum_{r=0}^{\infty} \frac{(-1)^r z^{n+2r}}{2^{n+2r}\, r!\, \Gamma(n+r+1)},$$

are admissible and summable functions and both sums range over the support of the summand. Below we prove that for such sums the principal part of a certificate recurrence operator annihilating the summand is a recurrence operator annihilating the sum.

Let $F(n,\mathbf{k})$ be annihilated by the k-free recurrence operator $\sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n) N^i \mathbf{K}^{\mathbf{j}}$ with orders I in n and $\mathbf{J}$ in $\mathbf{k}$. The definition of a summable and admissible function is exactly what is needed to compute a homogeneous recurrence relation for the sum

$$f(n) \;=\; \sum_{\mathbf{k}\in \mathrm{Supp}_F(n)} F(n,\mathbf{k}).$$

The summation range $\mathrm{Summ}_{F,I,\mathbf{J}}(n)$ is defined such that the recurrence can be summed, so if F is summable w.r.t. the recurrence we get

$$\sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n) \sum_{\mathbf{k}\in \mathrm{Summ}_{F,I,\mathbf{J}}(n)} F(n+i,\mathbf{k}+\mathbf{j}) \;=\; 0$$

and if the function is admissible w.r.t. the recurrence then we have

$$\sum_{\mathbf{k}\in \mathrm{Summ}_{F,I,\mathbf{J}}(n)} F(n+i,\mathbf{k}+\mathbf{j}) \;=\; \sum_{\mathbf{k}\in \mathrm{Supp}_F(n+i)-\mathbf{j}} F(n+i,\mathbf{k}+\mathbf{j}) \;=\; f(n+i)$$

and we get the following recurrence relation for $f(n)$:

$$\sum_{i=0}^{I}\Big(\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n)\Big)\, f(n+i) \;=\; 0.$$

The problem is that the coefficients of this recurrence, i.e., the sums of the polynomials, may all be zero, so that we get no recurrence at all. But using certificate recurrences instead, we always get a nontrivial recurrence for the sum.

Theorem 3.5. *Let the function $F(n,\mathbf{k})$, defined on $D \subseteq \mathbb{Z}^{r+1}$, be annihilated by the nontrivial certificate recurrence operator $P = S_0(n,N) + \sum_{i=1}^{r}(K_i - 1)S_i \in \mathbf{R}[n,\mathbf{k}]\langle N,\mathbf{K}\rangle$. Let F be admissible and summable w.r.t. the recurrence P (expanded to normal form). Then the sum*

$$f(n) = \sum_{\mathbf{k}\in Supp(n)} F(n,\mathbf{k}),$$

defined on $\mathcal{N}_F$, is annihilated by $S_0(n,N) \in \mathbf{R}[n]\langle N\rangle$.

Proof. If we expand the delta operators the certificate recurrence can be written in normal form as

$$P \;=\; \sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}) N^i \mathbf{K}^{\mathbf{j}}$$

with certain polynomials $a_{i,\mathbf{j}}(n,\mathbf{k}) \in \mathbf{R}[n,\mathbf{k}]$. It is easily seen (cf. Theorem 3.7) that the principal part of the certificate recurrence equals

$$S_0(n,N) \;=\; \sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j}) N^i.$$

Therefore all the polynomials $\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j})$ are free of $\mathbf{k}$ and at least one of them is nonzero. Thus we can, as above, sum the whole recurrence relation over the summation range and interchange the polynomial coefficients and the summation sign as follows

$$\sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} \sum_{\mathbf{k}\in \mathrm{Summ}_{F,I,\mathbf{J}}(n)} a_{i,\mathbf{j}}(n,\mathbf{k})\, F(n+i,\mathbf{k}+\mathbf{j}) \;=$$
$$\sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} \sum_{\mathbf{k}\in \mathrm{Supp}(n+i)} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j})\, F(n+i,\mathbf{k}) \;=$$
$$\sum_{i=0}^{I}\Big(\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j})\Big) \sum_{\mathbf{k}\in \mathrm{Supp}(n+i)} F(n+i,\mathbf{k}) \;=\; 0.$$

Therefore $f(n)$ is annihilated by $S_0(n,N)$. ☐

3.4 The Problems with Nonstandard Boundary Conditions

Unfortunately there are multiple sums that do not fit into the above standard summation framework. This means that we either do not sum over the support of the summand or that the summand is not defined for sufficiently many values outside of the support. We can still sum the recurrence but we will usually get an inhomogeneous recurrence relation for the sum. Telescoping eliminates only one summation sign, so the inhomogeneous parts of the recurrence will themselves be sums. Two simple examples for nonstandard summation that have closed form evaluations are (see Sections 5.5 and 5.6)

$$\sum_i \sum_{j=0}^{n} \binom{i+j}{i}^2 \binom{4n-2i-2j}{2n-2i} \;=\; (2n+1)\binom{2n}{n}^2 \tag{3.6}$$

$$\sum_{k_1} \sum_{k_2 \le k_1} (k_1-k_2)\binom{n}{k_1}\binom{n}{k_2} \;=\; n\,4^{n-1}\frac{(\frac{3}{2})_{n-1}}{(2)_{n-1}}. \tag{3.7}$$

Computing a recurrence for a sum with nonstandard boundary conditions can be a tedious task, as can be seen from the following example.

Suppose the summand of (3.6), denoted by $F(n,i,j)$, is annihilated by the certificate recurrence operator $\sum_p a_p(n)N^p + (I-1)S_1 + (J-1)\sum_{p,q} b_{p,q}(n,i,j)N^pJ^q$. We apply the operator to F and first sum over all integers i and the $(I-1)$ part vanishes. Then we sum w.r.t. j from 0 to n and get the following inhomogeneous recurrence relation for the sum $f(n)$:

$$\begin{aligned}&\sum_p a_p(n)f(n+p) \;-\; \sum_p a_p(n)\sum_i \sum_{j=n+1}^{n+p} F(n+p,i,j) \;+\\ &\sum_i\sum_{p,q} b_{p,q}(n,i,n+1)F(n+p,i+q,n+1) \;-\; \sum_i\sum_{p,q} b_{p,q}(n,i,0)F(n+p,i+q,0) \;=\; 0.\end{aligned} \tag{3.8}$$

The last three sums are the boundary values and have to be simplified, e.g., by computing a closed form or by finding a recurrence relation annihilating them all. This is usually a tedious and nontrivial task. Because of these difficulties we will compute sums with nonstandard boundary conditions only if the recurrences are simple.

There is an alternative approach to sums with nonstandard boundary conditions. We can use the method already mentionend in Subsection 2.7.3: introducing a new variable. With this method we are able to find homogeneous recurrence relations for sums with certain summation bounds automatically. The disadvantage is that the recurrences are rather huge.

Let t be a proper hypergeometric term in the variables n and $\mathbf{k}$, and let ϵ be a variable not occurring in t. Suppose that we are given a summation range $S(n)$ that is defined as a convex set of the form

$$S(n) \;=\; \{\mathbf{k} \in \mathbb{Z}^r \mid f_i n + \mathbf{g}_i \cdot \mathbf{k} + h_i \ge 0 \text{ for } i = [1\,..\,M]\} \tag{3.9}$$

where $M \in \mathbb{N}$, $f_i \in \mathbb{Z}$, $\mathbf{g}_i \in \mathbb{Z}^r$, and $h_i \in \mathbb{C}$. Suppose also that F_t is well-defined in $S(n)$. We want to compute a homogeneous recurrence relation for the sum

$$f(n) \;=\; \sum_{\mathbf{k}\in S(n)} F(n,\mathbf{k}).$$

Let $\tilde{t}$ be the proper hypergeometric term that is obtained if we replace in t every numerator factorial expression $(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p)$ that is zero or a negative integer for some $(n,\mathbf{k}) \in \mathbb{Z}^{r+1}$ by $(a_p n + \mathbf{b}_p \cdot \mathbf{k} + c_p + \epsilon)$. Thus $F_{\tilde{t}}$ is well-defined for all integer values of n and $\mathbf{k}$ as long as ϵ is nonzero and small. Now define the term s as

$$s \;=\; \tilde{t} \prod_{i=1}^{M} \frac{\Gamma(f_i n + \mathbf{g}_i \cdot \mathbf{k} + h_i + \epsilon + 1)}{\Gamma(f_i n + \mathbf{g}_i \cdot \mathbf{k} + h_i + 1)}.$$

The proper hypergeometric function F_s of s is well-defined for every integer tuple $(n,\mathbf{k}) \in \mathbb{Z}^{r+1}$ for sufficiently small nonzero ϵ. We define the function $F(n,\mathbf{k})$ for $(n,\mathbf{k}) \in \mathbb{Z}^{r+1}$ as $\lim_{\epsilon\to 0} F_s(n,\mathbf{k})$, and get

$$F(n,\mathbf{k}) \;=\; \begin{cases} F_t(n,\mathbf{k}) & \text{if } \mathbf{k} \in S(n) \\ 0 & \text{if } \mathbf{k} \notin S(n). \end{cases}$$

The sum of F over $S(n)$ now has standard boundary conditions, thus a recurrence relation for F immediately gives us a homogeneous recurrence relation for $f(n)$. A recurrence for F can be found by setting ϵ to zero in a recurrence for F_s. Note that the recurrence is nontrivial (if ϵ divides every term of the recurrence then it can be cancelled), but it might happen that the principal part of a certificate recurrence vanishes (although this was never observed). However, using k-free recurrences, we can always find a homogeneous recurrence relation for a nonstandard sum of this type.

We have just proved

Theorem 3.6. *Let $F(n,\mathbf{k})$ be a proper hypergeometric function, and let $S(n)$ be a summation range as defined in* (3.9)*. Then the sum $\sum_{\mathbf{k}\in S(n)} F(n,\mathbf{k})$ satisfies a homogeneous polynomial recurrence relation.*

In order to get a homogeneous recurrence relation for (3.7) we only have to find a recurrence for

$$(k_1 - k_2)\frac{n!^2 (k_1 - k_2 + \epsilon)!}{k_1!(n-k_1)!k_2!(n-k_2)!(k_1-k_2)!}.$$

It is not always necessary to multiply the term with new factors. Sometimes it is sufficient to introduce the ϵ into an already existing factorial expression, e.g., the term

$$\frac{(i+j+\epsilon)!^2 (4n-2i-2j+\epsilon)!}{i!^2 j!^2 (2n-2i)!(2n-2j)!}$$

transforms (3.6) into a sum with standard boundary conditions.

3.5 Generalizations of Sister Celine's Technique

Even with P-maximal structuresets, the ability of Sister Celine's technique to find recurrences for multiple sums is poor. The k-free recurrences are usually large, and we need a lot of time to find them. Therefore there is a need for faster algorithms. In this section we introduce two algorithms that are generalizations of Sister Celine's technique. It was already mentioned that the certificate recurrences computed with Sister Celine's technique have the superfluous property that the recurrence operators in the delta parts are (nearly always) free of the summation variables. The main idea for generalizations is that we look for certificate recurrences with the property that only the principal part is free of the summation variables.

First we show that under certain conditions every polynomial recurrence — and not only k-free recurrences — can be transformed into a certificate recurrence. The fact that nearly all these conditions correspond to systems of linear equations enables us to state the first generalization. Then we show that a part of the transformation into a certificate recurrence can be done a priori. This is our second generalization. Both algorithms use P-maximal structuresets, and it turns out that this is essential for their performance.

Unfortunately for none of the two generalizations we have an existence theory. The criterion we used for k-free recurrences — the size of the equation system — fails. A nontrivial certificate recurrence by definition has a nontrivial principal part, but not every solution of the linear equation systems (the central part of both algorithms is again the computation of the nullspace of a linear equation system) yields such a nontrivial principal part. This implies that there is no guarantee that any simpler certificate recurrence can be found. Hence our algorithms have to prove their value empirically: our implementations were able to solve dozens of examples (e.g., most identities that are proved in Chapter 5) that are beyond the computational power of Sister Celine's technique.

To get an impression for the observation that certificate recurrences whose delta parts contain the summation variables can be considerably simpler, compare the k-free recurrence given in Section 2.5 with the following, much simpler, "non-k-free" certificate recurrence operator annihilating $(-1)^k\binom{2n}{k}^3$:

$$\begin{aligned}
&6(2n+1)^2(3n+1)(3n+2) \ + \ 2(n+1)^2(2n+1)^2N \ - \\
&\Delta_k \ (-14+9k-24k^2-132n+135kn-63k^2n-518n^2+306kn^2-42k^2n^2-816n^3+ \\
&\ 192kn^3-440n^4 \ + \ (35+57k+24k^2+114n+165kn+63k^2n+122n^2+138kn^2+ \\
&\ 42k^2n^2+48n^3+24kn^3+8n^4) \ K-2(1+n)^2(1+2n)^2N-2(1+n)^2(1+2n)^2KN).
\end{aligned}$$

Note that the above recurrence yields a first order recurrence for Dixon's sum $f(n) = \sum_k(-1)^k\binom{2n}{k}^3$:

$$f(n+1) \ = \ -3\frac{(3n+1)(3n+2)}{(n+1)^2}f(n).$$

For nearly all double sum examples we considered, we found "non-k-free" certificate recurrences that are far simpler than the "k-free" certificate recurrences (if we were able to compute them at all). We give two simple double sum examples to illustrate this. The summand of the double sum in

$$\sum_i \sum_j \binom{r}{i}\binom{s}{j}\binom{t}{n-i-j} = \binom{r+s+t}{n}, \tag{3.10}$$

a generalization of the Vandermonde identity, is annihilated by the k-free certificate recurrence operator (the simplest we have found)

$$\begin{aligned}&(n-r-s-t) + (3n-2r-2s-2t+3)\,N + (3n-r-s-t+6)\,N^2 + (n+3)\,N^3 +\\ &\Delta_i((n-s-t+1)N + (n-s+2)N^2 + (n-t+2)N^2J + (n+3)N^3J) +\\ &\Delta_j((n-r-t+1)N + (2n-r-t+4)N^2 + (n+3)N^3),\end{aligned}$$

and also by the non-k-free certificate recurrence operator

$$(n-r-s-t) + (n+1)N + \Delta_i(iN) + \Delta_j(jN). \tag{3.11}$$

The summand of the double sum in

$$\sum_i \sum_j (-1)^{i+j}\binom{i+j}{i}\binom{n}{i}\binom{n}{j} = 1 \tag{3.12}$$

(a special case of an orthogonality relation, see Subsection 4.1.2) is annihilated by the k-free recurrence operator

$$\begin{aligned}&(n+1) - (2\,n+3)\,N + (n+2)\,N^2 +\\ &\Delta_i((n+1) - (n+1)\,J^2 + (n+1)\,I - 2\,(n+1)\,IJ + (n+1)\,IJ^2 - (2\,n+3)\,NJ +\\ &\ (2\,n+3)\,NIJ - (2\,n+3)\,NIJ^2 + (n+2)\,N^2J^2(n+2)\,N^2IJ^2) +\\ &\Delta_j(-(2\,n+3)\,N + (n+2)\,N^2 + (n+2)\,N^2J),\end{aligned}$$

and also by the non-k-free operator

$$\begin{aligned}&(n+1) - (n+1)\,N +\\ &\Delta_i(-(n+2\,i+1) + (1+2\,i+n)\,J - (n+1)\,NJ) +\\ &\Delta_j(2\,(n-j) - (n+1)\,N).\end{aligned} \tag{3.13}$$

These simple non-k-free certificate recurrences were computed by the Mathematica implementations of the generalizations of Sister Celine's technique (the functions `FindRecurrence` and `FindCertificate`, see Chapter 4), which are described below. Note that in the above non-k-free certificate recurrences the degree of the polynomials in the summation variables is quite small, at most 1 or 2.

Let us remark here that we have implemented various other generalizations, among them an algorithm similar to the approach of Wilf and Zeilberger ([WZ92a]) with rational functions as certificates, and a naive direct certificate recurrence finder (without P-maximal structuresets). But all these programs turned out to be much slower and much more cumbersome to use.

3.5.1 Recurrences Containing the Summation Variables

Let

$$\sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k})N^i\mathbf{K}^{\mathbf{j}} \tag{3.14}$$

be a homogeneous polynomial recurrence operator, where the polynomial coefficient may depend on $\mathbf{k}$. Let us perform the transformation process that leads to a certificate recurrence on it. We divide (3.14) by (K_1-1), the remainder of the division by (K_2-1), and so on. Assuming that every remainder is nonzero, we are able to give the remainder of the last division explicitly. It is easy to see that, in the case of a single summation variable k, we have

$$\begin{aligned}\sum_{i=0}^{I}\sum_{j=0}^{J} a_{i,j}(n,k)N^iK^j \;=\; & \\ (K-1)\Big(\sum_{i=0}^{I}\sum_{j=0}^{J-1}\Big(\sum_{l=j+1}^{J} a_{i,l}(n,k+j-l)\Big)N^iK^j\Big) \;+\; & \sum_{i=0}^{I}\Big(\sum_{j=0}^{J} a_{i,j}(n,k-j)\Big)N^i.\end{aligned}$$

In the multivariate case therefore the very last remainder equals

$$\sum_{i=0}^{I}\Big(\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j})\Big)N^i. \tag{3.15}$$

The recurrence (3.14) can be written as a nontrivial certificate recurrence in n with delta parts in $\mathbf{k}$, if the last remainder (3.15) is nontrivial and does not depend on the variables $\mathbf{k}$. This proves the following theorem.

Theorem 3.7. *Let $P=\sum_{i=0}^{I}\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k})N^i\mathbf{K}^{\mathbf{j}} \in \mathbf{R}[n,\mathbf{k}]\langle N,\mathbf{K}\rangle$, for some ring $\mathbf{R}$, be a nontrivial recurrence operator. If for all $i\in[0\,..\,I]$*

$$\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j}) \in \mathbf{R}[n] \tag{3.16}$$

and if there is an $i\in[0\,..\,I]$ such that

$$\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j}) \neq 0, \tag{3.17}$$

then P can be written as a nontrivial certificate recurrence in n with delta parts in $\mathbf{k}$, i.e.,

$$P \;=\; \sum_{i=0}^{I}\Big(\sum_{\mathbf{j}=\mathbf{0}}^{\mathbf{J}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j})\Big)N^i \;+\; \sum_{l=1}^{r}\Delta_{k_l}S_l$$

for some $S_l\in\mathbf{R}[n,\mathbf{k}]\langle N,\mathbf{K}\rangle$.

Consider again the identity (3.12). It is easily seen that the summand of its double sum is annihilated by

$$2(i+j+1) + (n-2i-2j-3)J - (n+2i+3)I + (n+2i+3)IJ - (n+1)NIJ. \quad (3.18)$$

The conditions of Theorem 3.7 are fulfilled and, indeed, this recurrence can be written as the certificate recurrence (3.13).

We are able to give an algorithm to find non-k-free recurrences that can be written as a certificate recurrence with k-free principal part; informally it is described as follows.

First generalization of Sister Celine's technique.

- The input consists of a proper hypergeometric term t in the hypergeometric variables n and $\mathbf{k} = (k_1, \ldots, k_r)$ and the additional parameters $\boldsymbol{\alpha}$, and a P-maximal structureset S for t. For the recurrence operator we make an Ansatz of the form

$$\sum_{(i,\mathbf{j})\in S} \Big(\sum_{\mathbf{l}=\mathbf{0}}^{\mathbf{M}_{i,\mathbf{j}}} a_{i,\mathbf{j},\mathbf{l}}(n)\mathbf{k}^{\mathbf{l}} \Big) \; N^{-i}\mathbf{K}^{-\mathbf{j}} \quad (3.19)$$

  where the $\mathbf{M}_{i,\mathbf{j}} \in \mathbb{N}_0^r$ are degree bounds for the generic polynomials. These degree bounds have to be given as input, too, since we have no theory that tells us how we should choose these bounds. We usually set all the degree bounds to a constant.

  Note that due to a notational conflict — N and $\mathbf{K}$ denote forward-shift operators but our recurrences involve backward shifted terms $F(n-i, \mathbf{k}-\mathbf{j})$ — the shift operators in the Ansatz have negative exponents.

- We have to determine the unknowns $a_{i,\mathbf{j},\mathbf{l}}(n)$ as polynomials in n and $\boldsymbol{\alpha}$ such that the recurrence operator can be written as a certificate recurrence with a principal part free of $\mathbf{k}$. Condition (3.16) of Theorem 3.7 for the Ansatz turns into the condition that

$$\sum_{\mathbf{j}\in S(i)} \sum_{\mathbf{l}=\mathbf{0}}^{\mathbf{M}_{i,\mathbf{j}}} a_{i,\mathbf{j},\mathbf{l}}(n)(\mathbf{k}+\mathbf{j})^{\mathbf{l}} \quad \in \mathbb{C}[n, \boldsymbol{\alpha}], \quad (3.20)$$

  for every i, where $S(i) = \{\mathbf{j} \mid (i,\mathbf{j}) \in S\}$. By expanding these polynomials and comparing the coefficient of every nontrivial monomial in $\mathbf{k}$ with zero, we get for every i a homogeneous linear equation system (over the integers) for the $a_{i,\mathbf{j},\mathbf{l}}$: the *reduction systems*. Solving the reduction systems (they always have nontrivial solutions) we get that some of the $a_{i,\mathbf{j},\mathbf{l}}$ (the *reducible* unknowns) can be expressed as linear combinations of the remaining unknowns (which are linearly independent in the space of solutions of the reductions systems).

- We replace the reducible $a_{i,\mathbf{j},\mathbf{l}}$ in the Ansatz by these linear combinations and get the *reduced Ansatz* involving a lower number of unknowns. The solutions of this reduced Ansatz yield certificate recurrences with a $\mathbf{k}$-free principal part, but, since condition (3.17) can not be used, it is possible that these certificate recurrences are trivial.

- The reduced Ansatz is solved for the remaining unknowns in the usual way: apply the operator to $F_t(n, \mathbf{k})$ and divide by $F_t(n, \mathbf{k})$ to get a rational equation for the unknowns; multiply with a common denominator, compare the coefficients of every monomial in $\mathbf{k}$ with zero to get a homogeneous linear equation system for the remaining unknowns. It remains to write the solutions of this equation system as certificate recurrences, cancelling the trivial ones.

As a remark let us state an open question: is it possible to replace the condition (3.17) of Theorem 3.7 by an "algorithmically simpler" condition, i.e., a condition that can be used in the Ansatz before setting up and solving the equation system? If so then it should be possible to develop an existence theory for non-k-free certificate recurrences that is not based on the existence theorem for k-free recurrences.

3.5.2 A Direct Attack on Certificate Recurrences

In this subsection we investigate the possibility of looking *directly* for certificate recurrences. We consider several approaches, but surprisingly none of them turns out to be better than the first generalization of Sister Celine's technique (Subsection 3.5.1); only one approach — the one that allows us to use P-maximal structuresets — is equally successful. So in this subsection we state a negative result: we tried to find a significantly faster algorithm than the first generalization and failed.

First let us investigate the method of Wilf and Zeilberger [WZ92a], i.e., we look for a recurrence operator of the form

$$\sum_{i=0}^{I} a_i(n)N^i + \sum_{l=1}^{r} \Delta_{k_l} \frac{p_l(n,\mathbf{k})}{q_l(n,\mathbf{k})}.$$

The difficulty is that in order to get a linear equation system we have to know (in addition to the knowledge of a degree bound in $\mathbf{k}$ for p_l) the denominator polynomials q_l. For the case of one summation variable it is possible (with Gosper's algorithm) to get this denominator polynomial. For several summation variables this is a difficult problem: if we add two arbitrary rational functions, the cancellation of denominator factors can be arbitrarily complicated. This problem was not solved in [WZ92a] in a satisfying way — Zeilberger's implementation[2] of this algorithm requests those denominators as input. Of course there are certain promising ways of guessing such denominator factors, e.g., one could use the product of denominators of the rational functions $F(n-i, \mathbf{k}-\mathbf{j})/F(n, \mathbf{k})$ for $(i, \mathbf{j})$ from a certain structureset S. But none of these approaches turned out to be successful. Although we tried hard to write a fast implementation of the Wilf-Zeilberger method, the programs were difficult to use, slow, and yielded results of unmanageable size.

[2]available from Zeilberger's homepage: `http://www.math.temple.edu/~zeilberg`

The next idea was to use recurrence operators in the delta parts, i.e., to try an Ansatz of the form

$$\sum_{i=0}^{I} a_i(n)N^i \;+\; \sum_{l=1}^{r} \Delta_{k_l} \sum_{(i,\mathbf{j})\in S_l} b_{l,i,\mathbf{j}}(n,\mathbf{k})\, N^i\mathbf{K}^{\mathbf{j}} \tag{3.21}$$

for the recurrence operator. How shall we choose the structuresets S_l? It is quite reasonable to take the sets that we get when we transform a recurrence $\sum_{i=0}^{I}\sum_{\mathbf{j}=0}^{\mathbf{J}} a_{i,\mathbf{j}}N^i\mathbf{K}^{\mathbf{j}}$. with a rectangular structureset $S_{I,\mathbf{J}}$ into a certificate recurrence. It is easy to see that we have to take $S_l = \{(i,0,\ldots,0,j_l,\ldots,j_r) \in S_{I,\mathbf{J}} \mid (i,0,\ldots,0,j_l+1,\ldots,j_r) \in S_{I,\mathbf{J}}\}$ for this purpose. Although this method is easier to use (and usually faster) than the Wilf-Zeilberger approach, it is still not not as successful as the generalization of Sister Celine's technique given above. The reason for this is that we did not use the successful tool for k-free recurrences, the P-maximal structureset. But it is not possible to use P-maximal structuresets for the Ansatz (3.21) since the transformation of a recurrence into a certificate recurrence inevitably destroys the shape of the structureset. The structureset of the certificate recurrence is larger, but the new points in it do not yield any additional information.

Instead we have to use an Ansatz of *pre-certificate* form, i.e., a recurrence operator that is nearly a certificate recurrence and allows us to use P-maximal structuresets. Such an Ansatz has the general form

$$\sum_{(i,\mathbf{j})\in S_0} a_{i,\mathbf{j}}(n,\mathbf{k})N^i\mathbf{K}^{\mathbf{j}} \;+\; \sum_{l=1}^{r} \Delta_{k_l} \sum_{(i,\mathbf{j})\in S_l} b_{l,i,\mathbf{j}}(n,\mathbf{k})\, N^i\mathbf{K}^{\mathbf{j}} \tag{3.22}$$

with the additional condition that $\sum_{\mathbf{j}} a_{i,\mathbf{j}}(n,\mathbf{k}-\mathbf{j})$ is free of $\mathbf{k}$ (cf. Theorem 3.7). Given an arbitrary structureset S we have to determine the sets S_i for $i = 0,1,\ldots,r$ such that every recurrence with structureset S can be written in the form (3.22), such that the structureset of the pre-certificate recurrence (i.e., the structureset of the recurrence obtained by expanding the delta operators in the generic form (3.22)) is not larger than S, and such that the S_i are as small as possible. Especially the last condition is vital for the performance of the method and is difficult to achieve due to an unpleasant property of (pre-)certificate recurrences: a recurrence can be written as a (pre-)certificate recurrence in many different ways.

To find the S_i we have to find out how much of the transformation into a certificate recurrence can be done without enlarging the structureset. Let us investigate the transformation of an arbitrary, not necessarily k-free, recurrence relation $\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n,\mathbf{k})F(n-i,\mathbf{k}-\mathbf{j})$ with an arbitrary (possibly P-maximal) structureset S into a certificate recurrence. The operator representation of this recurrence is

$$\sum_{(i,\mathbf{j})\in S} a_{i,\mathbf{j}}(n,\mathbf{k})N^{-i}\mathbf{K}^{-\mathbf{j}}. \tag{3.23}$$

For the sake of simplicity we first take only one summation variable, i.e., the recurrence equals $\sum_{(i,j)\in S} a_{i,j}(n,k)N^{-i}K^{-j}$. Instead of dividing the recurrence by $(K-1)$, we now regard the transformation into a certificate recurrence as a reduction problem. It is easily seen that every monomial $a_{i,j}(n,k)N^{-i}K^{-j}$ can — without enlarging the structureset — be written as

$$a_{i,j}(n,k-1)N^{-i}K^{-j-1} + (K-1)a_{i,j}(n,k-1)N^{-i}K^{-j-1},$$

if $(i,j+1) \in S$. The monomial $a_{i,j}(n,k-1)N^{-i}K^{-j-1}$ can, if $(i,j+2) \in S$, be further reduced and so on until a reduction is no longer possible, that is if eventually $(i,j+l) \notin S$ for some $l \in \mathbb{N}$. If we reduce all the monomials of the recurrence in this way completely, we have (with certain polynomials $b_{i,j}$ and $c_{i,j}$)

$$\sum_{(i,j)\in S} a_{i,j}N^{-i}K^{-j} = \sum_{(i,j)\in S_0} b_{i,j}N^{-i}K^{-j} + (K-1)\sum_{(i,j)\in S_1} c_{i,j}N^{-i}K^{-j-1}, \tag{3.24}$$

where $S_0 = \{(i,j) \in S \mid (i,j+1) \notin S\}$ and $S_1 = \{(i,j) \in S \mid (i,j+1) \in S\}$. We see that the structureset of the r.h.s. of (3.24) is again S. If S is a P-maximal structureset then the $(K-1)$-free part of (3.24) equals $\sum_i b_{i,\max(i)}N^{-i}K^{-\max(i)}$ where $\max(i)$ is the largest j such that $(i,j) \in S$.

This reduction can applied to recurrences with more summation variables than one, but in order to make the result unique we have to specify the order in which the reductions are performed. For this purpose we define the reducibility of an element of a structureset.

Definition 3.8. Let $S \subset \mathbb{Z}^{r+1}$ be a structureset, and let $p \in [1..r]$. An element $(i,j_1,\ldots,j_p,\ldots,j_r) \in S$ is said to be *reducible to* $(i,j_1,\ldots,j_p+1,\ldots,j_r)$ w.r.t. to S and p, iff $(i,j_1,\ldots,j_p+1,\ldots,j_r) \in S$ and there is no $q \in [1..r]$ with $q < p$ and $(i,j_1,\ldots,j_q+1,\ldots,j_p,\ldots,j_r) \in S$. If an element is not reducible w.r.t. S and p for all $p \in [1..r]$, then it is called *irreducible* w.r.t. S.

Let $(i,j_1,\ldots,j_r) \in S$ be reducible to $(i,j_1,\ldots,j_p+1,\ldots,j_r)$ w.r.t. to S and p. We can also reduce the corresponding monomial of recurrence (3.23), i.e., we can write

$$\begin{aligned} a_{i,\mathbf{j}}(n,\mathbf{k})N^{-i}K_1^{-j_1}\cdots K_r^{-j_r} = \\ & a_{i,\mathbf{j}}(n,k_1,\ldots,k_p-1,\ldots,k_r)N^{-i}K_1^{-j_1}\cdots K_p^{-j_s-1}\cdots K_r^{-j_r} + \\ & (K_p-1)(a_{i,\mathbf{j}}(n,k_1,\ldots,k_p-1,\ldots,k_r)N^{-i}K_1^{-j_1}\cdots K_p^{-j_s-1}\cdots K_r^{-j_r}) \end{aligned}$$

without enlarging the structureset. Note that, due to our definition of reducibility, this monomial (more exactly, the corresponding element of the structureset) can be reduced in one and only one way: we cannot reduce it w.r.t. $q < p$ and we may not reduce it w.r.t. $q > p$.

If we reduce all the monomials in the recurrence (3.23) completely (i.e., we reduce the part with no $(K_p - 1)$ in front until it is irreducible), then we can write the recurrence in the form

$$\begin{aligned} \sum_{(i,\mathbf{j})\in S_0} b_{i,\mathbf{k}}(n,\mathbf{k})N^{-i}K_1^{-j_1}\cdots K_r^{-j_r} + \\ \sum_{l=1}^{r}(K_l-1)\sum_{(i,\mathbf{j})\in S_l} c_{l,i,\mathbf{j}}(n,\mathbf{k})N^{-i}K_1^{-j_1}\cdots K_l^{-j_l-1}\cdots K_r^{-j_r} \end{aligned} \tag{3.25}$$

with certain $b_{i,\mathbf{j}}, c_{l,i,\mathbf{j}} \in \mathbf{R}[n, \mathbf{k}]$, and

$$S_0 = \{(i, \mathbf{j}) \in S \mid (i, \mathbf{j}) \text{ is irreducible w.r.t. } S\}$$
$$S_l = \{(i, \mathbf{j}) \in S \mid (i, \mathbf{j}) \text{ is reducible w.r.t. } S \text{ and } l\}.$$

This is exactly the *pre-certificate* recurrence we were looking for. A part of the transformation into a certificate recurrence has been performed, and this without destroying the shape of the structureset. Note that $|S| = \sum_{i=0}^{r} |S_i|$, i.e., we did not introduce any additional monomial $N^i \mathbf{K}^{\mathbf{j}}$. It is interesting to see that if we work with a structureset of the form $S_{I,\mathbf{J}}$ then (3.25) is already (up to a shift) in certificate recurrence form.

As an example, let us consider the recurrence (3.18). The structureset of this recurrence equals

$$\{(0,0,0), (0,0,-1), (0,-1,0), (0,-1,-1), (-1,-1,-1)\}. \tag{3.26}$$

We see that $(0,0,0)$ and $(-1,-1,-1)$ are irreducible, that $(0,0,-1)$ and $(0,-1,0)$ are reducible to $(0,0,0)$, and that $(0,-1,-1)$ is reducible to $(0,0,-1)$, which in turn is reducible to $(0,0,0)$. For the reducible monomials of the recurrence we have

$$\begin{aligned}
(n-2i-2j-3)J &= (n-2i-2j-1) + (J-1)(n-2i-2j-1) \\
(n+2i+3)I &= (n+2i+1) + (I-1)(n+2i+1) \\
(n+2i+3)IJ &= (n+2i+1)J + (I-1)((n+2i+1)J) \\
&= (n+2i+1) + (I-1)((n+2i+1)J) + (J-1)(n+2i+1).
\end{aligned}$$

So the recurrence (3.18) can be written as the pre-certificate

$$(n+1) - (n+1)NIJ + (I-1)(-(n+2i+1) + (n+2i+1)J) + (J-1)(2(n-j)).$$

Note that this pre-certificate recurrence is actually different from the certificate recurrence. The decomposition of the structureset (3.26) into structuresets S_0, S_1, and S_2 according to (3.25) is: $S_0 = \{(0,0,0), (-1,-1,-1)\}$, $S_1 = \{(0,-1,0), (0,-1,-1)\}$, and $S_2 = \{(0,0,-1)\}$.

We are now able to give the second generalization of Sister Celine's technique; informally it is described as follows.

Second generalization of Sister Celine's technique.

- The input are a proper hypergeometric function $F_t(n, \mathbf{k})$ and a P-maximal structureset S for it. We make an Ansatz of the form (3.25) with

$$b_{i,\mathbf{j}}(n,k) = \sum_{\mathbf{m}=\mathbf{0}}^{\mathbf{M}_{i,\mathbf{j}}} b_{i,\mathbf{j},\mathbf{m}}(n)\mathbf{k}^{\mathbf{m}} \quad \text{and} \quad c_{l,i,\mathbf{j}}(n,k) = \sum_{\mathbf{m}=\mathbf{0}}^{\mathbf{M}_{l,i,\mathbf{j}}} c_{l,i,\mathbf{j},\mathbf{m}}(n)\mathbf{k}^{\mathbf{m}}.$$

  As before, the degree bounds $\mathbf{M}_{i,\mathbf{j}}$ and $\mathbf{M}_{l,i,\mathbf{j}}$ have to be given by the user.

- We have to find the unknown polynomials such that, when we write the $(K_l - 1)$-free part of the pre-certificate recurrence as a certificate recurrence, the principal part of this certificate recurrence is free of $\mathbf{k}$. We do this as above, by setting up and solving the reduction systems for the $b_{i,\mathbf{j}}(n, \mathbf{k})$, and by reducing the Ansatz.

- The reduced Ansatz is then solved as usual, i.e., by solving a homogeneous linear equation system.

Let us compare the the pre-certificate method with the first generalization of Sister Celine's technique of the previous subsection. If the degree bounds are chosen to be constant, then the computational power of both generalizations is the same, i.e., they find the same solutions. The advantage of the pre-certificate method is that the reduction systems are much smaller. But since these reduction systems are equation systems over the integers, not much time is spent to solve them. Indeed, experiments with our Mathematica implementations `FindRecurrence` and `FindCertificate`, which are described in Chapter 4, show that the performances of both algorithms are nearly identical.

3.6 Summary

We give a short summary of the main results of this chapter.

- In Section 3.2 we introduced the notion of a certificate recurrence, i.e., a recurrence operator of the form $S_0(n, N) + \sum_{l=1}^{r} \Delta_l S_l(n, \mathbf{k}, N, \mathbf{K})$. We gave the first complete proof that every k-free recurrence can be transformed into a certificate recurrence with nontrivial S_0.

- In Section 3.3 we investigated sums with standard boundary conditions and showed that the principal part $S_0(n, N)$ of a certificate recurrence annihilating the summand is a recurrence annihilating the sum.

- Section 3.4 was devoted to sums with nonstandard boundary conditions.

- In Section 3.5 we gave two efficient generalizations of Sister Celine's technique. The first generalization is based on the observation that under certain simple conditions also recurrences that are not necessarily free of the summation variables, can be transformed into certificate recurrences. For the second generalization we used the fact that a part of the transformation into a certificate recurrence can be done a priori.

Chapter 4

My Mathematica Package MultiSum

4.1 Description of MultiSum

The Mathematica package MultiSum contains functions to generate computer proofs of hypergeometric multisum identities. It is loaded into the Mathematica session by

```
In[1]:= <<MultiSum.m

Out[1]= MultiSum - Kurt Wegschaider - RISC Linz - 1996-7
```

The most important function of MultiSum is `FindRecurrence`, which computes certificate recurrences for a proper hypergeometric function by using the generalization of Sister Celine's technique as described in Subsection 3.5.1. For proving identities we have two other useful functions: `SumCertificate` to extract the recurrence for the sum (with standard boundary conditions) from a certificate recurrence and `CheckRecurrence` to check whether a proper hypergeometric function satisfies a recurrence or not. As a very simple example let us see how we can use the package to prove the trinomial identity $\sum_{i,j} \binom{n}{j}\binom{j}{i} x^i y^{j-i} z^{n-j} = (x+y+z)^n$.

```
In[2]:= FindRecurrence[ Binomial[n,j] Binomial[j,i] x^i y^(j-i) z^(n-j),
 n, {i,j}]

Size of equation system : 3x4

Out[2]= {-(x + y + z) F[-1 + n, -1 + i, -1 + j] + F[n, -1 + i, -1 + j] ==

   Delta[i, y F[-1 + n, -1 + i, -1 + j] + z F[-1 + n, -1 + i, j] -

      F[n, -1 + i, j]] +
```

```
   Delta[j, z F[-1 + n, -1 + i, -1 + j] - F[n, -1 + i, -1 + j]]}

In[3]:= SumCertificate[%]

Out[3]= {-(x + y + z) SUM[-1 + n] + SUM[n] == 0}

In[4]:= CheckRecurrence[ %, (x+y+z)^n]

Out[4]= {True}
```

and the identity follows from checking the case $n = 0$.

As an alternative to `FindRecurrence` we implemented the functions `FindCertificate` and `FindRationalCertificate`, primarily to test whether algorithms that directly look for certificates are superior (they are not). Additionally, MultiSum contains several other functions, like `SimplifyRecurrence` or `CertificateToRecurrence`, to support manipulations of recurrences.

Computing a recurrence for the summand of a multisum can require a long time, so it is important to improve that part of the program, that uses the largest amount of time: the computation of the nullspace of a matrix. Mathematica provides the function `NullSpace` but it only works well for very small equation systems. Therefore we replaced it by the function `ENullSpace`, written by E. Aichinger, which turns out to be much more efficient than the Mathematica built-in algorithm. The most efficient nullspace function we currently have is `CNullSpace`, which was also written by E. Aichinger ([Aic97]). `CNullSpace` is a Mathematica function that calls a C program and uses Mathematica's MathLink facility ([Wol93]) and the computer algebra C library SACLIB ([C+93]).

Once installed, `CNullSpace` can be easily used by typing the following commands.

```
In[5]:= <<sacLink.m

In[6]:= Setup

--- Usage: CNullSpace [ Matrix ] ---

In[7]:= SetOptions[ MultiSum, EquationSolver -> CNullSpace]

Out[7]= {EquationSolver -> CNullSpace, WZ -> False, Protocol -> Automatic,

  TrivialSolutions -> False, VerbaetenBound -> 500,

  RepeatedDivision -> True}
```

By setting the option `EquationSolver` to the name of an arbitrary nullspace function the user of MultiSum is free to use any nullspace function he wishes. There are a few other options, described in detail later, that regulate the behaviour of the functions of MultiSum, e.g., `Protocol` controls the amount of information (e.g., the size of the equation system) printed during the computation and `WZ` is used to switch Verbaeten completion on or off.

MultiSum also provides the possibility to get online information about the usage of every function and option, e.g.,

```
In[8]:= ? FindRecurrence
```

displays information how to use `FindRecurrence`.

4.1.1 The Input Structure

All the recurrence finding programs take as input the summand of the given multiple sum, the main variable(s), and the summation variable(s). Additionally, the input may contain orders for the recurrence, structuresets or degreebounds. In the following we describe the structure of the input.

Every Mathematica symbol can serve as a variable, e.g., `n`, `k1`, or `var`. The main variable(s), from now on denoted by *MainVars*, and the summation variable(s), denoted by *SumVars*, are either variables or list of variables. The functions look for certificate recurrences for the summand in *MainVars* with delta parts in *SumVars*. Thus the resulting recurrence for the standard sum will be a recurrence in the *MainVars*. Usually *SumVars* is a list of variables and *MainVars* is a single variable, but occasionally the sum involves two discrete main variables and it may be of advantage to look for a recurrence in both of them.

The *Summand* is defined, similar to the input for the Paule-Schorn implementation of Zeilberger's algorithm ([PS95]), as

$$\begin{array}{rcl}
\mathit{Summand} & := & <\mathit{hgterm}> \\
<\mathit{hgterm}> & := & <\mathit{simpleterm}> \\
 & \mathit{or} & \texttt{Power[} <\mathit{hgterm}> \texttt{,} <\mathit{integer}> \texttt{]} \\
 & \mathit{or} & \texttt{Times[} <\mathit{hgterm}> \texttt{,} <\mathit{hgterm}> \texttt{]} \\
<\mathit{simpleterm}> & := & <\mathit{rational}> \\
 & \mathit{or} & \texttt{Binomial[}<\mathit{intlinpoly}> \texttt{,} <\mathit{intlinpoly}>\texttt{]} \\
 & \mathit{or} & \texttt{Factorial[} <\mathit{intlinpoly}> \texttt{]} \\
 & \mathit{or} & \texttt{Gamma[} <\mathit{intlinpoly}> \texttt{]} \\
 & \mathit{or} & \texttt{Power[} <\mathit{constrational}> \texttt{,} <\mathit{intlinpoly}> \texttt{]}
\end{array}$$

where $<\mathit{integer}>$ is an arbitrary integer, $<\mathit{rational}>$ is any rational function, $<\mathit{intlinpoly}>$ is a polynomial that is integer-linear in the *MainVars* and *SumVars*, and $<\mathit{constrational}>$ is a rational function that does neither contain the *MainVars* nor the *SumVars*. As usual, the mathematica expressions `Power[x,y]`, `Times[x,y]`, and `Factorial[x]` may be abbreviated by `x^y`, `x*y`, and `x!`. Note that the definition of the *Summand* differs in two points from the definition of a proper hypergeometric term: a rational function can be used instead of the polynomial part, and the term may contain binomial coefficients. The binomial coefficient

`Binomial[n,k]` is interpreted as `n! / (k! (n-k)!)`, and the resulting recurrence is a recurrence for the Gamma functions, and does, as already mentioned several times in Section 2.7, not necessarily hold for the binomial coefficient if n is a negative integer.

To look for recurrences we need a structureset and the simplest way to define one is by specifying orders for the recurrence. The order(s) for the recurrence in the main variable(s), denoted by *MainOrders*, is a nonnegative integer if *MainVars* is a single variable, and is a list of nonnegative integers if *MainVars* is list of variables. Similarly, *SumOrders* is a nonnegative integer or a list of nonnegative integers, depending on *SumVars*. If *MainOrders* or *SumOrders* are lists then they must have the same length as *MainVars* respectively *SumVars*. The structureset defined by these orders is a rectangular Wilf-Zeilberger structureset.

We also may explicitly give a structureset as input. *StructureSet* must have the following form:

$$\begin{array}{rcl} \textit{StructureSet} & := & \{ < \textit{strctelem} >, \ldots, < \textit{strctelem} > \} \\ < \textit{strctelem} > & := & \{ < \textit{integer} >, \ldots, < \textit{integer} > \} \end{array}$$

where the last list has the same length as *MainVars* and *SumVars* together.

The generalizations of Sister Celine's technique require a degree bound for the unknown polynomial coefficients in the *SumVars*. *DegreeBound* must have the following form

$$\begin{array}{rcl} \textit{DegreeBound} & := & < \textit{nonneginteger} > \\ & \textit{or} & \{ < \textit{nonneginteger} >, \ldots, < \textit{nonneginteger} > \} \\ & \textit{or} & \texttt{Fillup} + < \textit{integer} > \end{array}$$

where the list is of the same length as *SumVars*. If *DegreeBound* is a list of integers then the degree of every polynomial coefficient in the i-th summation variable is bounded by the i-th element of *DegreeBound*, if *DegreeBound* is a single integer then it is a bound for the degree in every summation variable. The symbol `Fillup` specifies that for every polynomial coefficient the degree bound is chosen to be maximal such that the resulting polynomial equation has the same degree in the *SumVars* as with polynomial coefficients free of the *SumVars* (so that the linear equation system has the same number of equations). We may add an arbitrary integer to `Fillup`. However, `Fillup` is hardly ever used, since small integer values like 1 or 2 are usually sufficient for the degree bound.

The output of `FindRecurrence` and the related functions is a list of certificate recurrences that involve the symbol `F` to denote the given summand and the symbol `Delta[var, term]` to denote the forward-shift difference operator in the variable `var`. The output of the function `SumCertificate` is a list of recurrences involving only the symbol `SUM`, which denotes the sum with standard boundary conditions of the summand.

4.1.2 FindRecurrence and SufficientSet

The function `FindRecurrence` is the central function of MULTISUM. It tries to find certificate recurrences for a hypergeometric *Summand* in the *MainVars* with delta parts in the *SumVars*. The certificate recurrences are either the certificate form of k-free recurrences computed with Sister Celine's technique (if no *DegreeBound* is specified) or are certificate recurrences computed with the generalization of Sister Celine's technique described in Subsection 3.5.1 (if a

DegreeBound is specified). `FindRecurrence` works with P-maximal structuresets by computing the Verbaeten completion of every structureset, unless we switch off Verbaeten completion by setting the options `WZ` to `True`.

There are several ways of invoking the function:

`FindRecurrence[` *Summand*, *MainVars*, *MainOrders*, *SumVars*, *SumOrder*] and
`FindRecurrence[` *Summand*, *MainVars*, *MainOrders*, *SumVars*, *SumOrder*, *DegreeBound*]
try to find recurrences on the Verbaeten completion of the structureset specified by the orders.

`FindRecurrence[` *Summand*, *MainVars*, *SumVars*] and
`FindRecurrence[` *Summand*, *MainVars*, *SumVars*,*DegreeBound*]
find recurrences by trying higher and higher orders until a recurrence has been found.

`FindRecurrence[` *Summand*, *MainVars*, *SumVars*, *StructureSet*] and
`FindRecurrence[` *Summand*, *MainVars*, *SumVars*, *StructureSet*, *DegreeBound*]
tries to find recurrences on the Verbaeten completion of *StructureSet*.

We demonstrate how the function is used by proving the identity (a certain orthogonality relation, see [AP92])

$$\sum_i \sum_j (-1)^{i+j} \binom{i+j}{i} \binom{m}{i} \binom{n}{j} \;=\; \delta_{n,m}, \quad \text{integers } n, m \geq 0. \tag{4.1}$$

A recurrence for the summand is easily found:

```
In[8]:= FindRecurrence[ (-1)^(i+j) Binomial[i+j,i] Binomial[m,i]
 Binomial[n,j], n, {i,j}]

Size of equation system : 2x2
Size of equation system : 7x5
Size of equation system : 16x11

Out[8]= {(-m + n) F[n, -1 + i, -2 + j] ==

  Delta[i, -(n F[-1 + n, -1 + i, -2 + j]) + 2 n F[-1 + n, -1 + i, -1 + j] -

     n F[-1 + n, -1 + i, j] + (-1 - 2 n) F[n, -1 + i, -1 + j] +

     (1 + 2 n) F[n, -1 + i, j] + (-1 - n) F[1 + n, -1 + i, j]] +

  Delta[j, n F[-1 + n, -1 + i, -2 + j] + (m - n) F[n, -1 + i, -2 + j]]}
```

which proves the identity if $n \neq m$. We can also use the function call

```
In[8]:= FindRecurrence[ (-1)^(i+j) Binomial[i+j,i] Binomial[m,i]
 Binomial[n,j], n, 1, {i,j}, {1,1}]
```

to get the same result. We find a simple recurrence for the summand in the case $n = m$ by using a small *DegreeBound*. Note that the following recurrence yields a first order recurrence for the double sum, whereas the simplest k-free recurrence we found is of order 2 (see Section 3.5).

```
In[9]:= FindRecurrence[ (-1)^(i+j) Binomial[i+j,i] Binomial[n,i]
 Binomial[n,j], n, {i,j}, {1,0}]

Size of equation system : 4x2
Size of equation system : 13x8

Out[9]= {n F[-1 + n, -1 + i, -1 + j] - n F[n, -1 + i, -1 + j] ==

  Delta[i, -(n F[-1 + n, -1 + i, -1 + j]) + n F[-1 + n, -1 + i, j] +

     (-2 + 2 i - n) F[n, -1 + i, j]] +

  Delta[j, n F[n, -1 + i, -1 + j]]}
```

To see that looking for recurrences with more than one main variable can be advantageous, we call

```
In[10]:= FindRecurrence[ (-1)^(i+j) Binomial[i+j,i] Binomial[m,i]
 Binomial[n,j], {n,m}, {0,0}, {i,j}, {1,1}]
```

and find among the six solutions the following simple one, that can be used to prove (4.1) without having to consider the case $n = m$ separately.

```
Out[10]= {-(1 + n) F[n, m, -1 + i, -1 + j] +

 (1 + m) F[1 + n, 1 + m, -1 + i, -1 + j] ==

 Delta[i, (1 + n) F[n, m, -1 + i, -1 + j] + (-1 - n) F[n, m, -1 + i, j] +

    (2 + m + n) F[1 + n, m, -1 + i, j] +

    (-1 - m) F[1 + n, 1 + m, -1 + i, j]] +

 Delta[j, (-1 - m) F[1 + n, 1 + m, -1 + i, -1 + j]]}

In[11]:= SumCertificate[ % ]

Out[11]= {-(1 + n) SUM[n, m] + (1 + m) SUM[1 + n, 1 + m] == 0}
```

We also implemented the function `SufficientSet` that returns a structureset such that a k-free recurrence exists on it, i.e., such that the number of unknowns exceeds the number of equations:

`SufficientSet[` *Summand*, *MainVars*, *SumVars*`]`

If it is called with single variables, then the sufficient set according to Verbaeten's theory is returned. If *MainVars* or *SumVars* is a list of variables (even consisting of only one element), then the structureset is found by trying higher and higher orders for the recurrence. It is recommended not to use it for multisums: the structuresets are extremely large (e.g., the structureset suggested for the simple example above has more then 300 elements). If the options `WZ` is set to `True` then only rectangular Wilf-Zeilberger structuresets are used.

4.1.3 FindCertificate and FindRationalCertificate

We also implemented functions that directly look for a certificate recurrence: as described in Section 3.5.2 an Ansatz for the recurrence of the form

$$\sum_{(i,\mathbf{j})\in S_0} a_{i,\mathbf{j}}\, F(n-i,\mathbf{k}-\mathbf{j}) \;+\; \sum_{l=0}^{r} \Delta_{k_l} \sum_{(i,\mathbf{j})\in S_l} b_{l,i,\mathbf{j}}\, F(n-i,\mathbf{k}-\mathbf{j})$$

is made. The function to compute certificate recurrences with this approach — the second generalization of Sister Celine's technique — is called `FindCertificate`. As described, the structuresets S_l can be computed from a usual structureset, so `FindCertificate` can be used in exactly the same way as `FindRecurrence` — the only difference is that the *DegreeBound* is no longer optional.

`FindCertificate[` *Summand*, *MainVars*, *MainOrders*, *SumVars*, *SumOrder*, *DegreeBound*`]`

`FindCertificate[` *Summand*, *MainVars*, *SumVars*, *DegreeBound*`]`

`FindCertificate[` *Summand*, *MainVars*, *SumVars*, *StructureSet*, *DegreeBound*`]`

Not only that this function is used in the same way as `FindRecurrence`, it also has the same computational power. But we also can use `FindCertificate` in an essentially different way, by giving the sets S_l as input:

`FindCertificate[` *Summand*, *MainVars*, *MainStructureSet*, *SumVars*, *ListOfSumStructure-Sets*, *DegreeBound*`]`
where *ListOfSumStructureSets* is a list of structuresets (one structureset for every summation variable). This function call is not very useful to find recurrences (too many parameters to specify), nevertheless we give an example of how to use it. The identity (a generalized Vandermonde identity, see p. 248 in [GKP94])

$$\sum_i \sum_j \binom{r}{i}\binom{s}{j}\binom{t}{n-i-j} \;=\; \binom{r+s+t}{n}, \quad \text{integer } n,$$

can be proved with the following recurrence.

```
In[12]:= FindCertificate[ Binomial[r,i] Binomial[s,j] Binomial[t,n-i-j],
 n, {{1,0,0},{0,0,0}}, {i,j}, { {{0,0,0}}, {{0,0,0}} },1]

Size of equation system : 13x10

Out[12]=
{(-1 + n - r - s - t) F[-1 + n, i, j] + n F[n, i, j] ==

   Delta[i, -(i F[n, i, j])] + Delta[j, -(j F[n, i, j])]}
```

We also implemented a function that tries to find certificate recurrences that have rational functions as certificates, i.e., the approach described by Wilf and Zeilberger in [WZ92a]. As already mentioned there is no satisfying way to find the denominators of the rational functions, so as in Zeilberger's implementation we have to give them as input:

`FindRationalCertificate[` *Summand*, *MainVars*, *MainOrders*, *SumVars*, *ListOfPolynomials*, *DegreeBound*`]`
where *ListOfPolynomials* is a list of polynomials (one for each summation variable) which are used as denominators. The program was only added to our package for demonstration purposes: we showed that usually such a rational function approach is much slower even when we know the denominators (we got these denominators by transforming the delta parts of a certificate recurrence found by `FindRecurrence` into rational functions by using `CertificateToRational`).

4.1.4 Miscellaneous Functions and Options

The following functions are provided to support the manipulation of recurrences. They can be called either with a single recurrence or a list of recurrences (except `RecurrencePlus`).

- `SumCertificate[` *CertificateRecurrence*`]` computes the recurrence for the sum with standard boundary conditions from a given certificate recurrence for the summand.
- `CheckRecurrence[` *Recurrence*, *Term*`]` returns `True` if the hypergeometric *Term* satisfies *Recurrence*, `False` otherwise. The recurrence may be any recurrence that a function of MultiSum returns (e.g., `FindRecurrence`, `SumCertificate`).
- `CertificateToRecurrence[` *CertificateRecurrence*`]` transforms a certificate recurrence into a pure recurrence (without delta parts).
- `RecurrenceToCertificate[` *Recurrence*`]` transforms a pure recurrence into a certificate recurrence. The function assumes that there is only one main variable, otherwise the number of main variables has to given as second argument: `RecurrenceToCertificate[` *Recurrence*, *NumberOfMainVars*`]`. The behaviour of the function is controlled by the option `RepeatedDivision`.
- `RecurrencePlus[` *Recurrence1*, *Recurrence2*`]` adds the two recurrences.

- `RecurrenceTimes[` *Recurrence*, *Factor*`]` multiplies *Recurrence* with *Factor*.
- `SimplifyRecurrence[` *Recurrence*`]` simplifies a recurrence by, e.g., factoring the polynomial coefficients.
- `ShiftRecurrence[` *Recurrence*`]` shifts *Recurrence* to a normal form, i.e., a recurrence without negative shifts.
- `ShiftRecurrence[` *Recurrence*, { *Variable*, *Integer* }`]` shifts the recurrence in the *Variable* by *Integer*.
- `CertificateToRational[` *CertificateRecurrence*, *Term*`]` transforms the delta parts of a certificate recurrence into rational multiples of *Term*.
- `RecurrenceToRational[` *Recurrence*, *Term*`]` transforms a whole recurrence into a rational multiple of *Term* (is 0 if *Term* satisfies the recurrence).

There are six options that control the behaviour of the functions. They are either set permanently, e.g., by

```
SetOptions[ MultiSum, Protocol -> True, EquationSolver -> CNullSpace]
```

or for only one function call by appending the option(s) as last argument(s):

```
FindRecurrence[ Binomial[n,k], n, k, WZ -> True, Protocol -> False]
```

The current value of the options can be displayed with `Options[ MultiSum]`.

- `EquationSolver` contains the name of the nullspace algorithm.
- `WZ`: if set to `True` then no Verbaeten completion is done with the structuresets.
- `Protocol` controls the amount of information printed: with `Automatic` only the size of the equation system is printed, with `True` more information like the Verbaeten completion of the structureset are printed, with `All` nearly all intermediate results are printed (for debugging purposes only), and with `None` (or any other value) nothing is printed.
- `VerbaetenBound` is the maximal number of tuples returned by the subprogram computing the Verbaeten completion (in case of an infinite set of integer lattice points satisfying the inequalities).
- `TrivialSolutions`: if set to `True`, solutions with trivial principal part are included into the set solutions returned by `FindRecurrence` and related programs.
- `RepeatedDivision` controls the behaviour of the function `RecurrenceToCertificate`. This option is usually only used by function calls inside the package itself. `RecurrenceToCertificate` transforms a pure recurrence into a certificate recurrence by dividing the recurrence operator successively by the delta operators. If the option is set to `False` then every division by a delta operator is performed exactly once, so the resulting recurrence may have trivial principal part. If it is `True` or any other value then it multiple divisions and multiplications with summation variables (according to Theorem 3.2) are allowed.

4.2 Running Time

We improved Sister Celine's technique considerably by using Verbaeten completion and certificate recurrences that contain the summation variables. Now it is time give numerical evidence that the programs we implemented are faster than previous implementations. In the following we give the running times, which were used to compute recurrences for the summands of the following four double sums[1]:

$$(1) \qquad \sum_j \sum_k \binom{n}{k}\binom{n+k}{k}\binom{k}{j}^3$$

$$(2) \qquad \sum_i \sum_j \binom{i+j}{i}\binom{n-i}{j}\binom{n-j}{n-i-j}$$

$$(3) \qquad \sum_i \sum_j \binom{i+j}{i}\binom{m-i+j}{j}\binom{n-j+i}{n-j}\binom{m+n-i-j}{m-i}$$

$$(4) \qquad \sum_i \sum_{j=0}^{n} \binom{i+j}{i}^2 \binom{4n-2i-2j}{2n-2i}$$

The following table contains the size of the equation system, the order of the resulting recurrence for the double sum, and the running time (in seconds) used to compute certificate recurrences for the summands. We used the function `FindRecurrence` with Verbaeten completion and Aichinger's SACLIB `CNullSpace` and computed certificate recurrences that contain the summation variables.

	time in seconds	size of equation system	order of recurrence
(1)	5	27×21	2
(2)	10	33×19	2
(3)	86	50×23	2
(4)	325	63×48	1

This table and the following tables contain only the time actually used to compute the recurrence, it does not include the unsuccessful tries with smaller parameters: e.g., the time given for (2) is the time used by the function call

```
In[1]:= FindRecurrence[ Binomial[i+j,i] Binomial[n-i,j] Binomial[n-j,n-i-j],
n, 1, {i,j}, {1,0}, 1].
```

In the following we check how the choice of the nullspace algorithm, Verbaeten completion, and the use of certificate recurrences with summation variables influence the running time. The computations were done in Mathematica 2.2 on a SGI workstation.

Since most of the computation time is used for solving a system of linear equation, we first check how the running time is affected by using other Nullspace programs. The following table contains the running times for the computation of the same recurrences consumed by three different Nullspace algorithms.

[1]See Chapter 5 for more details on these double sums.

	Aichinger's SACLIB `CNullSpace`	Aichinger's Mathematica `ENullSpace`	Mathematica built-in `NullSpace`
(1)	5	15	8
(2)	10	35	60
(3)	86	731	—
(4)	325	5460	—

We easily see that the Mathematica built-in `NullSpace` works quite well for small examples, but is prohibitively slow for all larger equation systems: we were not able to compute the examples (3) and (4) with it. It is therefore necessary to replace it, preferably by a program that, like `CNullSpace`, does not work inside of Mathematica. But also E. Aichinger's Mathematica `ENullSpace` algorithm is a reasonable alternative to the the built-in function.

Next we check the influence of Verbaeten completion on the running time. The following table contains the times used to find the certificate recurrences if we use only WZ structuresets (i.e., rectangular structuresets $S_{I,\mathbf{J}}$).

	time in seconds	size of equation system
(1)	280	75×84
(2)	140	88×39
(3)	1175	98×39
(4)	13130	148×98

We see that the equation systems are much larger and that the effect on the running time is tremendous.

Let us compare our generalization of Sister Celine's technique with the usual Sister Celine technique, i.e., let us compute k-free recurrences. Since k-free recurrences are usually much larger than recurrences containing the summation variables, the time used to find them is much higher. Indeed, we were unable to find k-free recurrences for the summands of (3) and (4).

	time in seconds	size of equation system	order of recurrence
(1)	145	62×40	7
(2)	5	27×11	3

Note that (2) is one of the rare examples where a k-free recurrence can be found in smaller time than a certificate recurrence.

What happens if we compute k-free recurrences by using rectangular Wilf-Zeilberger structuresets instead of P-maximal structuresets (i.e., no Verbaeten completion). This table contains the times used to find the same k-free recurrences as above:

	time in seconds	size of equation system
(1)	—	320×128
(2)	618	189×36

Just a look at the size of the equation systems will convince everybody of the superiority of the Verbaeten completion technique.

Now we investigate our functions to find certificate recurrences directly: `FindCertificate` and `FindRationalCertificate`. The following table contains the running time in seconds and in brackets the size of the equation systems and the maximal degree (in the summation variables) of the polynomials in the delta parts.

	`FindCertificate` with P-maximal structuresets	`FindRationalCertificate`
(1)	4 (27×21, 2)	131 (66×42, 5)
(2)	9 (33×19, 1)	183 (94×59, 6)
(3)	70 (50×23, 1)	45090 (111×75, 5)
(4)	387 (63×48, 2)	48 (53×33, 3)

Using `FindCertificate` with P-maximal structuresets we have to solve equation systems of the same size as we had when we used `FindRecurrence`, and we found the same solutions in nearly the same time. Thus the behaviour of the fastest versions of `FindRecurrence` and `FindCertificate` is (nearly) identical.

Let us turn to the program `FindRationalCertificate`, i.e., our implementation of Zeilberger's approach to compute rational function certificates. Since we have no natural way of finding the denominators of the rational certificates a priori, the question arises, what denominators we have chosen as input. We simply transformed the solution we got with `FindRecurrence` into rational function certificates, and used their denominators as input. It is interesting that they turned out to be smaller than those originally used in [WZ92a]; for instance, a dramatic difference can be observed for Strehl's example ([Str94]) in Section 5.2. Even with this a priori knowledge, `FindRationalCertificate` usually does not have a chance against `FindRecurrence`. The only exception is example (4), for which we found a significantly simpler recurrence with rational certificates: a recurrence of order 0 (see Section 5.5).

Chapter 5

Some Computer Generated Proofs

5.1 Introduction

In this chapter we give, using our Mathematica package MultiSum, computer generated proofs of several binomial summation identities. The central part of a proof is the certificate recurrence for the proper hypergeometric summand, and this recurrence can be verified independently. Although the creative part of these proofs — finding the recurrence — is done by the computer, there remain some steps that have to be done by the human. These steps include checking that enough initial values of the conjectured identity are identical (which can be nontrivial if the term contains additional parameters), computing a recurrence for the sum (which is only a problem if the sum has nonstandard boundary conditions), and showing that a recurrence that we found for the proper hypergeometric interpretation of the binomial summand also holds for the binomial summand itself (only a problem outside the set of well-defined values of the proper hypergeometric term).

We would like to emphasize that without showing that the recurrence holds for the binomial summand and not just for the proper hypergeometric function, most of the proofs are not complete (semi-rigorous ?). The main problem is the difference between the binomial coefficient $\binom{n}{k}$ and its proper hypergeometric interpretation $n!/(k!(n-k)!)$: the latter is not defined for $n \in \{-1, -2, \ldots\}$. But the standard summation technique (Definition 3.4 and Theorem 3.5) requires that the recurrence holds for enough values outside of the summation range, so it is very often necessary to show that the recurrence holds for the binomial summand at these negative values. This problem was already investigated in Section 2.7, where we showed that we can usually overcome the problem by using a *polynomial argument* or a *limit argument*. There are only a few examples, e.g.,

$$\sum_i \sum_j \binom{r}{i}\binom{s}{j}\binom{t}{n-i-j} = \binom{r+s+t}{n},$$

where it not necessary to make such considerations (here because the summation variables do not occur in a numerator factorial). Sometimes such a proof can be achieved by a polynomial argument, e.g., the summand of

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\binom{i+j}{i}^2\binom{4n-2i-2j}{2n-2i}$$

is a polynomial in j if n and i are fixed integers, and if a polynomial recurrence holds for enough values of j it holds for all j. But for most sums we will use a limit argument, e.g., the summand

$$\sum_i\sum_j(-1)^{i+j}\binom{i+j}{i}\binom{n}{i}\binom{m}{j}$$

can be handled by replacing every i by $i+\epsilon$ and taking the limit

$$\lim_{\epsilon\to 0}(-1)^{i+j+\epsilon}\frac{(i+j+\epsilon)!\,n!\,m!}{(i+\epsilon)!\,j!\,(i+\epsilon)!\,(n-i-\epsilon)!\,j!\,(m-j)!} = (-1)^{i+j}\binom{i+j}{i}\binom{n}{i}\binom{m}{j}$$

Here we used (and will frequently use in this chapter) that for arbitrary integers x, y,

$$\lim_{\epsilon\to 0}\frac{(x+y+\epsilon)!}{x!\,(y+\epsilon)!} = \binom{x+y}{x}.$$

Note that, as usual, $x!$ is defined as $\Gamma(x+1)$ for nonintegral x, and that the binomial coefficient is defined (see Section 1.2) as

$$\binom{x}{y} = \lim_{\epsilon\to 0}\frac{(x+\epsilon)!}{y!\,(x-y+\epsilon)!},$$

where $x, y\in\mathbb{C}$ and where y is an integer whenever x is a negative integer. Sometimes the limit arguments can be quite involved (see, e.g., the Carlitz sums), and in rare instances we cannot achieve our goal. But usually we can use limit arguments to reduce the number of *critical points*, i.e., the points where we do not know whether the recurrence holds or not: if we get two different sets of critical points with two limit arguments it suffices to worry only about those values where the recurrence involves values from both sets (see, e.g., the Carlitz sums).

All the recurrences in this chapter are computed with `FindRecurrence`, i.e., the implementation of the first generalization of Sister Celine's technique (Subsection 3.5.1). To every recurrence we state the function call that computed it and the consumed running time. All computations were done in Mathematica 2.2 on a SGI workstation and used E. Aichinger's `CNullSpace` function ([Aic97]).

5.2 The Apéry-Schmidt-Strehl Identity

The beautiful identity

$$\sum_k\sum_j\binom{n}{k}\binom{n+k}{k}\binom{k}{j}^3 = \sum_k\binom{n}{k}^2\binom{n+k}{k}^2,\quad\text{integer } n\geq 0 \tag{5.1}$$

originated from a number-theoretical question that A. L. Schmidt had asked, and was proved in six different ways by V. Strehl in [Str94]. In his proof of the irrationality of $\zeta(3)$ (see [vdP78]) Apéry used that the famous Apéry numbers $\sum_k \binom{n}{k}^2 \binom{n+k}{k}^2$ are annihilated by the recurrence operator

$$(n+1)^3 - (2n+3)(17n^2+51n+39)N + (n+2)^3 N^2. \tag{5.2}$$

We prove (5.1) by showing that the double sum satisfies the same recurrence relation. Among Strehl's proofs of identity (5.1) there is also one computed by D. Zeilberger with his program for finding rational certificates. It is instructive to compare the "monstrously looking" rational certificates given in [Str94] with our simple and elegant certificate recurrence (5.3) given below.

Proof of (5.1). The factorial interpretation of the summand is the proper hypergeometric function

$$F_t(n,k,j) \;=\; \frac{(n+k)!\,k!}{(n-k)!\,j!^3\,(k-j)!^3}.$$

We first show that every recurrence for $F_t(n,k,j)$ also holds for the binomial summand $F(n,k,j) = \binom{n}{k}\binom{n+k}{k}\binom{k}{j}^3$. Since F and F_t coincide on the set of well-defined values of F_t, we only have to show that the recurrence holds for every n, j and k with $n \in \mathbb{N}_0$ and $j,k \in \mathbb{Z}$ (negative values for k are the only problem). For such n,k,j we have

$$\begin{aligned}\lim_{\epsilon\to 0} F_t(n,j,k+\epsilon) \;&=\; \lim_{\epsilon\to 0} \frac{n!\,(n+k+\epsilon)!\,(k+\epsilon)!^3}{(k+\epsilon)!\,(n-k-\epsilon)!\,n!\,(k+\epsilon)!\,j!^3\,(k-j+\epsilon)!^3} \\ &=\; \binom{n}{k}\binom{n+k}{k}\binom{k}{j}^3\end{aligned}$$

and, by using a limit argument, we conclude that every recurrence for F_t is also satisfied by $F(n,j,k)$ for all $n \in \mathbb{N}_0$ and $j,k \in \mathbb{Z}$.

Our programs find the following recurrence operator annihilating $F_t(n,j,k)$:

$$(1+n)^3 - (3+2\,n)\,(39+51\,n+17\,n^2)\,N + (2+n)^3\,N^2 \;-\; \Delta_j S_j \;-\; \Delta_k S_k, \tag{5.3}$$

where

$$\begin{aligned} S_j \;&=\; -\,(1+n)^3\,K + (3+2\,n)\,(21+6\,j-12\,j^2-2\,k+30\,j\,k-12\,k^2+27\,n+9\,n^2)\,N \\ &\quad -\,(2+n)^3\,K\,N^2 \\ S_k \;&=\; -\,(1+n)^3 + (3+2\,n)\,(-21-48\,j-24\,j^2+26\,k+24\,j\,k-6\,k^2+3\,n+n^2)\,J\,N \\ &\quad -\,(2+n)^3\,N^2. \end{aligned}$$

Thus the left hand side satisfies the famous Apéry recurrence (5.1).

For the sake of completeness, let us also prove with MultiSum (an obvious alternative would be Zeilberger's fast algorithm) that the Apéry numbers satisfy the recurrence (5.2). A certificate recurrence annihilating $\binom{n}{k}^2\binom{n+k}{k}^2$ is

$$(1+n)^3 - (3+2n)(39+51n+17n^2)N + (2+n)^3N^2 -$$
$$\Delta_k(-(1+n)^3 + (3+2n)(3+4k+8k^2+3n+n^2)N - (2+n)^3N^2).$$

Thus both sums satisfy the same recurrence relation and it remains to compare the initial values: for $n=0$ both sums equal 1, for $n=1$ they both evaluate to 5. □

It might be instructive to illustrate how the recurrence (5.3) could be obtained by using the tools from the package MultiSum. First lets try to find a k-free recurrence and, indeed, we find one (in 226 seconds) which yields a recurrence for the sum of order seven.

```
In[4]:= SumCertificate[ FindRecurrence [ Binomial[n,k] Binomial[ n+k,k]
 Binomial[k,j]^3, n, {j, k}, 0]]

Size of equation system : 2x2
Size of equation system : 14x8
Size of equation system : 32x20
Size of equation system : 39x26
Size of equation system : 72x50

Out[4]=
                   3
{(2 - n) (-3 + n)  (1 + n) (1 + 2 n) (3 + 2 n) (5 + 2 n) (-4 + 3 n)

    SUM[-4 + n] + (-2 + n) (1 + n) (-5 + 2 n) (3 + 2 n) (5 + 2 n)
                            2        3       4
    (-412 - 107 n + 863 n  - 533 n  + 93 n ) SUM[-3 + n] +

   (-3 + 2 n) (3 + 2 n) (5 + 2 n)
                             2         3          4         5        6
    (872 + 4222 n - 5779 n  - 1688 n  + 4866 n  - 2166 n  + 297 n )

    SUM[-2 + n] + (-5 + 2 n) (-1 + 2 n) (5 + 2 n)
                             2         3         4         5        6
    (1032 - 2402 n - 419 n  + 4676 n  - 454 n  - 1594 n  + 201 n )

    SUM[-1 + n] + (-5 + 2 n) (1 + 2 n) (5 + 2 n)
                              2         3         4         5        6
    (-1032 - 2402 n + 419 n  + 4676 n  + 454 n  - 1594 n  - 201 n ) SUM[n] +

   (-5 + 2 n) (-3 + 2 n) (3 + 2 n)
```

```
                         2          3          4          5         6
    (-872 + 4222 n + 5779 n  - 1688 n  - 4866 n  - 2166 n  - 297 n )

    SUM[1 + n] + (-1 + n) (2 + n) (-5 + 2 n) (-3 + 2 n) (5 + 2 n)
                          2        3       4
    (412 - 107 n - 863 n  - 533 n  - 93 n ) SUM[2 + n] +
                             3
   (-1 + n) (2 + n) (3 + n)  (-5 + 2 n) (-3 + 2 n) (-1 + 2 n) (4 + 3 n)

    SUM[3 + n] == 0}
```

Note that the recurrence for this example is from `SUM[n-4]` to `SUM[n+3]` which is due to the Verbaeten completion. The user might want to normalize this range; for this purpose the function `ShiftRecurrence` can be used.

Although we are able to prove the identity with the above recurrence, we prefer to find a second order recurrence relation. So we try again with the degree bound 1, and this time we find a recurrence of order 3 in 300 seconds.

```
In[5]:= SumCertificate[ FindRecurrence [ Binomial[n,k] Binomial[ n+k,k]
 Binomial[k,j]^3, n, {j, k}, 1]]

Size of equation system : 2x2
Size of equation system : 24x20
Size of equation system : 48x62

Out[5]=
         4
{(-1 + n)  n (1 + 2 n) SUM[-2 + n] +
                                         2       3
    (1 - 2 n) (1 - n) n (-5 + 7 n + 17 n  - 33 n ) SUM[-1 + n] +
                                         2       3
    (1 - n) n (1 + 2 n) (-5 - 7 n + 17 n  + 33 n ) SUM[n] +
                       3
    (-1 + n) n (1 + n)  (-1 + 2 n) SUM[1 + n] == 0}
```

To find a second order recurrence we have to use the degree bound 2. The program needs 40 seconds to find it.

```
In[6]:= SumCertificate[ FindRecurrence [ Binomial[n,k] Binomial[ n+k,k]
 Binomial[k,j]^3, n, {j, k}, 2]]

Size of equation system : 2x2
Size of equation system : 36x40
```

```
Out[6]=
          3
{(-1 + n)  n (1 + n) SUM[-2 + n] +
                                                      2
    (1 - 2 n) (-1 - n) n (-5 + 17 n - 17 n ) SUM[-1 + n] +
     4
    n  (1 + n) SUM[n] == 0}
```

To be honest, with the above functions calls we did not find exactly one recurrence each time, but sets of solutions (2, 11, respectively 2 solutions). The reason is that the automatic choice of the orders is not always the best. With a more specialized choice of the parameters we can find smaller sets of solutions in shorter time:

```
In[7]:= FindRecurrence [ Binomial[n,k] Binomial[ n+k,k] Binomial[k,j]^3,
 n, 0, {j, k}, {0,1},  2]]

Size of equation system : 27x21

Out[7]=
  3
{n  F[-1 + n, -1 + j, -1 + k] -
                                    2
    (1 + 2 n) (5 + 17 n + 17 n ) F[n, -1 + j, -1 + k] +
             3
    (1 + n)  F[1 + n, -1 + j, -1 + k] ==
                 3
   Delta[j, -(n  F[-1 + n, -1 + j, k]) +
                                     2                   2            2
      2 (1 + 2 n) (6 - 12 j + 6 j  + k + 3 j k - 3 k  + 4 n + 4 n )

       F[n, -1 + j, -1 + k] + (1 + 2 n)
                             2                       2        2
       (-23 + 48 j - 24 j  - 22 k + 24 j k - 6 k  + n + n ) F[n, -1 + j, k] -
               3
      (1 + n)  F[1 + n, -1 + j, k]] +
                  3
    Delta[k, -(n  F[-1 + n, -1 + j, -1 + k]) +
                                     2                      2        2
      (1 + 2 n) (-7 + 24 j - 24 j  - 10 k + 24 j k - 6 k  + n + n )
                                                 3
       F[n, -1 + j, -1 + k] - (1 + n)  F[1 + n, -1 + j, -1 + k]]}
```

yields exactly one solution in only 5 seconds.

5.3 Two Summation Problems of Carlitz

In the problem section of the American Mathematical Monthly (problem E1999, [Car68]), L. Carlitz asked for proofs of the following two statements.

1. Put

$$S_n = \sum_{i+j+k=n} \binom{i+j}{i}\binom{j+k}{j}\binom{k+i}{k}, \quad \text{integer } n \geq 0.$$

   Show that $S_n - S_{n-1} = \binom{2n}{n}$.

2. Put

$$R_{m,n} = \sum_{i+j+k \leq \min(n,m)} \binom{i+j}{i}\binom{j+k}{j}\binom{m-i-j}{k}\binom{n-j-k}{i}, \quad \text{integers } m,n \geq 0.$$

   Show that $R_{m,n} - 2R_{m-1,n-1} = \binom{n+m}{m}$.

Note that the first problem is equivalent to proving the identity

$$\sum_i \sum_j \binom{i+j}{i}\binom{n-i}{j}\binom{n-j}{n-i-j} = \sum_{k=0}^{n} \binom{2k}{k}. \tag{5.4}$$

Proof of (5.4). The summand of the double sum in (5.4) can be written as the proper hypergeometric function

$$f(n,i,j) = \frac{(i+j)!\,(n-i)!\,(n-j)!}{i!^2\,j!^2\,(n-i-j)!^2}.$$

For $n \in \mathbb{N}_0$, and $i,j \in \mathbb{Z}$ we have

$$\lim_{\delta \to 0} \lim_{\epsilon \to 0} f(n+\epsilon+\delta, i+\epsilon, j+\delta) = \binom{i+j}{i}\binom{n-i}{n-i-j}\binom{n-j}{n-i-j}$$

and we see that it coincides with the binomial summand $\binom{i+j}{i}\binom{n-i}{j}\binom{n-j}{n-i-j}$ unless $j < -i < -n$. But this implies $j \leq -n-2$, so the recurrence, given below, holds also for the binomial summand for all $n \in \mathbb{N}_0$ and $i,j \in \mathbb{Z}$ as can be seen without explicitly checking at the critical values: the recurrence is of order one in j and the critical values are too far away from the summation range.

Note that such substitutions like $n \to n+\epsilon+\delta$ are easily found and that we also can use other substitutions. But unfortunately no limit of the proper hypergeometric term (with such a substitution) exactly yields the given summand.

The function $f(n,i,j)$ and thus the summand of (5.4) is annihilated by[1]

$$
\begin{aligned}
&-2\,(3+2\,n)+(8+5\,n)\,N+(-2-n)\,N^2-\\
&\Delta_i(4\,(1-i+n)+2\,(-1+i-n)\,J+(-5+4\,i+3\,j-4\,n)\,N+(-3-5\,i-3\,j+n)\,J\,N+\\
&(2+n)\,J\,N^2)-\\
&\Delta_j(2\,(1+2\,i)+(-3-4\,i-3\,j-n)\,N+(2+n)\,N^2)
\end{aligned}
$$

The recurrence annihilating the double sum thus is

$$((n+2)N-(4n+6))(N-1)$$

and we can easily check that the sum on the right hand side of (5.4) is also annihilated by this operator. It remains to check the initial values: for $n=0$ and $n=1$ both sides are 1 respectively 2. $\square$

It is interesting that we can also find[2] a simple k-free recurrence for the summand of the double sum in (5.4). But then the resulting recurrence for the sum is a third order recurrence.

Proof that $R_{m,n}-2R_{m-1,n-1}=\binom{m+n}{m}$. The proper hypergeometric function corresponding to the summand of the triple sum is

$$f(m,n,i,j,k) \;=\; \frac{(i+j)!\,(j+k)!\,(m-i-j)!\,(n-j-k)!}{i!^2\,j!^2\,k!^2\,(m-i-j-k)!\,(n-i-j-k)!}$$

and we easily see that

$$
\begin{aligned}
&\lim_{\beta\to0}\lim_{\gamma\to0}\lim_{\delta\to0}\lim_{\epsilon\to0} f(n+\epsilon+\delta+\gamma,m+\beta,i+\epsilon,j+\delta,k+\gamma)\\
&\qquad= \binom{i+j}{i}\binom{j+k}{j}\binom{m-i-j}{k}\binom{n-j-k}{n-i-j-k} \qquad (5.5)
\end{aligned}
$$

and, similarly,

$$
\begin{aligned}
&\lim_{\beta\to0}\lim_{\epsilon\to0}\lim_{\delta\to0}\lim_{\gamma\to0} f(n+\beta,m+\epsilon+\delta+\gamma,i+\epsilon,j+\delta,k+\gamma)\\
&\qquad= \binom{i+j}{j}\binom{j+k}{k}\binom{m-i-j}{m-i-j-k}\binom{n-j-k}{i} \qquad (5.6)
\end{aligned}
$$

Note that if $n\le m$ then $R_{m,n}$ equals the sum w.r.t. to $i,j,k\in\mathbb{Z}$ over (5.5), a sum with standard boundary conditions. For $n\ge m$ we have to take (5.6) as summand to get standard boundary conditions.

---

[1] `FindRecurrence[ Binomial[i+j, i] Binomial[n-i, j] Binomial[n-j, n-i-j], n, 1, {i,j}, {1,0}, 1]` in 10 seconds solving a 33×19 equation system.

[2] `FindRecurrence[ Binomial[i+j, i] Binomial[n-i, j] Binomial[n-j, n-i-j], n, 0, {i,j}, {1,1}]` in 5 seconds solving a 27×11 equation system.

The following recurrence[3] annihilates $f(m,n,i,j,k)$ and thus both (5.5) and (5.6).

$$\begin{aligned}
&2\,M + 2\,N - 2\,M\,N - M^2\,N - M\,N^2 + M^2\,N^2 - \\
&\Delta_i(K - J\,M - K\,M - K\,N + J\,M\,N + K\,M\,N - J\,K\,M\,N + J\,K\,M^2\,N + J\,K\,M\,N^2 - \\
&\;J\,K\,M^2\,N^2) - \\
&\Delta_j(-1 - M - K\,N + M\,N + K\,M^2\,N + K\,M\,N^2 - K\,M^2\,N^2) - \\
&\Delta_k(1 - M - 2\,N + M\,N + M^2\,N + M\,N^2 - M^2\,N^2)
\end{aligned}$$

If $n < m$ we apply the above recurrence to (5.5) and sum w.r.t. $i,j,k \in \mathbb{Z}$ and the sums involved are $R_{m,n}$, $R_{m+1,n}$, $R_{m,n+1}$, $R_{m+1,n+1}$, $R_{m+1,n+2}$, $R_{m+2,n+1}$, and $R_{m+2,n+2}$. Thus we get a recurrence for $R_{m,n}$ $(n < m)$, which factors into[4]

$$(NM - M - N)\,(NM - 2).$$

The left factor of this recurrence annihilates $\binom{m+n+2}{m+1}$. Similarly, if $m < n$ we can apply the recurrence to (5.6), sum w.r.t. to all $i,j,k \in \mathbb{Z}$ and get the same recurrence for $R_{m,n}$ (if $m < n$).

We cannot use the above recurrence in the case $n = m$, since the summation involves sums with nonstandard boundary conditions (because then n-shifts and m-shifts are involved with $n+\alpha_1 > m+\beta_1$ and with $n+\alpha_2 < m+\beta_2$ so that neither standard summation over (5.5) nor standard summation over (5.6) always yield $R_{n,m}$). We have to handle this case separately. If $n = m$ then the summand equals $\binom{i+j}{i}\binom{j+k}{j}\binom{n-i-j}{k}\binom{n-j-k}{n-i-j-k}$ and is annihilated by[5]

$$\begin{aligned}
&-4\,(3 + 2\,n) + 2\,(5 + 3\,n)\,N + (-2 - n)\,N^2 - \\
&\Delta_i((6 + 4\,i + 3\,n)\,K + (-4 - 2\,j - n)\,J\,N + (2\,j + 4\,k - 3\,n)\,K\,N + (-4 - 2\,j - n)\,J\,K\,N + \\
&(2 + n)\,J\,K\,N^2) - \\
&\Delta_j(6 - 4\,i + 5\,n + (-2 - 2\,j - n)\,N - 2\,(4 + 2\,k + n)\,K\,N + (2 + n)\,K\,N^2) - \\
&\Delta_k(6 + 4\,i + 3\,n + (-8 + 2\,j - 5\,n)\,N + (2 + n)\,N^2)
\end{aligned}$$

and summing w.r.t. $i,j,k \in \mathbb{Z}$ yields the recurrence

$$((n+2)N - 2(3+2n))\,(N-2)$$

annihilating $R_{n,n}$. The left factor $(n+2)N - 2(2n+3)$ annihilates $\binom{2n+2}{n+1}$ and it remains to check the initial values: $R_{m+1,1} - 2R_{m,0} = m+2$ and $R_{1,n+1} - 2R_{0,n} = n+2$. $\square$

[3]`FindRecurrence[ Binomial[i+j, i] Binomial[j+k, j] Binomial[m-i-j, k] Binomial[n-j-k, i], {n,m}, {0,0}, {i,j,k}, {0,1,0}]` in 28 seconds. This returns two linearly independent solutions, which we have to combine to get this simple recurrence.

[4]Note that in this chapter every factorization of a recurrence operator are done by hand.

[5]`FindRecurrence[ Binomial[i+j, i] Binomial[j+k, j] Binomial[n-i-j, k] Binomial[n-j-k, n-i-j-k], n, 0, {i,j,k}, {0,1,0}, 1]` in 30 seconds.

5.4 Another Summation Problem of Carlitz

The following identities, for nonnegative integers n and m, are due to Carlitz [Car64]:

$$\sum_{i+k=m}\sum_{j+l=n}\binom{i+j}{i}\binom{j+k}{j}\binom{k+l}{k}\binom{i+l}{l}$$
$$= \frac{(m+n+1)!}{m!n!}\sum_k \frac{1}{2k+1}\binom{m}{k}\binom{n}{k} = \sum_{i=0}^{m}\sum_{j=0}^{n}\binom{i+j}{i}^2\binom{n+m-i-j}{m-i}, \quad (5.7)$$

where n and m are nonnegative integers. The first sum can be rewritten as

$$\sum_i\sum_j\binom{i+j}{i}\binom{m-i+j}{j}\binom{n-j+i}{n-j}\binom{m+n-i-j}{m-i}. \quad (5.8)$$

Let us denote the sum (5.8) by $F_{m,n}$ and the two last sums in (5.7) by $G_{m,n}$, respectively $H_{m,n}$.

Proof. The proper hypergeometric term corresponding to the summand of $F_{m,n}$ is

$$f(m,n,i,j) = \frac{(i+j)!\,(m+j-i)!\,(m+n-i-j)!\,(n+i-j)!}{i!^2\,j!^2\,(m-i)!^2\,(n-j)!^2}.$$

Let $n,m\in\mathbb{N}_0$ and $i,j\in\mathbb{Z}$. It is easily seen that

$$\lim_{\delta\to 0}\lim_{\epsilon\to 0} f(m,n+\delta,i+\epsilon,j) = \binom{i+j}{j}\binom{m-i+j}{j}\binom{n-j+i}{i}\binom{m+n-i-j}{m-i} \quad (5.9)$$

which equals the binomial summand of $F_{m,n}$ unless $j>n$. Taking a different limit we get

$$\lim_{\delta\to 0}\lim_{\epsilon\to 0}\lim_{\gamma\to 0} f(m+\epsilon,n+\delta+\gamma,i+\epsilon,j+\delta)$$
$$= \binom{i+j}{i}\binom{m+j-i}{m-i}\binom{n-j+i}{n-j}\binom{m+n-i-j}{m-i} \quad (5.10)$$

which equals the binomial summand of $F_{m,n}$ unless $j<0$. To get a recurrence for the binomial summand of $F_{n,m}$ we take the limit of a recurrence for the proper hypergeometric function $f(m,n,i,j)$. So for those values of $n,m,i,j\in\mathbb{Z}$ such that the recurrence does not involve a j-n-shift with $j+\alpha>n+\beta$, we take the limit (5.9), and for those values such that the recurrence does not involve a j-shift with $j+\alpha<0$ we take the limit (5.10). Therefore each recurrence for the proper hypergeometric function also holds for the binomial summand, except possibly at those values where the recurrence involves a shift with $j+\alpha<0$ and a shift with $j+\beta>n$. The recurrence, given below, is of order one in j, so that this is impossible for this recurrence and for nonnegative n.

The function $f(m,n,i,j)$ and thus the summand of $F_{m,n}$ is annihilated by the recurrence operator[6]

$$
\begin{aligned}
&-(m+n+4)M^2 + 2(n+2)M^2N + (m+n+4)N^2 - 2(m+2)MN^2 + (m-n)M^2N^2 - \\
&\quad \Delta_i((m+n+4)M^2 - (n+2)M^2N - (n+2)JM^2N + (m+2)JMN^2 - \\
&\quad (m-n)JM^2N^2) - \\
&\quad \Delta_j(-(n+2)M^2N - (m+n+4)N^2 + 2(m+2)MN^2 - (m-n)M^2N^2)
\end{aligned}
$$

The resulting recurrence operator annihilating $F_{n,m}$ factors nicely into

$$-(M+N+MN)\left((m+n+3)M-(m+n+3)N+(m-n)MN\right) \tag{5.11}$$

It turns out that the summand of $G_{m,n}$ and therefore $G_{m,n}$ itself is annihilated by the right factor of (5.11), namely by[7]

$$(m+n+3)M - (m+n+3)N + (m-n)MN.$$

The summand of $H_{m,n}$ is annihilated by the simple recurrence operator $NM - N - M$, so summing it w.r.t. all i and w.r.t. $j = 0, \ldots, n{+}1$ yields the following inhomogeneous recurrence relation

$$(MN - M - N)H_{m,n} = \sum_i \binom{n+i+1}{i}^2 \binom{n+m+1-i-(n+1)}{m+1-i} = \binom{m+n+2}{n+1}^2$$

The last equality holds because the summand of the sum w.r.t. i is zero unless $i = m+1$. Since the recurrence (5.11) for $F_{m,n}$ can also be written as

$$\left((m-n)NM + (m+n+4)M - (m+n+4)N\right)(MN - M - N)$$

and the left factor of it annihilates $\binom{m+n+2}{n+1}^2$ we only have to check the initial values. Because of symmetry it suffices to check the identities for $(0,m)$ and $(1,m)$, $m \in \mathbb{N}$. For the first case, i.e., $n = 0$ and $m \in \mathbb{N}$:

$$
\begin{aligned}
F_{m,0} &= \sum_i \binom{i}{i}\binom{m-i}{m-i} = m+1, \\
G_{m,0} &= m+1, \\
H_{m,0} &= \sum_{i=0}^{m} \binom{i}{i}^2 \binom{m-i}{m-i} = m+1;
\end{aligned}
$$

[6] `FindRecurrence[ Binomial[i+j, i] Binomial[m-i+j, j] Binomial[n-j+i, n-j] Binomial[m+n-i-j, m-i], {n,m}, {i,j}]` in 53 seconds.

[7] `FindRecurrence[(m+n+1)!/((2k+1) m!  n!)  Binomial[m,k] Binomial[n,k], {n,m}, {k}]` in less than 1 second.

for the second case, i.e., $n=1$ and $m \in \mathbb{N}$:

$$\begin{aligned}
F_{m,1} &= 2\sum_{i=0}^{m}(m-i+1)(i+1) \;=\; \tfrac{1}{3}(m+1)(m+2)(m+3),\\
G_{m,1} &= \tfrac{1}{3}(m+1)(m+2)(m+3),\\
H_{m,1} &= \sum_{i=1}^{m+1} i + \sum_{i=1}^{m+1} i^2 \;=\; \tfrac{1}{3}(m+1)(m+2)(m+3).
\end{aligned}$$

□

The identity $F_{m,n}=G_{m,n}$ can also be proved (as in [WZ92a]) by finding a recurrence in n: the summand of $F_{m,n}$ is annihilated by[8]

$$\begin{aligned}
&-2\,(2+m+n)^2\,(3+m+n)+(3+m+n)\,(14+3\,m+15\,n+2\,m\,n+4\,n^2)\,N-\\
&(2+n)^2\,(5+2\,n)\,N^2-\\
&\Delta_i((2+m+n)\,(3+m+n)\,(2-2\,i+2\,m+n)+(-16+13\,i-4\,j-3\,i\,j-17\,m+\\
&\;3\,i\,m+3\,j\,m-i\,j\,m-3\,m^2+j\,m^2-20\,n+13\,i\,n-4\,j\,n-i\,j\,n-17\,m\,n+2\,i\,m\,n+\\
&\;j\,m\,n-2\,m^2\,n-8\,n^2+3\,i\,n^2-j\,n^2-4\,m\,n^2-n^3)\,N+(1+j)\,(-2+i-m-n)\\
&\;(3+m+n)\,J\,N+(2+n)\,(5-2\,i+5\,j+m+j\,m+2\,n-i\,n+2\,j\,n)\,J\,N^2)-\\
&\Delta_j((-20-10\,i-2\,j-4\,m-2\,i\,m-5\,j\,m-j\,m^2-28\,n-9\,i\,n-j\,n-4\,m\,n-i\,m\,n-\\
&\;2\,j\,m\,n-13\,n^2-2\,i\,n^2-m\,n^2-2\,n^3)\,N+(2+n)^2\,(5+2\,n)\,N^2)
\end{aligned}$$

and we are able to find, by Zeilberger's fast algorithm, the same recurrence for $G_{n,m}$.

5.5 The Andrews-Paule Sums

The identity

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\binom{i+j}{i}^2\binom{4n-2i-2j}{2n-2i} \;=\; (2n+1)\binom{2n}{n}^2, \quad \text{integer } n \geq 0, \tag{5.12}$$

was stated as problem E3376 in the American Mathematical Monthly by R. Blodgett. It was solved by G. Andrews and P. Paule ([AP92], [AP93]) by proving the more general identity (5.14). Note that P. Paule also gave two direct proofs of (5.12) by applying Zeilberger's algorithm in a nontrivial way ([AP93]).

[8] `FindRecurrence[ Binomial[i+j, i] Binomial[m-i+j, j] Binomial[n-j+i, n-j] Binomial[m+n-i-j, m-i], n, 0, {i,j}, 1, 1]` in 86 seconds.

Let us denote the double sum in (5.12) by $f(n)$. It is not difficult to find a recurrence for the summand of $f(n)$: it is annihilated by[9]

$$\begin{aligned}
&16\,(1+n)^2\,(1+2\,n)^2\,(5+4\,n) - (1+n)\,(1+2\,n)^2\,(3+2\,n)^2\,N - \\
&\Delta_i(a_1(n,i,j) + a_2(n,i,j)I + a_3(n,i,j)J + a_4(n,i,j)JN + a_5(n,i,j)IJN + \\
&\ a_6(n,i,j)J^2N + a_7(n,i,j)IJ^2N) - \\
&\Delta_j(b_1(n,i,j) + b_2(n,i,j)J + b_3(n,i,j)N + b_4(n,i,j)JN)
\end{aligned}$$

where $a_1,\ldots,a_7$, and $b_1,\ldots,b_4$ are polynomials of degree at most 2 in i and j. But since the sum has nonstandard boundary conditions, we have to compute boundary values, which is difficult for larger recurrences. So we prefer to prove the following generalizations that satisfy simpler recurrences:

$$\sum_{i=0}^{n}\sum_{j=0}^{m}\binom{i+j}{i}^2\binom{2m+2n-2i-2j}{2n-2i} = (m+n+1)\binom{m+n}{m}^2, \tag{5.13}$$

$$\sum_{i=0}^{\lfloor\frac{m}{2}\rfloor}\sum_{j=0}^{\lfloor\frac{n}{2}\rfloor}\binom{i+j}{i}^2\binom{m+n-2i-2j}{n-2i} = \frac{\lfloor\frac{m+n+1}{2}\rfloor!\,\lfloor\frac{m+n+2}{2}\rfloor!}{\lfloor\frac{m}{2}\rfloor!\,\lfloor\frac{m+1}{2}\rfloor!\,\lfloor\frac{n}{2}\rfloor!\,\lfloor\frac{n+1}{2}\rfloor!}. \tag{5.14}$$

Let us call the double sums in (5.13) and (5.14) $g(n,m)$, respectively $h(n,m)$.

The Andrews-Paule Proof of (5.14). The summand of (5.14) is annihilated by the recurrence operator[10] $NM - N - M$; summing it w.r.t. $i \in \mathbb{Z}$ and $j = 0,\ldots,\lfloor\frac{m+1}{2}\rfloor$ we get

$$\begin{aligned}
&h(n+1,m+1) - h(n+1,m) - h(n,m+1) \\
&\qquad = [m \text{ odd}]\sum_i \binom{i+\frac{m+1}{2}}{i}^2\binom{n-2i}{n-2i+1} = [m \text{ odd}][n \text{ odd}]\binom{\frac{m+n+2}{2}}{\frac{m+1}{2}}^2,
\end{aligned}$$

where $[m \text{ odd}]$ is defined, following [GKP94], as 1 if m is odd and 0 otherwise. The last equality holds because the summand of the sum w.r.t. i is nonzero only if $2i = n+1$. Verifying that the r.h.s. of (5.14) satisfies the same recurrence and comparing the initial values $h(n,0) = h(0,n) = \lfloor\frac{n}{2}\rfloor + 1$ proves (5.14) (and also (5.12) and (5.13)). This is exactly the proof given by Andrews and Paule ([AP92], [AP93]). □

The creative part in the above proof consists of finding the closed form for $h(n,m)$. It would be highly desirable to find a proof without a priori knowledge of this closed form. This can be easily achieved if we can find a two-dimensional Gosper representation for the summand of $h(n,m)$: $h(n,m,i,j) = (\Delta_i h_1) + (\Delta_j h_2)$. From one of P. Paule's proofs ([Pau92]) of

[9] `FindRecurrence[ Binomial[i+j, i]`^{2} `Binomial[4n-2i-2j, 2n-2i], n, 0, {i,j}, {1,1}, 2]` in 325 seconds.

[10] `FindRecurrence[ Binomial[i+j, i]`^{2} `Binomial[n+m-2i-2j, n-2i], {n,m}, {i,j}]` in less than 1 second.

identity (5.14) one can derive that such a Gosper representation exists for the summand of $f(n)$ (although this was not explicitly done in [Pau92]). So it is likely that there also exists a Gosper representation for $h(n,m,i,j)$. In our context such a representation is a certificate recurrence with a principal part consisting of exactly one term. Unfortunately we are unable to find it *directly* with our programs, but it can be computed with a linear combination of certificate recurrences found by `FindRecurrence`.

The function `FindRecurrence` delivers the following two recurrences[11] annihilating the summand $h(n,m,i,j)$ of $h(n,m)$:

$$\begin{aligned}
&-(m+n+2)+(n+1)\,N-\\
&\Delta_i((m+n-i-2j+1)-(n+1)\,N+i\,N^2-i\,J\,N^2)-\\
&\Delta_j((i+2j+1)-i\,N^2),
\end{aligned}\tag{5.15}$$

and

$$\begin{aligned}
&(m+n+2)-(3n+2m+9)\,N+2\,(n+3)\,N^2-\\
&\Delta_i(-(m+n+i+3)+(2\,m+3\,n+9)\,N-(2n-i+2j+6)\,N^2-\\
&\quad(i-2j-2)\,J\,N^2)-\\
&\Delta_j((i+1)-(i-2j)\,N^2).
\end{aligned}\tag{5.16}$$

We combine the two recurrences (we may add two recurrences, multiply a recurrence with a polynomial, and shift a recurrence) to obtain a certificate recurrence with a one-term principal part. Indeed, the linear combination $\frac{1}{2}(n+2)((5.15)+(5.16)) - (n+3)\,N\,(5.15)$ yields the recurrence (5.17), which is the desired Gosper representation. Note that these manipulations can be done with the MultiSum functions `RecurrencePlus`, `RecurrenceTimes`, and `ShiftRecurrence`.

A Computer Generated Proof of (5.14). The summand $h(n,m,i,j)$ of $h(n,m)$ is annihilated by

$$\begin{aligned}
&(m+1)\,N\ -\\
&\Delta_i\big(-(1+i+j)(2+n)+(2+3i+6j-m+n+in+2jn)\,N+(i-j)(2+n)N^2-\\
&\quad(-1+i-j)(2+n)N^2J-i(3+n)N^3+i(3+n)N^3J\big)-\\
&\Delta_j\big((1+i+j)(2+n)-(1+i+2j)(3+n)N-(i-j)(2+n)N^2+i(3+n)N^3\big).
\end{aligned}\tag{5.17}$$

Summing this recurrence w.r.t. all i and w.r.t. $j=0,\ldots,\lfloor\frac{m}{2}\rfloor$ we get

$$(m+1)\,h(n+1,m)\ =\ b_2-b_1,$$

where the boundary values b_1 and b_2 are

$$\begin{aligned}
b_1\ =\ \sum_i\big(&(i+1)(n+2)\,h(n,m,i,0)-(i+1)(n+3)\,h(n+1,m,i,0)\\
&-i(n+2)\,h(n+2,m,i,0)+i(n+3)\,h(n+3,m,i,0)\big)
\end{aligned}$$

[11] `FindRecurrence[ Binomial[i+j, i]^2 Binomial[n+m-2i-2j, n-2i], n, 0, {i,j}, {0,1}, 1]` in 4 seconds.

$$\begin{aligned} b_2 \;=\; \sum_i \Big(& (i + \left\lfloor \frac{m}{2} \right\rfloor + 2)(n+2)\, h(n, m, i, \left\lfloor \frac{m}{2} \right\rfloor + 1) \\ & - (i + 2\left\lfloor \frac{m}{2} \right\rfloor + 3)(n+3)\, h(n+1, m, i, \left\lfloor \frac{m}{2} \right\rfloor + 1) \\ & - (i - \left\lfloor \frac{m}{2} \right\rfloor - 1)(n+2)\, h(n+2, m, i, \left\lfloor \frac{m}{2} \right\rfloor + 1) \\ & + i(n+3)\, h(n+3, m, i, \left\lfloor \frac{m}{2} \right\rfloor + 1)\Big). \end{aligned}$$

We easily see that $i\,h(n+2, m, i, 0) = i\,h(n, m, i-1, 0)$ and that $i\,h(n+3, m, i, 0) = i\,h(n+1, m, i-1, 0)$ so that the first boundary value b_1 is zero. The second boundary value can be evaluated since $h(n+\alpha, m, i, \lfloor\frac{m}{2}\rfloor + 1)$ is zero unless $2i = n + \alpha$ or $2i = n + \alpha - 1$. We distinguish between odd and even n and between odd and even m, and obtain b_2 as[12]:

$$\frac{1}{(m+1)}\, b_2 \;=\; \begin{cases} \dfrac{\left(\frac{m+n+2}{2}\right)!^2}{\left(\frac{n}{2}\right)!\,\left(\frac{n+2}{2}\right)!\,\left(\frac{m}{2}\right)!^2} & m \text{ even}, n \text{ even}, \\[2ex] \dfrac{\left(\frac{m+n+1}{2}\right)!\,\left(\frac{m+n+3}{2}\right)!}{\left(\frac{n+1}{2}\right)!^2\,\left(\frac{m}{2}\right)!^2} & m \text{ even}, n \text{ odd}, \\[2ex] \dfrac{\left(\frac{m+n+1}{2}\right)!\,\left(\frac{m+n+3}{2}\right)!}{\left(\frac{n}{2}\right)!\,\left(\frac{n+2}{2}\right)!\,\left(\frac{m-1}{2}\right)!\,\left(\frac{m+1}{2}\right)!} & m \text{ odd}, n \text{ even}, \\[2ex] \dfrac{\left(\frac{m+n+2}{2}\right)!^2}{\left(\frac{n+1}{2}\right)!^2\,\left(\frac{m-1}{2}\right)!\,\left(\frac{m+1}{2}\right)!} & m \text{ odd}, n \text{ odd}. \end{cases}$$

We replace n by $n-1$, and use the floor and the ceiling function to get a closed form for $h(n, m)$:

$$h(n, m) \;=\; \frac{\left\lfloor\frac{m+n+1}{2}\right\rfloor!\,\left\lceil\frac{m+n+1}{2}\right\rceil!}{\left\lfloor\frac{n}{2}\right\rfloor!\,\left\lceil\frac{n}{2}\right\rceil!\,\left\lfloor\frac{m}{2}\right\rfloor!\,\left\lceil\frac{m}{2}\right\rceil!}.$$

It is easily seen that this closed form can be rewritten as the r.h.s. of (5.14). □

Let us turn to a Gosper representation of the summand $f(n, i, j)$ of $f(n)$, i.e., a certificate recurrence with principal part consisting of one term: $a(n) f(n, i, j) = \Delta_i(S_1 f(n, i, j)) + \Delta_j(S_2 f_2(n, i, j))$. As above, we are unable to find it directly with `FindRecurrence`, but we can compute it with a linear combination of certificate recurrences returned by `FindRecurrence`. Unfortunately, the recurrences S_1 and S_2 in the delta parts are too large to compute boundary values. It is a surprise that, by using the function `FindRationalCertificate`, we are able to compute a much simpler representation with rational function certificates: $f(n, i, j) = \Delta_i(R_1 f(n, i, j)) + \Delta_j(R_2 f(n, i, j))$. But note that we shall use `FindRecurrence` in an implicit way: To get suitable denominator polynomials for the rational functions R_1 and R_2, we transform a polynomial certificate recurrence (found by `FindRecurrence`) into a recurrence with rational certificates (by using `CertificateToRational`) and `FindRationalCertificate`

[12] Of course we used a computer algebra system for the tedious simplifications.

returns the simple recurrence given below. Note that this is the only example where we found a recurrence with rational function certificates that is significantly simpler than any certificate recurrence with polynomial recurrence operators that we found.

A WZ–style Proof of (5.12). The summand $f(n,i,j)$ of $f(n)$ satisfies the following recurrence relation[13]

$$(2n+1)\, f(n,i,j) \;=\; \Delta_i(R_1\, f(n,i,j)) \,+\, \Delta_j(R_2\, f(n,i,j))$$

where

$$R_1 \;=\; \frac{i^2\,(1-2\,i-j-3\,i\,j-2\,j^2+5\,n-i\,n+j\,n-4\,i\,j\,n-4\,j^2\,n+6\,n^2+2\,i\,n^2+6\,j\,n^2)}{(1+j)^2\,(1-2\,i+2\,n)}$$
$$R_2 \;=\; \frac{(2\,i-4\,i^2+j-3\,i\,j-n+9\,i\,n-4\,i^2\,n+3\,j\,n-4\,i\,j\,n-2\,n^2+6\,i\,n^2+2\,j\,n^2)}{1-2\,i+2\,n}$$

Summing w.r.t. all i and w.r.t. $j=0,\ldots,n$ we get

$$\begin{aligned}
&(2n+1)\sum_i\sum_{j=0}^{n}\binom{i+j}{i}^2\binom{4n-2i-2j}{2n-2i}\\
&= \sum_i \frac{1-i-4\,i^2+3\,n+2\,i\,n-4\,i^2\,n+3\,n^2+2\,i\,n^2+2\,n^3}{1-2\,i+2\,n}\binom{2n-2i-2}{2n-2i}\binom{n+i+1}{i}^2\\
&\qquad + \sum_{i\ge 0}\frac{-2\,i+4\,i^2+n-9\,i\,n+4\,i^2\,n+2\,n^2-6\,i\,n^2}{1-2\,i+2\,n}\binom{4n-2i}{2n-2i}.
\end{aligned}$$

The last sum is zero, as can be checked with Gosper's algorithm, and the first sum evaluates to $(n+1)^2\binom{2n+1}{n}^2$ (because the summand is zero unless $i=n$). So we get the closed form $f(n) \;=\; (2n+1)\binom{2n}{n}^2$. □

In the editorial comment to the solution of the AMM problem E3376 A. A. Jagers stated the similar looking sum

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\binom{i+j}{j}^2\binom{2n-i-j}{n-j}^2 \;=\; \frac{1}{2}\binom{4n+2}{2n+1} \tag{5.18}$$

which was generalized to

$$\sum_{i=0}^{n}\sum_{j=0}^{m}\binom{i+j}{j}^2\binom{n+m-i-j}{n-j}^2 \;=\; \frac{1}{2}\binom{2m+2n+2}{2n+1} \tag{5.19}$$

by P. Paule.

[13] `FindRationalCertificate[ Binomial[i+j, i]^2 Binomial[4n-2i-2j, 2n-2i], n, 0, {i,j}, { (j+1)^2 (2n-2i+1), (2n-2i+1) }, 3]` in 48 seconds.

Proof of (5.19). The summand of (5.19) is annihilated by[14]

$$(2+m+n)\,(3+2\,m+2\,n)-(1+n)\,(3+2\,n)\,N-$$
$$\Delta_i(-3-5\,i-3\,j-m-2\,i\,m-j\,m-2\,n-3\,i\,n-2\,j\,n+(3-i+3\,j+m+j\,m+2\,n-$$
$$i\,n+2\,j\,n)\,J\,N)-$$
$$\Delta_j(-3+5\,i+3\,j-6\,m+2\,i\,m+j\,m-2\,m^2-5\,n+3\,i\,n+2\,j\,n-4\,m\,n-2\,n^2+$$
$$(1+n)\,(3+2\,n)\,N)$$

Taking the sum w.r.t. j and i with $0\le i\le m$ we observe that the boundary values are zero. Thus we get a homogeneous recurrence relation for the double sum

$$(2+m+n)\,(3+2\,m+2\,n)-(1+n)\,(3+2\,n)\,N$$

which also annihilates the right hand side. Once again the identity follows from checking the case $n=0$. $\square$

Alternative Proof of (5.19). Instead of considering a sum with nonstandard boundary conditions, we can also take $\binom{i+j}{i}\binom{i+j}{j}\binom{n+m-i-j}{n-j}\binom{n+m-i-j}{m-i}$ as summand and get a sum with standard boundary condition. However, we were unable to show that every recurrence for the corresponding proper hypergeometric functions yields a recurrence for this binomial summand: neither a polynomial argument nor a limit argument worked. So we have to check by plugging in whether the binomial summand satisfies the above recurrence at the critical values (these are $i=-1$ and $i=m$) and, indeed, it does. $\square$

5.6 The Sums of John Essam

John Essam ([Ess96]) asked for proofs of the following two identities:

$$\sum_{k_2=0}^{n}\sum_{k_1=0}^{k_2}(k_2-k_1+1)\frac{n!(n+1)!}{k_1!(k_2+1)!(n-k_1+1)!(n-k_2)!} = 4^n\frac{\left(\frac{3}{2}\right)_n}{(2)_n}$$

$$\sum_{k_3=0}^{n}\sum_{k_2=0}^{k_3}\sum_{k_1=0}^{k_2}\frac{(k_2-k_1+1)(k_3-k_1+2)(k_3-k_2+1)\;n!(n+1)!(n+2)!}{k_1!(k_2+1)!(k_3+2)!(n-k_1+2)!(n-k_2+1)!(n-k_3)!} = 8^n\frac{\left(\frac{3}{2}\right)_n}{(3)_n}.$$

Essam stated that he was able to prove the first one using the method of k-free recurrences, but was unable to handle the triple sum. In the following we give proofs of both identities by showing the equivalent forms

$$\sum_{k_1}\sum_{k_2\le k_1}\frac{k_1-k_2}{n}\binom{n}{k_1}\binom{n}{k_2} = 4^{n-1}\frac{\left(\frac{3}{2}\right)_{n-1}}{(2)_{n-1}} \qquad (5.20)$$

$$\sum_{k_1}\sum_{k_2\le k_1}\sum_{k_3\le k_2}\frac{(k_1-k_2)(k_1-k_3)(k_2-k_3)}{n^2(n-1)}\binom{n}{k_1}\binom{n}{k_2}\binom{n}{k_3} = 8^{n-2}\frac{\left(\frac{3}{2}\right)_{n-2}}{(3)_{n-2}}. \qquad (5.21)$$

[14] `FindRecurrence[ Binomial[i+j, j]^2 Binomial[n+m-i-j, n-j]^2, n, 0, {i,j}, {1,0}, 1]` in 4 seconds.

Finding recurrences for the summands is simple: our implementation returns very simple recurrences within a few seconds. The nontrivial parts of the proofs come from the simplification of the boundary values.

Proof of (5.20). In the following let $f(n, k_1, k_2)$ denote the summand of the double sum on the l.h.s. of (5.20), and let $f(n, k_1) = \sum_{k_2 \leq k_1} f(n, k_1, k_2)$ and $f(n) = \sum_{k_1} f(n, k_1)$. A recurrence operator annihilating $f(n, k_1, k_2)$ is easily found:

$$(NK_1K_2 - K_1K_2 - K_1 - K_2 - 1)f(n, k_1, k_2) \; = \; 0.$$

We first sum the recurrence w.r.t. k_2 from $-\infty$ to k_1 and we get

$$f(n+1, k_1+1) - 2f(n, k_1) - 2f(n, k_1+1) \; = \; f(n, k_1, k_1+1) - f(n, k_1+1, k_1+1).$$

The last boundary term $f(n, k_1+1, k_1+1)$ is zero, and summing w.r.t. k_1 yields

$$f(n+1) - 4f(n) \; = \; \sum_{k_1} \frac{-1}{n} \binom{n}{k_1} \binom{n}{k_1+1} \; = \; -\frac{1}{n} \binom{2n}{n+1}.$$

The last equality holds by Vandermonde convolution. Therefore, a recurrence operator annihilating the double sum is

$$((n+2)N - 2(2n+1))\,(N-4)$$

and we can check that it also annihilates the conjectured closed form. Comparing the initial values completes the proof. Note that $4f(n) - f(n+1) = C_n$, the n-th Catalan number. □

Proof of (5.21). Let us denote the summand of the triple sum on the l.h.s. of (5.21) by $f(n, k_1, k_2, k_3)$, and let $f(n, k_1, k_2) = \sum_{k_3 \leq k_2} f(n, k_1, k_2, k_3)$, $f(n, k_1) = \sum_{k_2 \leq k_1} f(n, k_1, k_2)$, and $f(n) = \sum_{k_1} f(n, k_1)$. Again, we easily find a recurrence relation for the summand:

$$(NK_1K_2K_3 - K_1K_2K_3 - K_1K_2 - K_1K_3 - K_2K_3 - K_1 - K_2 - K_3 - 1)\, f(n, k_1, k_2, k_3) \; = \; 0. \tag{5.22}$$

Summing recurrence (5.22) w.r.t. to k_3 (with $k_3 \leq k_2$) yields an inhomogeneous recurrence relation for $f(n, k_1, k_2)$.

$$\begin{aligned}(NK_1K_2 - 2K_1K_2 - 2K_1 - 2K_2 - 2)\, f(n, k_1, k_2) \; &= \; f(n, k_1, k_2, k_2+1) \\ + f(n, k_1+1, k_2, k_2+1) - f(n, k_1, k_2+1, &k_2+1) - f(n, k_1+1, k_2+1, k_2+1)\end{aligned}$$

The last two boundary terms are zero. Summing this recurrence w.r.t. k_2 (with $k_2 \leq k_1$) gives

$$\begin{aligned}(NK_1 - 4K_1 - 4)\, f(n, k_1) \; &= \; \sum_{k_2 \leq k_1} f(n, k_1, k_2, k_2+1) \\ &+ \sum_{k_2 \leq k_1} f(n, k_1+1, k_2, k_2+1) + 2f(n, k_1, k_1+1) - 2f(n, k_1+1, k_1+1).\end{aligned}$$

The last boundary term is again zero, and summing w.r.t. k_1 yields the following inhomogeneous recurrence relation for $f(n)$.

$$\begin{aligned} f(n+1) - 8f(n) &= \sum_{k_1} \sum_{k_2 \le k_1} f(n, k_1, k_2, k_2+1) \\ &+ \sum_{k_1} \sum_{k_2 \le k_1} f(n, k_1+1, k_2, k_2+1) \ + \ 2 \sum_{k_1} \sum_{k_3 \le k_1+1} f(n, k_1, k_1+1, k_3) \end{aligned} \tag{5.23}$$

It is possible to find a closed form for the boundary values. Using the fact that $f(n, k_1, k_2, k_3) = 0$ if $k_1 = k_2$ or $k_1 = k_3$ or $k_2 = k_3$, the boundary values can be rewritten as

$$2 \sum_{k_1} \sum_{k_2 \le k_1} f(n, k_1, k_2, k_2+1) \ + \ 2 \sum_{k_1} \sum_{k_2 \le k_1} f(n, k_1, k_1+1, k_2). \tag{5.24}$$

Interchanging the order of summation in the second sum and renaming the summation variables gives

$$\begin{aligned} 2 \sum_{k_1} \sum_{k_2 \le k_1} f(n, k_1, k_1+1, k_2) &= 2 \sum_{k_2} \sum_{k_1 \ge k_2} f(n, k_1, k_1+1, k_2) \\ &= 2 \sum_{k_1} \sum_{k_2 \ge k_1} f(n, k_2, k_2+1, k_1). \end{aligned}$$

Thus we can, by using the symmetry relation $f(n, k_1, k_2, k_3) = f(n, k_3, k_1, k_2)$, write the two sums (5.24) as one sum with *standard* boundary conditions

$$2 \sum_{k_1} \sum_{k_2} f(n, k_1, k_2, k_2+1) \ = \ \frac{-2}{n^2(n-1)} \sum_{k_1} \sum_{k_2} (k_1-k_2)(k_1-k_2-1) \binom{n}{k_1} \binom{n}{k_2} \binom{n}{k_2+1}. \tag{5.25}$$

The inner sum w.r.t k_1 can be evaluated in closed form

$$\sum_{k_1} (k_1-k_2)(k_1-k_2-1) \binom{n}{k_1} \ = \ 2^{n-2}(4k_2 + 4k_2^2 - 4k_2 n - n + n^2),$$

and the remaining sum w.r.t. k_2, too, can be evaluated in closed form

$$\sum_{k_2} (4k_2 + 4k_2^2 - 4k_2 n - n + n^2) \binom{n}{k_2} \binom{n}{k_2+1} \ = \ \frac{3n(n-1)}{2n-1} \binom{2n}{n+1}.$$

Thus we have found a closed form for the boundary values, and recurrence (5.22) for $f(n)$ now equals

$$f(n+1) - 8f(n) \ = \ \frac{-3\, 2^{n-1}}{n(2n-1)} \binom{2n}{n+1}.$$

A homogeneous recurrence operator annihilating $f(n)$ is therefore

$$((n+2)N - (8n-4))(N-8).$$

Checking that the r.h.s. of (5.21) is annihilated by this recurrence operator and comparing the initial values completes the proof. $\square$

Second Proof of (5.20). The identity (5.20) can also be proved entirely by the computer, using the method explained in Section 3.4. We multiply the summand of the double sum with $\frac{(k_1-k_2+\epsilon)!}{(k_1-k_2)!}$ and get the function

$$f_\epsilon(n,k_1,k_2) \;=\; \frac{(k_1-k_2+\epsilon)!\,n!\,(n-1)!}{(k_1-k_2-1)!\,k_1!\,k_2!\,(n-k_1)!\,(n-k_2)!}.$$

Now the the double sum on the l.h.s. of (5.20) equals $\sum_{k_1}\sum_{k_2}\lim_{\epsilon\to 0} f_\epsilon(n,k_1,k_2)$, which has standard boundary conditions. So we can compute a recurrence for the double sum from the following recurrence[15] that annihilates $f_\epsilon(n,k_1,k_2)$.

$$\begin{aligned}
&8\,n\,(1+2\,n)+(-6+5\,\epsilon+\epsilon^2-14\,n+4\,\epsilon\,n-8\,n^2)\,N+(2+n)\,(1-\epsilon+n)\,N^2\;-\\
&\Delta_{k_1}(n\,(1+n)+(-3-\epsilon+4\,k_1-7\,n)\,n\,K_2-n\,(1+n)\,K_1\,K_2+(-4-\epsilon-8\,k_1-4\,n)\,n\,{K_2}^2+\\
&\quad(-5-4\,k_1-n)\,n\,K_1\,{K_2}^2+(4-\epsilon-2\,k_1-\epsilon\,k_1+7\,n-\epsilon\,n-2\,k_1\,n+3\,n^2)\,K_2\,N+(1+n)\\
&\quad(-2\,k_1+n)\,K_1\,K_2\,N+(2-4\,\epsilon-\epsilon^2+2\,k_1+\epsilon\,k_1+7\,n-3\,\epsilon\,n+2\,k_1\,n+5\,n^2)\,{K_2}^2\,N+\\
&\quad(2-2\,\epsilon-\epsilon\,k_1+8\,n-\epsilon\,n+4\,k_1\,n+2\,n^2)\,K_1\,{K_2}^2\,N+(-1+\epsilon-n)\,(2+n)\,{K_2}^2\,N^2+\\
&\quad(-1+\epsilon-n)\,(2+n)\,K_1\,{K_2}^2\,N^2)\;-\\
&\Delta_{k_2}((-8-\epsilon-16\,n)\,n+(-4-\epsilon-8\,k_1-4\,n)\,n\,K_2+(6-5\,\epsilon-\epsilon^2+14\,n-4\,\epsilon\,n+8\,n^2)\,N+\\
&\quad(2-4\,\epsilon-\epsilon^2+2\,k_1+\epsilon\,k_1+7\,n-3\,\epsilon\,n+2\,k_1\,n+5\,n^2)\,K_2\,N+(-1+\epsilon-n)\,(2+n)\,N^2+\\
&\quad(-1+\epsilon-n)\,(2+n)\,K_2\,N^2).
\end{aligned}$$

The recurrence operator annihilating the double sum is

$$8\,n\,(1+2\,n)+(-6-14\,n-8\,n^2)\,N+(2+n)\,(1+n)\,N^2. \tag{5.26}$$

Again, checking that the r.h.s. is annihilated by (5.26) and comparing initial values completes the proof. □

The computer failed, due to a memory overflow, to find a proof for (5.21) in the same way.

5.7 Miscellaneous Sums

5.7.1 A Sum of Hoon Hong

Recently, Hoon Hong ([Hon96]) needed a closed form evaluation of the following sum

$$\sum_{p,q,P,Q,i,j}(-1)^{i+p+P}\binom{n}{i}\binom{i}{p}\binom{i}{P}\binom{j}{q}\binom{j}{Q} \tag{5.27}$$

[15] `FindRecurrence[(k1-k2+ε)! n! (n-1)! / ((k1 - k2 -1)! k1! k2! (n-k1)! (n-k2)!), n, 1, {k1,k2}, {1,1}, {1,0}]` in 16 seconds.

where $p+q=r$, $P+Q=R$, $p+Q=u$, $q+P=v$, and $i+j=n$, and where r, R, u, v, n are arbitrary integers.

Solving the system of constraints, the six-fold sum (5.27) reduces to the two-fold sum

$$\sum_{i,q}(-1)^{i+r+v}\binom{n}{i}\binom{i}{r-q}\binom{i}{v-q}\binom{n-i}{q}\binom{n-i}{u-r+q}. \tag{5.28}$$

Proof of (5.28). The summand of (5.28) can be written as the proper hypergeometric term

$$\frac{n!\,i!\,(n-i)!}{(r-q)!\,(i-r+q)!\,(v-q)!\,(i-v+q)!\,q!\,(n-i-q)!\,(u-r+Q)!\,(n-i-u+r-q)!}$$

and we see that for the critical values $i<0$ and $i>n$ a limit argument extends every recurrence for the proper hypergeometric function to a recurrence for the binomial summand. The following recurrence operator[16] annihilates the summand of (5.28):

$$(n+1-u-v)N \;+\; \Delta_i((i+2q-r-v)N) \;+\; \Delta_q((n+1-i-2q+r-u)N \;-\; (n+1)).$$

This immediately yields that the sum is zero if $u+v\neq n$.

We set $u=n-v$ and now the summand is annihilated by[17]

$$\begin{aligned}&(n+1)^2-(n+1-r)(n+1-v)N \;+\\ &\Delta_i((i+2q+iq+q^2-r-v-rv)N) \;+\\ &\Delta_Q((n+1)(n+q+2)-(n+2)(r+v-q-n-1)N).\end{aligned}$$

By checking equality at the initial values $n=0$ and for the zeroes of the leading coefficients $n=v$ and $n=r$ we prove the get the closed form evaluation

$$\binom{n}{v}\binom{n}{r}$$

for the sum (5.28) if $u+v=n$. ☐

5.7.2 A Sum of Suzie Dent

The following identity is due to Suzie Dent ([Den96]). The double sum is "an eigenvalue of a certain incidence matrix indexed by partitions". Let v and k be nonnegative integers with $k\geq 2v$, then

$$\sum_{s=0}^{k}\sum_{b\geq 0}(-1)^b\binom{s}{b}\binom{k-s}{2v-b}\binom{k-2v}{s-b} \;=\; \binom{k-v}{k-2v}2^{k-2v}. \tag{5.29}$$

[16] `FindRecurrence[ (-1)`$^{i+r+v}$` Binomial[n, i] Binomial[i, r-q] Binomial[i, v-q] Binomial[n-i, q] Binomial[n-i, u-r+q], n, 0, {i,q}, {0,1}, 1]` in 8 seconds.

[17] `FindRecurrence[ (-1)`$^{i+r+v}$` Binomial[n, i] Binomial[i, r-q] Binomial[i, v-q] Binomial[n-i, q] Binomial[n-i, n-v-r+q], n, 1, {i,q}, {0,1}, {1,2}]` in 83 seconds.

Proof of (5.29). It easily seen that the double sum is a standard sum, i.e., we can sum b and s over all integer values, and that a limit argument extends every recurrence for the proper hypergeometric interpretation of $(-1)^b\binom{s}{b}\binom{k-s}{2v-b}\binom{k-2v}{s-b}$ to a recurrence for the binomial summand itself. The binomial summand is annihilated by[18]

$$\begin{aligned}&2(v-k-1)+(k-2v+1)K-\\&\Delta_b(1+b+k-2v+(b-s-1)KS)\ -\ \Delta_s((b-s)K).\end{aligned}$$

The right hand side of (5.29) is also annihilated by $2(v-k-1)+(k-2v+1)K$, so checking the initial value for $k=2v$ proves the identity. □

5.7.3 A Sum Found in "Concrete Mathematics"

The following sum with five parameters is stated on page 172 of [GKP94]. For integers l, m, n with $n \geq 0$, we have

$$\sum_{j,k}(-1)^{j+k}\binom{j+k}{k+l}\binom{r}{j}\binom{n}{k}\binom{s+n-j-k}{m-j}\ =\ (-1)^l\binom{n+r}{n+l}\binom{s-r}{m-n-l}. \tag{5.30}$$

Proof of (5.30). Call the summand of (5.30) $f(n,j,k)$. The proper hypergeometric term corresponding to f is

$$g(n,j,k)\ =\ (-1)^{j+k}\frac{(j+k)!\,r!\,n!\,(s+n-j-k)!}{(k+l)!\,(j-l)!\,j!\,(r-j)!\,k!\,(n-k)!\,(m-j)!\,(s+n-m-k)!}$$

The only dangerous situation is that for integers j,k we have $j+k<0$. We take limits depending on the sign of l: if l is a nonnegative integer then $\lim_{\epsilon\to 0} g(n,j+\epsilon,k)=f(n,j,k)$ and if l is negative then $\lim_{\epsilon\to 0} g(n,j,k+\epsilon)=f(n,j,k)$. The following recurrence[19] annihilates $g(n,j,k)$ and thus $f(n,j,k)$:

$$\begin{aligned}&(1+n)\,(l-m+n)\,(1+n+r)+(1+n)\,(1+l+n)\,(1+l-m+n-r+s)\,N-\\&\Delta_j((1+n)\,(1+j+n)\,(-2-l+m-2\,n-s)+\\&\ (1+k+j\,k+l+n-j\,n+k\,n+l\,n)\,(-2-l+m-2\,n-s)\,K\,N)-\\&\Delta_k((1+n)\,(1+j-k+j\,l-j\,m+2\,n+j\,n-k\,n+n^2-r-j\,r-k\,r-l\,r+m\,r-n\,r+s+\\&\ j\,s+n\,s)+(1+n)\,(1+l+n)\,(-1-l+m-n+r-s)\,N)\end{aligned}$$

and the resulting recurrence for the sum also annihilates the right hand side. By using a variant of the Vandermonde convolution, we can evaluate the double sum for $n=0$,

$$\sum_j(-1)^j\binom{j}{l}\binom{r}{j}\binom{s-j}{m-j}\ =\ \sum_j(-1)^j\binom{r}{l}\binom{r-l}{j-l}\binom{s-j}{m-j}\ =\ (-1)^l\binom{r}{l}\binom{s-r}{m-l},$$

completing the proof. □

[18] `FindRecurrence[ (-1)`^{b}` Binomial[s, b] Binomial[k-s, 2v-b] Binomial[k-2v, s-b], k, 0, {b,s}, {1,0}, 1]` in 7 seconds.

[19] `FindRecurrence[ (-1)`$^{j+k}$` Binomial[j+k, k+l] Binomial[r, j] Binomial[n, k] Binomial[s+n-j-k, m-j], n, 0, {j,k}, 1, 1]` in 865 seconds.

5.7.4 A Summation Identity of Stechkin

The following identity is due to Stechkin [Ste75]:

$$\sum_{j_1}\sum_{j_2}\binom{j_2-j_1}{q-j_1}\binom{r_2-r_1-j_2+j_1}{p-r_1-q+j_1}\binom{r_1}{j_1}\binom{r_2-r_1}{j_2-j_1}\binom{n-r_2}{l-j_2} = \binom{p}{q}\binom{n-p}{l-q}\binom{r_2-r_1}{p-r_1} \tag{5.31}$$

with nonnegative integers l and q.

Proof of (5.31). It is easily seen that the summand equals

$$\frac{r_1!(r_2-r_1)!(n-r_2)!}{(q-j_1)!(j_2-q)!(p-r_1-q+j_1)!(r_2+q-p-j_2)!j_1!(r_1-j_1)!(l-j_2)!(n-r_2-l+j_2)!},$$

so we do not have to consider singularities. The summand of (5.31) is annihilated by[20]

$$\begin{aligned}&(l-q)\,(p-q)+(-1+l-n+p-q)\,(1+q)\,Q-\\&\Delta_{j_1}(j_1\,(1-j_2-p+q+r_2)\,Q+j_1\,(1+j_2-l+n-r_2)\,J_2\,Q)-\\&\Delta_{j_2}((1+q)\,(j_2-l+n-r_2)\,Q).\end{aligned}$$

The right hand side of (5.31) also satisfies the recurrence for the double sum, so we have to show that the initial values are identical, i.e., that

$$\sum_{j_2}\binom{r_2-r_1-j_2}{p-r_1}\binom{r_2-r_1}{j_2}\binom{n-r_2}{l-j_2} = \binom{n-p}{l}\binom{r_2-r_1}{p-r_1}. \tag{5.32}$$

The summand of (5.32) is annihilated by[21]

$$(-l+n-p)\;-\;(1+l)\,L\;-\;\Delta_{j_2}(j_2\,L),$$

completing the proof. □

5.7.5 A Certain Definite Sum

The following identity was proved in [WZ92a] (and was conjectured by Szondy and Varga)

$$\sum_{j,m}(-1)^{j+n}\frac{(2n-2m+j)!(2k+1+j)!}{j!(n-m)!(i+m)!(m-j)!(n-m+j)!(2k-2i-2m+1+j)!} =$$
$$2^{2n}\frac{(2k+1)!(k-i)!}{(i!(2k-2i+1)!n!(k-i-n)!} \tag{5.33}$$

for $n \in \mathbb{N}_0$.

[20] `FindRecurrence[ Binomial[j2-j1, q-j1] Binomial[r2-r1-j2+j1, p-r1-q+j1] Binomial[r1, j1] Binomial[r2-r1, j2-j1] Binomial[n-r2, l-j2], q, 0, {j1,j2}, {1,1}, 1]` in 60 seconds and the combination of the two linearly independent solution.

[21] `FindRecurrence[ Binomial[r2-r1-j2, p-r1] Binomial[r2-r1, j2] Binomial[n-r2, l-j2], l, 0, j2, 1, 1]` in 2 seconds and the combination of the two linearly independent solutions.

Proof of (5.33). The summand is annihilated by[22]

$$\begin{aligned}
&4(i-k+n)+(1+n)N\\
&\Delta_j(2(1-j+2m-2n)M+(-j+m-n)MN)-\\
&\Delta_m(2(-1-j+2m-2n)+(-1-n)N)
\end{aligned}$$

and the identity follows from checking the initial values. □

5.7.6 A Sum Found in "A=B"

We found the identity

$$\sum_r \sum_s (-1)^{n+r+s} \binom{n}{r}\binom{n}{s}\binom{n+s}{s}\binom{n+r}{r}\binom{2n-r-s}{n} = \sum_k \binom{n}{k}^4 \tag{5.34}$$

on page 33 of [PWZ96]. The computer-generated proof for it turned out to be rather long and was difficult to find.

Proof. We easily see that the singularities are no problem, and that the summand of the double sum is annihilated by[23]

$$\begin{aligned}
&-360\,(1+n)\,(2+n)\,(3+4\,n)\,(5+4\,n)\,(62+147\,n+135\,n^2+58\,n^3+10\,n^4)-\\
&180\,(2+n)\,(3+2\,n)\,(7+9\,n+3\,n^2)\,(62+147\,n+135\,n^2+58\,n^3+10\,n^4)\,N+\\
&90\,(2+n)^4\,(62+147\,n+135\,n^2+58\,n^3+10\,n^4)\,N^2-\\
&\Delta_r S_r-\Delta_s S_s
\end{aligned}$$

where

$S_r = (10\,(-172000-930776\,n-2170952\,n^2-2855018\,n^3-2318064\,n^4-1190900\,n^5-378342\,n^6-68008\,n^7-5300\,n^8+100112\,r+428544\,n\,r+781492\,n^2\,r+786140\,n^3\,r+470726\,n^4\,r+167685\,n^5\,r+32901\,n^6\,r+2744\,n^7\,r+46800\,r^2+176904\,n\,r^2+269892\,n^2\,r^2+211266\,n^3\,r^2+88416\,n^4\,r^2+18342\,n^5\,r^2+1404\,n^6\,r^2+84640\,s+395640\,n\,s+789872\,n^2\,s+872290\,n^3\,s+575146\,n^4\,s+226344\,n^5\,s+49224\,n^6\,s+4564\,n^7\,s-12224\,r\,s-31616\,n\,r\,s-18120\,n^2\,r\,s+18910\,n^3\,r\,s+28619\,n^4\,r\,s+13031\,n^5\,r\,s+2060\,n^6\,r\,s+23400\,r^2\,s+76752\,n\,r^2\,s+96570\,n^2\,r^2\,s+57348\,n^3\,r^2\,s+15534\,n^4\,r^2\,s+1404\,n^5\,r^2\,s)\,N+2\,(-694240-4651336\,n-12274440\,n^2-16952824\,n^3-13368203\,n^4-6007031\,n^5-1383578\,n^6-100548\,n^7+8640\,n^8+281200\,r-5312\,n\,r-2808372\,n^2\,r-6265820\,n^3\,r-6200404\,n^4\,r-3221278\,n^5\,r-857482\,n^6\,r-92340\,n^7\,r+48080\,r^2-19720\,n\,r^2-474612\,n^2\,r^2-887284\,n^3\,r^2-708137\,n^4\,r^2-266039\,n^5\,r^2-38600\,n^6\,r^2+194680\,s+869732\,n\,s+1616744\,n^2\,s+1613631\,n^3\,s+902483\,n^4\,s+250138\,n^5\,s+13212\,n^6\,s-5760\,n^7\,s-206800\,r\,s-1123616\,n\,r\,s-2261768\,n^2\,r\,s-2161596\,n^3\,r\,s-995294\,n^4\,r\,s-178522\,n^5\,r\,s-180\,n^6\,r\,s+42040\,r^2\,s-75700\,n\,r^2\,s-484216\,n^2\,r^2\,s-652659\,n^3\,r^2\,s-356329\,n^4\,r^2\,s-70460\,n^5\,r^2\,s)\,N\,R+180\,(2+n)\,(3660+15164\,n+24765\,n^2+18673\,n^3+4042\,n^4-3171\,n^5-2215\,n^6-414\,n^7+4230\,r+20357\,n\,r+39412\,n^2\,r+$

[22] `FindRecurrence[ (-1)`$^{j+n}$` (2n-2m+j)! (2k+1+j)! / (j! (n-m)! (i+m)! (m-j)! (n-m+j)! (2k-2i-2m+1+j)!), n, 0, {j,m}, {1,1}, 1]` in 2800 seconds.

[23] `FindRecurrence[ (-1)`$^{n+r+s}$` Binomial[n, r] Binomial[n, s] Binomial[n+s, s] Binomial[n+r, r] Binomial[2n-r-s, n], n, 2, {r,s}, {1,1 }, {1,2}]` in 4517 seconds solving a 127×81 system.

$39519\,n^3\,r + 21715\,n^4\,r + 6209\,n^5\,r + 722\,n^6\,r - 970\,r^2 - 3419\,n\,r^2 - 4764\,n^2\,r^2 - 3285\,n^3\,r^2 - 1122\,n^4\,r^2 - 152\,n^5\,r^2 + 1680\,s + 7758\,n\,s + 15157\,n^2\,s + 16048\,n^3\,s + 9703\,n^4\,s + 3170\,n^5\,s + 436\,n^6\,s - 1950\,r\,s - 6231\,n\,r\,s - 7790\,n^2\,r\,s - 4736\,n^3\,r\,s - 1387\,n^4\,r\,s - 154\,n^5\,r\,s - 230\,r^2\,s - 659\,n\,r^2\,s - 694\,n^2\,r^2\,s - 319\,n^3\,r^2\,s - 54\,n^4\,r^2\,s)\,S + (8426800 + 40487696\,n + 84133600\,n^2 + 98717620\,n^3 + 71581755\,n^4 + 32938651\,n^5 + 9454225\,n^6 + 1566287\,n^7 + 116790\,n^8 + 1434440\,r + 6884004\,n\,r + 13898894\,n^2\,r + 15215401\,n^3\,r + 9724138\,n^4\,r + 3635349\,n^5\,r + 745658\,n^6\,r + 67020\,n^7\,r - 969000\,r^2 - 2727572\,n\,r^2 - 2525230\,n^2\,r^2 - 495193\,n^3\,r^2 + 513846\,n^4\,r^2 + 279671\,n^5\,r^2 + 36070\,n^6\,r^2 + 883920\,s + 2732384\,n\,s + 2297156\,n^2\,s - 1443108\,n^3\,s - 3971824\,n^4\,s - 2938202\,n^5\,s - 985130\,n^6\,s - 128300\,n^7\,s + 915120\,r\,s + 3848032\,n\,r\,s + 6563816\,n^2\,r\,s + 5732588\,n^3\,r\,s + 2638812\,n^4\,r\,s + 576188\,n^5\,r\,s + 39700\,n^6\,r\,s - 318080\,r^2\,s - 616120\,n\,r^2\,s + 2732\,n^2\,r^2\,s + 731838\,n^3\,r^2\,s + 557318\,n^4\,r^2\,s + 126880\,n^5\,r^2\,s)\,N\,S + 10\,(2+n)^2\,(8396 + 33368\,n + 52761\,n^2 + 42296\,n^3 + 17914\,n^4 + 3703\,n^5 + 274\,n^6 - 11700\,r - 38376\,n\,r - 48285\,n^2\,r - 28674\,n^3\,r - 7767\,n^4\,r - 702\,n^5\,r - 1116\,s - 3204\,n\,s - 3753\,n^2\,s - 2259\,n^3\,s - 702\,n^4\,s - 90\,n^5\,s)\,N^2\,S + 180\,(2+n)\,(-13100 - 25874\,n + 3000\,n^2 + 44770\,n^3 + 43340\,n^4 + 16627\,n^5 + 2085\,n^6 - 102\,n^7 - 18980\,r - 50388\,n\,r - 43950\,n^2\,r - 7034\,n^3\,r + 9967\,n^4\,r + 5351\,n^5\,r + 766\,n^6\,r - 6600\,r^2 - 20566\,n\,r^2 - 25000\,n^2\,r^2 - 14678\,n^3\,r^2 - 4094\,n^4\,r^2 - 420\,n^5\,r^2 - 11300\,s - 37508\,n\,s - 48899\,n^2\,s - 30697\,n^3\,s - 8648\,n^4\,s - 438\,n^5\,s + 168\,n^6\,s - 7810\,r\,s - 24037\,n\,r\,s - 28720\,n^2\,r\,s - 16448\,n^3\,r\,s - 4413\,n^4\,r\,s - 422\,n^5\,r\,s - 230\,r^2\,s - 659\,n\,r^2\,s - 694\,n^2\,r^2\,s - 319\,n^3\,r^2\,s - 54\,n^4\,r^2\,s)\,R\,S + (1172460 + 8293288\,n + 22882409\,n^2 + 32565731\,n^3 + 25834446\,n^4 + 11056379\,n^5 + 1963717\,n^6 - 136494\,n^7 - 71640\,n^8 - 476400\,r - 100468\,n\,r + 4947316\,n^2\,r + 12028321\,n^3\,r + 12866679\,n^4\,r + 7308469\,n^5\,r + 2172893\,n^6\,r + 269190\,n^7\,r + 30180\,r^2 + 338260\,n\,r^2 + 1000043\,n^2\,r^2 + 1245074\,n^3\,r^2 + 723213\,n^4\,r^2 + 178394\,n^5\,r^2 + 10780\,n^6\,r^2 - 950060\,s - 4293340\,n\,s - 7847041\,n^2\,s - 7310382\,n^3\,s - 3455233\,n^4\,s - 569432\,n^5\,s + 125568\,n^6\,s + 45000\,n^7\,s + 48920\,r\,s + 87664\,n\,r\,s - 249974\,n^2\,r\,s - 940458\,n^3\,r\,s - 1119074\,n^4\,r\,s - 583402\,n^5\,r\,s - 113580\,n^6\,r\,s + 98260\,r^2\,s + 635996\,n\,r^2\,s + 1428071\,n^2\,r^2\,s + 1469856\,n^3\,r^2\,s + 713351\,n^4\,r^2\,s + 132730\,n^5\,r^2\,s)\,N\,R\,S + 9\,(2+n)\,(-14180 - 50096\,n - 78779\,n^2 - 76639\,n^3 - 50827\,n^4 - 21073\,n^5 - 4198\,n^6 - 160\,n^7 - 32100\,r - 123504\,n\,r - 201275\,n^2\,r - 182287\,n^3\,r - 99355\,n^4\,r - 31393\,n^5\,r - 4470\,n^6\,r + 27060\,s + 107124\,n\,s + 173179\,n^2\,s + 142034\,n^3\,s + 58769\,n^4\,s + 9770\,n^5\,s + 27060\,r\,s + 107124\,n\,r\,s + 173179\,n^2\,r\,s + 142034\,n^3\,r\,s + 58769\,n^4\,r\,s + 9770\,n^5\,r\,s)\,N^2\,R\,S + 180\,(2 + n)\,(-3480 - 12684\,n - 16102\,n^2 - 4195\,n^3 + 9671\,n^4 + 10829\,n^5 + 4595\,n^6 + 734\,n^7 - 2280\,r - 14126\,n\,r - 31622\,n^2\,r - 34783\,n^3\,r - 20328\,n^4\,r - 6055\,n^5\,r - 722\,n^6\,r + 1200\,r^2 + 4078\,n\,r^2 + 5458\,n^2\,r^2 + 3604\,n^3\,r^2 + 1176\,n^4\,r^2 + 152\,n^5\,r^2 - 1680\,s - 7758\,n\,s - 15157\,n^2\,s - 16048\,n^3\,s - 9703\,n^4\,s - 3170\,n^5\,s - 436\,n^6\,s + 1950\,r\,s + 6231\,n\,r\,s + 7790\,n^2\,r\,s + 4736\,n^3\,r\,s + 1387\,n^4\,r\,s + 154\,n^5\,r\,s + 230\,r^2\,s + 659\,n\,r^2\,s + 694\,n^2\,r^2\,s + 319\,n^3\,r^2\,s + 54\,n^4\,r^2\,s)\,S^2 + (-792560 - 2451416\,n - 2050124\,n^2 + 473354\,n^3 + 808028\,n^4 - 974976\,n^5 - 1457588\,n^6 - 637786\,n^7 - 96100\,n^8 + 926520\,r + 4002528\,n\,r + 7461826\,n^2\,r + 7575890\,n^3\,r + 4253420\,n^4\,r + 1109772\,n^5\,r + 10364\,n^6\,r - 37040\,n^7\,r + 361880\,r^2 + 952608\,n\,r^2 + 846338\,n^2\,r^2 + 359038\,n^3\,r^2 + 233548\,n^4\,r^2 + 182040\,n^5\,r^2 + 50420\,n^6\,r^2 + 1664120\,s + 9325636\,n\,s + 22440698\,n^2\,s + 29848565\,n^3\,s + 23601975\,n^4\,s + 11068439\,n^5\,s + 2846777\,n^6\,s + 309550\,n^7\,s + 539660\,r\,s + 2475196\,n\,r\,s + 4952497\,n^2\,r\,s + 5484282\,n^3\,r\,s + 3502537\,n^4\,r\,s + 1209296\,n^5\,r\,s + 174780\,n^6\,r\,s - 98260\,r^2\,s - 635996\,n\,r^2\,s - 1428071\,n^2\,r^2\,s - 1469856\,n^3\,r^2\,s - 713351\,n^4\,r^2\,s - 132730\,n^5\,r^2\,s)\,N\,S^2 + 10\,(2+n)^2\,(-9512 - 37688\,n - 59718\,n^2 - 48308\,n^3 - 20875\,n^4 - 4495\,n^5 - 364\,n^6 + 11700\,r + 38376\,n\,r + 48285\,n^2\,r + 28674\,n^3\,r + 7767\,n^4\,r + 702\,n^5\,r + 1116\,s + 3204\,n\,s + 3753\,n^2\,s + 2259\,n^3\,s + 702\,n^4\,s + 90\,n^5\,s)\,N^2\,S^2 + 720\,(2+n)\,(1570 + 4907\,n + 5725\,n^2 + 2755\,n^3 + 130\,n^4 - 321\,n^5 - 78\,n^6 + 1210\,r + 3701\,n\,r + 4379\,n^2\,r + 2464\,n^3\,r + 638\,n^4\,r + 56\,n^5\,r)\,(2 - 2\,n + r + s)\,R\,S^2 + 36\,(1+n)\,(2+n)\,(-3200 - 16860\,n - 29381\,n^2 - 19378\,n^3 - 501\,n^4 + 4492\,n^5 + 1340\,n^6 + 6070\,r + 17093\,n\,r + 16034\,n^2\,r + 4418\,n^3\,r - 1231\,n^4\,r - 610\,n^5\,r - 1150\,r^2 - 2145\,n\,r^2 - 1325\,n^2\,r^2 - 270\,n^3\,r^2 + 970\,s + 933\,n\,s - 3181\,n^2\,s - 6187\,n^3\,s - 3871\,n^4\,s - 830\,n^5\,s)\,N\,R\,S^2 + 9\,(2+n)\,(-39800 - 160240\,n - 260442\,n^2 - 206830\,n^3 - 72650\,n^4 - 1610\,n^5 + 4612\,n^6 + 560\,n^7 - 28120\,r - 115968\,n\,r - 196418\,n^2\,r - 166698\,n^3\,r - 68418\,n^4\,r - 9350\,n^5\,r + 780\,n^6\,r - 27060\,s - 107124\,n\,s - 173179\,n^2\,s - 142034\,n^3\,s - 58769\,n^4\,s - 9770\,n^5\,s - 27060\,r\,s - 107124\,n\,r\,s - 173179\,n^2\,r\,s - 142034\,n^3\,r\,s - 58769\,n^4\,r\,s - 9770\,n^5\,r\,s)\,N^2\,R\,S^2)$

and

$S_s = (360\,(1+n)\,(2+n)\,(3+4\,n)\,(5+4\,n)\,(62+147\,n+135\,n^2+58\,n^3+10\,n^4)+10\,(218872+1156856\,n+2651570\,n^2+3444050\,n^3+2773896\,n^4+1419266\,n^5+450738\,n^6+81292\,n^7+6380\,n^8-100112\,r-428544\,n\,r-781492\,n^2\,r-786140\,n^3\,r-470726\,n^4\,r-167685\,n^5\,r-32901\,n^6\,r-2744\,n^7\,r-46800\,r^2-176904\,n\,r^2-269892\,n^2\,r^2-211266\,n^3\,r^2-88416\,n^4\,r^2-18342\,n^5\,r^2-1404\,n^6\,r^2-84640\,s-395640\,n\,s-789872\,n^2\,s-872290\,n^3\,s-575146\,n^4\,s-226344\,n^5\,s-49224\,n^6\,s-4564\,n^7\,s+12224\,r\,s+31616\,n\,r\,s+18120\,n^2\,r\,s-18910\,n^3\,r\,s-28619\,n^4\,r\,s-13031\,n^5\,r\,s-2060\,n^6\,r\,s-23400\,r^2\,s-76752\,n\,r^2\,s-96570\,n^2\,r^2\,s-57348\,n^3\,r^2\,s-15534\,n^4\,r^2\,s-1404\,n^5\,r^2\,s)\,N-90\,(2+n)^4\,(62+147\,n+135\,n^2+58\,n^3+10\,n^4)\,N^2+180\,(2+n)\,(-1800-4926\,n-945\,n^2+11853\,n^3+19374\,n^4+13999\,n^5+5031\,n^6+734\,n^7-4230\,r-20357\,n\,r-39412\,n^2\,r-39519\,n^3\,r-21715\,n^4\,r-6209\,n^5\,r-722\,n^6\,r+970\,r^2+3419\,n\,r^2+4764\,n^2\,r^2+3285\,n^3\,r^2+1122\,n^4\,r^2+152\,n^5\,r^2-1680\,s-7758\,n\,s-15157\,n^2\,s-16048\,n^3\,s-9703\,n^4\,s-3170\,n^5\,s-436\,n^6\,s+1950\,r\,s+6231\,n\,r\,s+7790\,n^2\,r\,s+4736\,n^3\,r\,s+1387\,n^4\,r\,s+154\,n^5\,r\,s+230\,r^2\,s+659\,n\,r^2\,s+694\,n^2\,r^2\,s+319\,n^3\,r^2\,s+54\,n^4\,r^2\,s)\,S+10\,(-373144-1806936\,n-3756110\,n^2-4370746\,n^3-3107581\,n^4-1379061\,n^5-372054\,n^6-55666\,n^7-3534\,n^8-23480\,r-116540\,n\,r-246714\,n^2\,r-287123\,n^3\,r-198538\,n^4\,r-82153\,n^5\,r-19108\,n^6\,r-1968\,n^7\,r+46800\,r^2+165204\,n\,r^2+231516\,n^2\,r^2+162981\,n^3\,r^2+59742\,n^4\,r^2+10575\,n^5\,r^2+702\,n^6\,r^2-77848\,s-328320\,n\,s-571898\,n^2\,s-525640\,n^3\,s-267466\,n^4\,s-70257\,n^5\,s-6771\,n^6\,s+248\,n^7\,s-1712\,r\,s-7928\,n\,r\,s-8940\,n^2\,r\,s+2872\,n^3\,r\,s+11294\,n^4\,r\,s+7118\,n^5\,r\,s+1448\,n^6\,r\,s+23400\,r^2\,s+76752\,n\,r^2\,s+96570\,n^2\,r^2\,s+57348\,n^3\,r^2\,s+15534\,n^4\,r^2\,s+1404\,n^5\,r^2\,s)\,N\,S+10\,(2+n)^2\,(-10628-40892\,n-63471\,n^2-50567\,n^3-21577\,n^4-4585\,n^5-364\,n^6+11700\,r+38376\,n\,r+48285\,n^2\,r+28674\,n^3\,r+7767\,n^4\,r+702\,n^5\,r+1116\,s+3204\,n\,s+3753\,n^2\,s+2259\,n^3\,s+702\,n^4\,s+90\,n^5\,s)\,N^2\,S).$

Finding a recurrence annihilating $\binom{n}{k}^4$ is simple:

$$4\,(-3-4\,n)\,(1+n)\,(5+4\,n)-2\,(3+2\,n)\,(7+9\,n+3\,n^2)\,N+(2+n)^3\,N^2-$$
$$\Delta_k((1+n)\,(50+10\,k+101\,n+8\,k\,n+49\,n^2)+(1+n)\,(20+10\,k+25\,n+8\,k\,n+7\,n^2)\,K+$$
$$2\,(3+2\,n)\,(7+9\,n+3\,n^2)\,N+(44+46\,k+51\,n+66\,k\,n+6\,n^2+24\,k\,n^2-6\,n^3)\,K\,N-$$
$$(2+n)^3\,N^2-(2+n)^3\,K\,N^2).$$

So both sums satisfy the same recurrence relation

$$4\,(-3-4\,n)\,(1+n)\,(5+4\,n)-2\,(3+2\,n)\,(7+9\,n+3\,n^2)\,N+(2+n)^3\,N^2.$$

It remains to check the initial values: For $n=0$ both sums are 1, for $n=1$ they are 2. $\square$

References

[Aic97] E. Aichinger. A linear system solver for Mathematica that uses SACLIB (Version 1.0). Technical Report 97-05, RISC-Linz, J. Kepler University Linz, Austria, February 1997.

[AP92] G. E. Andrews and P. Paule. A higher degree binomial coefficient identity. *Amer. Math. Monthly*, 99:64, 1992.

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

[C$^+$93] G. Collins et al. Saclib 1.1 user's guide. Technical Report 93-19, RISC-Linz, J. Kepler University Linz, Austria, March 1993.

[Car64] L. Carlitz. A binomial identity arising from a sorting problem. *SIAM Review*, 6:20–30, 1964.

[Car68] L. Carlitz. Summations of products of binomial coefficients. *Amer. Math. Monthly*, 75(8):906–908, 1968.

[Chy94] F. Chyzak. Holonomic systems and automatic proofs of identities. Research Report 2371, Institut National de Recherche en Informatique et en Automatique, October 1994.

[CS97] F. Chyzak and B. Salvy. Non-commutative elimination in Ore algebras proves multivariate holonomic identities. *Journal of Symbolic Computation*, 1997. Submitted.

[Den96] S. Dent. letter to P. Paule, August 1996.

[Ess96] J. Essam. letter to P. Paule, November 1996.

[Fas47] M. C. Fasenmyer. Some generalized hypergeometric polynomials. *Bull. Amer. Math. Soc.*, 54:806–812, 1947.

[Fas49] M. C. Fasenmyer. A note on pure recurrence relations. *Amer. Math. Monthly*, 56:14–17, 1949.

[GKP94] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics*. Addison-Wesley, Reading, Massachusetts, second edition, 1994.

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

[Hon96] H. Hong. personal communication to P. Paule, December 1996.

[Hor92] J. Hornegger. *Hypergeometrische Summation und polynomiale Rekursion.* Diplomarbeit, Universität Erlangen-Nürnberg, 1992.

[Pau92] P. Paule. Solution of a seminaire homework example (28th slc). Technical Report 92-59, RISC-Linz, 1992.

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

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

[Rai60] E. Rainville. *Special Functions.* The Macmillan Company, New York, 1960.

[Sch95] M. Schorn. *Contributions to Symbolic Summation.* Diplomarbeit, RISC, Johannes Kepler Universität Linz, 1995.

[Ste75] B. S. Stechkin. Generalized valences. *Math. Notes*, 17:252–258, 1975.

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

[vdP78] A. van der Poorten. A proof that Euler missed ... Apéry's proof of the irrationality of $\zeta(3)$. *Math. Intelligencer*, 1:195–203, 1978.

[Ver74] P. Verbaeten. The automatic construction of pure recurrence relations. In *ACM-SIGSAM Bulletin*, volume 8, pages 96–98. EUROSAM, 1974.

[Ver76] P. Verbaeten. *Rekursiebetrekkingen voor lineaire hypergeometrische funkties.* Proefschrift voor het doctoraat in de toegepaste wetenschapen, Katholieke Universiteit te Leuven, Heverlee, Belgium, 1976.

[Wil91] H. S. Wilf. Sums of closed form functions satisfy recurrence relations. unpublished, available at `http://www.cis.upenn.edu/~wilf/`, 1991.

[Wol93] Wolfram Research. *MathLink Reference Guide, Version 2.2*, 1993.

[WZ92a] H. S. Wilf and D. Zeilberger. An algorithmic proof theory for hypergeometric (ordinary and "q") multisum/integral identities. *J. Comp. Appl. Math.*, 32:321–368, 1992.

[WZ92b] H. S. Wilf and D. Zeilberger. Rational function certification of multisum/integral/"q" identities. *Bull. Amer. Math. Soc. (New Series)*, 27:148–153, 1992.

[Yen93] L. Yen. *Contributions to the proof theory of hypergeometric identities.* PhD thesis, University of Pennsylvania, 1993.

[Zei90a] D. Zeilberger. A fast algorithm for proving terminating hypergeometric identities. *Discr. Math.*, 80:207–211, 1990.

[Zei90b] D. Zeilberger. A holonomic system approach to special functions identities. *Journal of Computational and Applied Math.*, 32:321–368, 1990.

Index