J. Symbolic Computation (1994) **11**, 1–000

A Mathematica Version of Zeilberger's Algorithm for Proving Binomial Coefficient Identities

PETER PAULE† AND MARKUS SCHORN†

†*Research Institute for Symbolic Computation, J. Kepler University, A–4040 Linz, Austria*

(Received 21 March 1994)

Based on Gosper's algorithm for *indefinite* hypergeometric summation, Zeilberger's algorithm for proving binomial coefficient identities constitutes a recent breakthrough in symbolic computation. *Mathematica* implementations of these algorithms are described. Nontrivial examples are given in order to illustrate the usage of these packages which are available by email-request to the first-named author.

1. Introduction

Gosper's algorithm for *indefinite* hypergeometric summation, see e.g. Gosper (1978) or Lafon (1983) or Graham, Knuth and Patashnik (1989), belongs to the standard methods implemented in most computer algebra systems. Exceptions are, for instance, the 2.x-Versions of the *Mathematica* system where symbolic summation is done by different means. A brief discussion is given in section 5. Current interest in Gosper's algorithm is mainly due to the fact, observed by Zeilberger (1990a, 1990b), that it also can be used in a non-obvious and nontrivial way for *definite* hypergeometric summation. For instance, for verifying or finding binomial identities "automatically", finding recurrence operators annihilating hypergeometric sums etc. A generalization of that approach can be found in Wilf and Zeilberger (1992), which is also an excellent source for examples and further references. *Maple* versions of Zeilberger's algorithm have been written by Zeilberger (1991) and Koornwinder (1993). The authors of this paper implemented the algorithms of Gosper and Zeilberger in the *Mathematica* system (see also Paule and Schorn (1993)). It is the goal of this paper to introduce these implementations to potential users. A special emphasis is put on illustrating how to apply the packages to concrete and nontrivial problems, see section 4. In section 2 the algorithmic background is briefly discussed. Section 3 explains the usage of the packages such as installation, features etc.

2. Theoretical Background

The underlying ideas of the Gosper and of the Zeilberger algorithm are well documented, see the literature cited. Thus in describing the theoretical background we restrict ourselves to a brief sketch of the fundamentals. This section also contains some remarks concerning particular features of the implementation. In order to get a more complete

0747–7171/90/000000 + 00 $03.00/0

picture of what is peculiar or new in our packages, the reader is referred to the detailed feature description given in the next section.

The basic mechanism of Zeilberger's "fast" algorithm for proving binomial coefficient identities can be described as follows. Let the double-indexed sequence $(f_{n,k})_{n,k}$ be hypergeometric in both indices, i.e. where the quotients $f_{n+1,k}/f_{n,k}$ and $f_{n,k+1}/f_{n,k}$ are rational functions in n and k over same suitable (e.g. computable) ground-field containing the rational numbers. Then, under mild conditions, one can guarantee the existence of a linear recurrence

$$p_d(n)f_{n+d,k} + p_{d-1}(n)f_{n+d-1,k} + \cdots + p_0(n)f_{n,k} = g_{n,k+1} - g_{n,k}$$

where the coefficients are polynomials in n, and where $(g_{n,k})$ is hypergeometric in k (and n). Given the order d, the coefficient polynomials as well as the hypergeometric sequence $g_{n,k}$ are manufactured by Gosper's algorithm. In general, d is not known in advance but an upper bound can be computed. For proofs and more details of the method see e.g. Zeilberger (1990*b*) or Wilf/Zeilberger (1992). For tutorials see Zeilberger (1993), a reprinted version is contained in this issue, or Wilf (1993). The latter also discusses the historical roots of the method referring to Sister Celine's technique, see e.g. Fasenmyer (1949), or to the work of Verbaeten, see e.g. Verbaeten (1974). Now summing both sides of the recurrence above, for instance, for k from a to b, $a \leq b$, one obtains by observing that the right hand sum is a telescoping one:

$$(p_d(n)N^d + p_{d-1}(n)N^{d-1} + \cdots + p_0(n)I)\sum_{k=a}^{b} f_{n,k} = g_{n,b+1} - g_{n,a}$$

where N is the shift-operator with respect to n, i.e. $Nf_{n,k} = f_{n+1,k}$. It might also be that a or b depend in same way on n, see section 3.3.3. In this case, by introducing corresponding correction terms, one has to modify the inhomogeneous part, in order to keep the left hand side of the equation in the form as stated above. This is done automatically by our package. But in many instances the summand $f_{n,k}$ induces summation bounds in a natural way. For instance, if $f_{n,k} = \binom{n}{k}x^k$ then $f_{n,k} = 0$ outside the integer interval $0 \leq k \leq n$. Therefore the finite support of the summand sequence is described by the summation bounds 0 and n. Now, summing for bounds $a \leq -1$ and $b \geq n+1$ results in a *homogeneous* recursion for the sum, i.e. $g_{n,b+1} - g_{n,a} = 0$, which is true for many sums arising in applications. The linear recurrence can be taken as the certificate for the sum. As demonstrated in section 4, there are manifold ways to make concrete use of it. Two straightforward applications are proving and finding. For instance, the certificate of $S_n := \sum_{k=0}^{n}\binom{n}{k}x^k$ is $(1+x)S_n - S_{n+1} = 0$. Solving, over a suitable domain, this recurrence of order 1 with initial value $S_0 = 1$ is equivalent to *finding* a closed form evaluation of the sum representation of S_n. On the other hand, one could *prove* the binomial theorem by checking that $(1+x)^n$ is the solution of the recurrence found by the algorithm. Note that, given the closed evaluation as a hypergeometric term, also this check in general can be done by computer. This is true, because the problem can be reduced to checking a *rational* function to be zero. The reduction is an immediate consequence of using the rational certificate, in this example $S_{n+1}/S_n = (1+x)$.

Another application would be to decide whether two sums over hypergeometric sequences represent the same sequence. This is also related to the question whether the recurrence can be used as a canonical form, see section 4.3.

The main ingredient in Zeilberger's algorithm is Gosper's algorithm for indefinite hypergeometric summation. One peculiarity of our implementation is that it computes Petkovšek's canonical form of Gosper's representation of a rational function, see Petkovšek (1992). We also want to point out that due to the special structure of the input, see section 3, which delivers information essentially in already factored form, this form is computed without any resultant computation. In more general contexts this procedure is unavoidable, see also section 5.

Besides the canonical form computation, the main part of Gosper's algorithm consists in solving a system of homogeneous linear equations with polynomial (resp. rational function) coefficients. With respect to this problem we want to point out that the built-in *Mathematica* functions, such as `Solve` or `NullSpace`, turned out to be much too slow even for modest applications. Following Knuth's lines, see pages 425-427 in Knuth (1969), K. Eichhorn wrote an algorithm, in *Mathematica* code, which runs satisfactory on a huge number of examples, and which is used in the present version.

3. Usage of the Packages

3.1. How to get it

Our packages are available by email request to the first named author:

`Peter.Paule@risc.uni-linz.ac.at`

Besides three files containing the algorithms in *Mathematica* code and a `read.me` file, you will also get a file containing all *Mathematica* examples presented in this paper, i.e. following the paper on your computer you don't have to type in.

3.2. Installation

Name each of the three files as indicated in their top lines and place them in the directory where you run *Mathematica*. Start a *Mathematica* session and load our implementation with the *Mathematica* command `<< zb_alg.m`. All the other files are loaded automatically.

If you want to use the examples of the provided example file, you have to load it separately. Make sure that `zb_alg.m` is already loaded, and use the *Mathematica* command `<< examples.m` to make the examples available.

3.3. Solving Problems

Our package has two interfaces. Given a hypergeometric sum, you can run an extended version of Gosper's algorithm to find a closed form for a sum. With an extended version of Zeilberger's fast algorithm you can try to come up with a recurrence for a sum.

1 `Gosper[ SUMMAND, RANGE, ORDER]`

2 `Zb[ SUMMAND, RANGE, RECVAR, ORDER]`

The interface to Gosper's algorithm can be used without supplying an `ORDER`. This will run the standard algorithm of Gosper, i.e. where `ORDER` $= 0$.

3.3.1. Illustrating Examples

1 First we load the package:

```
In[1]:= <<zb_alg.m
Out[1]= Peter Paule and Markus Schorn's implementation loaded.
        (Version 1.2)
```

2 An easy example:

```
In[2]:= Gosper[k^2, {k, 0, n-1}]
                2                      (-1 + n) n (-1 + 2 n)
Out[2]= {SUM[k , {k, 0, -1 + n}] == ---------------------}
                                                6
```

3 This demonstrates the feature of the extended version of Gosper's algorithm. Note that since we did not specify the bounds, the anti-difference is calculated, not the sum. This is indicated by using `delta`:

```
In[3]:= Gosper[k!, k]
Out[3]= Try higher order

In[4]:= Gosper[k!, k, 1]
Out[4]= {SUM[k k!, {delta, k}] == k!}
```

4 A classic example for Zeilberger's algorithm. Since the bounds are naturally induced, see 3.3.3, we do not specify them explicitly.

```
In[5]:= Zb[Binomial[n,k] x^k y^(n-k), k, n, 1]
Out[5]= {(x + y) SUM[n] - SUM[1 + n] == 0}
```

5 An example for Zeilberger's algorithm where you have to specify at least the lower bound.

```
In[6]:= Zb[1/(2k+1) Binomial[2n+1, n-k], {k, 0, n}, n, 1]
                                                      (2 + 2 n)!
Out[6]= {8 (1 + n) SUM[n] + (-3 - 2 n) SUM[1 + n] == -(----------)}
                                                             2
                                                      (1 + n)!
```

3.3.2. The input structure of the summand

$$\begin{aligned}
\texttt{SUMMAND} \quad &:= \quad <\mathit{hypGeomTerm}>. \\
<\mathit{hypGeomTerm}> \quad &:= \quad <\mathit{simpleHypGeomTerm}> \\
\mathit{or} \quad & \quad <\mathit{hypGeomTerm}> \wedge <\mathit{integer}> \\
\mathit{or} \quad & \quad <\mathit{hypGeomTerm}> * <\mathit{hypGeomTerm}>.
\end{aligned}$$

$$
\begin{array}{rcl}
<\mathit{simpleHypGeomTerm}> & := & <\mathit{ratFun}> \\
 & \mathit{or} & <\mathit{factorial}> \\
 & \mathit{or} & <\mathit{binomial}> \\
 & \mathit{or} & <\mathit{expFun}>. \\
<\mathit{factorial}> & := & <\mathit{intLinPoly}>!. \\
<\mathit{binomial}> & := & \texttt{Binomial}[<\mathit{intLinPoly}>,<\mathit{intLinPoly}>]. \\
<\mathit{expFun}> & := & <\mathit{constRatFun}> \wedge <\mathit{intLinRatFun}>. \\
<\mathit{intLinRatFun}> & := & <\mathit{integer}> * \texttt{SUMVAR} + <\mathit{integer}> * \texttt{RECVAR} + \\
 & & <\mathit{constRatFun}>. \\
<\mathit{intLinPoly}> & := & <\mathit{integer}> * \texttt{SUMVAR} + <\mathit{integer}> * \texttt{RECVAR} + \\
 & & <\mathit{constPoly}>. \\
<\mathit{constRatFun}> & := & <\mathit{constPoly}>/<\mathit{constPoly}>. \\
<\mathit{constPoly}> & := & <\mathit{poly}>,\ \mathit{free\ of}\ \texttt{SUMVAR}\ \mathit{and}\ \texttt{RECVAR}. \\
<\mathit{ratFun}> & := & <\mathit{poly}>/<\mathit{poly}>. \\
<\mathit{poly}> & := & \mathit{any\ polynomial\ you\ can\ type\ in\ Mathematica.}
\end{array}
$$

The algorithm is guaranteed to run on any input where $<\mathit{ratFun}>$ is a polynomial. Additionally, we accept an input if the irreducible factors of the denominator of $<\mathit{ratFun}>$ are integer-linear in `SUMVAR`.

3.3.3. The specification of the summation range

$$
\begin{array}{rcl}
\texttt{RANGE} & := & \texttt{SUMVAR} \\
 & \mathit{or} & \{\texttt{SUMVAR}, <\mathit{bound}>, <\mathit{bound}>\}. \\
<\mathit{bound}> & := & \pm\texttt{Infinity} \\
 & \mathit{or} & <\mathit{ratFun}>,\ \mathit{free\ of}\ \texttt{SUMVAR}.
\end{array}
$$

If one specifies a bound to be `Infinity` (`-Infinity`) then this bound is assumed to be *naturally induced*. An upper (lower) bound is naturally induced if there is some integer k_0, such that the summand evaluates to 0 for all summation indices greater (less) than this k_0.

Specifying no bounds has a different meaning depending on the algorithm used. In Zeilberger's algorithm we will assume that both bounds are naturally induced (see example 4).

In Gosper's algorithm, however, this is not reasonable. Therefore, if no bounds are specified in a `Gosper` call, the anti-difference is computed (see example 3).

3.4. Tuning of the algorithms

The global *Mathematica* variable `RunMode` can be set to different values.

`Fast` The algorithm computes what is needed to obtain a result. You might not be able to obtain a proof certificate using this mode.

`Provable` Additionally some intermediate results are computed and stored so that you may obtain a proof certificate after running the algorithm. In this mode the algorithms use more storage and are slower, too.
`Proving` After each run of an algorithm a proof certificate is generated.

If you specify the `ORDER` argument higher than actually necessary, usually there exist more than one solution. The global *Mathematica* variable `SolAmount` indicates how many of these solutions are to be computed.

3.5. Obtaining a proof certificate

As described above, running the algorithm setting `RunMode = Proving`, the algorithm will append a proof certificate to the file specified in the *Mathematica* variable `FileName`.

Alternatively you can run the algorithm with the setting `RunMode = Provable`, in order to obtain the proof certificate by explicitly typing `Prove` after you ran the algorithm on your problem.

3.6. What is new?

Special features of the package are:

1. The input structure, which is close to *Mathematica* syntax.
2. The extension to Gosper's algorithm that looks for a polynomial factor that makes your input sum Gosper summable.
3. The possibility to specify bounds in Zeilberger's algorithm.
4. All exceptions to a result are found. The proof certificate contains a correct proof which mentions these exceptions explicitly. A simple example is to ask for the Gosper summation of $\sum_{k=0}^{n}(-1)^k\binom{a}{k}$. (Note the case $a = 0$.)

Some shortcomings are:

1. Binomials $\binom{n}{k}$ are interpreted as abbreviation for the well known factorial representation $\frac{n!}{k!(n-k)!}$. In some applications one might need to interpret the factorial expressions in terms of the Gamma function, for instance, to get $1/n! = 0$ for negative integer n.
2. Rising or falling factorials are not provided as input types. But for a user preferring input types like those it is easy to overcome this problem by writing appropriate *Mathematica* rules.
3. The results of the algorithm need additional simplification, sometimes.

It is our hope that all these shortcomings will be removed in a new release which is planned to be available in summer 1995.

3.6.1. Extension of Gosper's algorithm

If a sum is not Gosper summable you can try to find a polynomial factor which makes this sum Gosper summable. The package does this for you. You have to supply the maximum degree of the polynomial the algorithm looks for. For applications see section 4.1.

3.6.2. Evaluation model for hypergeometric terms

The algorithm dealing with bounds and finding the exceptions to the recurrences is based on an evaluation model established by the authors. This model serves as the basis of a correctness proof for this implementation. A detailed description is planned for the 1995 release.

4. Examples

In this section we present nontrivial examples illustrating several features of our package as well as the wide range of its applicability. For each session we assume that the `zb_alg.m` package has been loaded as explained in (3.3.1).

4.1. Make it Gosper-summable

Graham, Knuth and Patashnik (1989) discussing a sum-analogue to the problem of integrating $x\ e^{-x^2}$ or e^{-x^2}, respectively, come across the identity (eq. (5.18), ibid.)

$$\sum_{k=0}^{m} \binom{r}{k}(\frac{r}{2} - k) = \frac{m+1}{2}\binom{r}{m+1}.$$

After trying

```
In[7]:= Gosper[Binomial[r,k], {k,0,m}]
Out[7]= Try higher order
```

the identity above can be obtained as follows:

```
In[8]:= Gosper[Binomial[r,k], {k,0,m}, 1]
             (-2 k + r) r!                        r!
Out[8]= {SUM[-------------, {k, 0, m}] == ----------------}
             k! (-k + r)!                   m! (-1 - m + r)!
```

Calling the Gosper procedure with "order" 1 is equivalent to multiplying a polynomial factor of degree 1 in undetermined coefficients to the given hypergeometric summand. In case there exist coefficients which turns the sum into a Gosper-summable one, those are computed by Gosper's algorithm and the answer is returned as in the example above. The same applies for choosing higher "orders".

This way homework exercise 5.57 (ibid.) is solved as:

```
In[9]:= Gosper[Binomial[n,k] z^k,k,1]
               k                                          k
              z  (-k - k z + n z) n!                     z  n!
Out[ 9]= {SUM[----------------------, {delta, k}] == -------------------}
                   k! (-k + n)!                      (-1 + k)! (-k + n)!
```

Similarly,

```
In[10]:= Gosper[Binomial[n,k]^(-1),k]
Out[10]= Try higher order
```

and the extension delivers

```
In[11]:= Gosper[Binomial[n,k]^(-1), {k,1,r}, 1]
                    (-2 k + n) k! (1 - k + n)!
Out[11]= {SUM[-(--------------------------), {k, 1, r}] ==
                           (1 - k + n) n!

            (n - r)! (1 + r)!
>      -1 + -----------------}
                    n!
```

which is entry (2.6) in Gould (1972). For more involved summations the minimal degree, corresponding to the setting of "order", for the Gosper-polynomial increases. For instance, considering $\sum_k \binom{n}{k}^3$ the minimal degree ("order") to make it Gosper-summable is 3.

4.2. The Polynomial Multiplier

In this subsection we remark on a feature concerning the input structure which is a trivial one from technical point of view, but important for applications as we shall see in the example. It concerns admitting an arbitrary polynomial as a multiplicative constituent of the summand. In terms of the description of section 3.3.2 this corresponds to an appropriate specification of *ratFun*.

In the standard hypergeometric summation/transformation tables the specific entries almost always arise in fully factored form (over the complex numbers). Thus from hypergeometric point-of-view one might be tempted to neglect the importance of admitting an arbitrary polynomial multiplier. What can be gained from that will be clear by having a look at the following example.

Jackson (1988), using character theoretic methods and the group algebra of the symmetric group, derived certain properties of the number of permutations, with only p-cycles, for an arbitrary but fixed p, which are expressible as the product of a full cycle and a fixed point free involution. In Jackson's study the following sum turns out to be crucial:

$$T(m,p) := \sum_{i=0}^{mp-1} (-1)^i \binom{mp-1}{i}^{-1} \{[x^i]\frac{(1-x^2)^{mp/2}}{(1+x)}\}\{[x^i]\frac{(1-(-x)^p)^m}{(1+x)}\}.$$

Writing $[x^i]p(x)$ denotes the i-th coefficient in the polynomial $p(x)$.

In a study of the case $p = 4$, Andrews (1988) pointed out that the evaluation of $T(m,4) = 0$ for even m is an easy exercise having a computer algebra system at hand. In fact, it helps in the analysis of the summand sequence, especially in observing its symmetry properties. For *odd* m, the beautiful evaluation

$$T(m,4) = 8^m \binom{m}{(m-1)/2} \binom{4m}{2m}^{-1}$$

of the "sufficiently complicated sum" (Andrews) was found by Jackson (1988) via deriving some integral representation for $T(m,p)$, while Andrews (1988) derived it by using heavy hypergeometric machinery. At that time Zeilberger's algorithm hasn't been available yet. Now it is, and the problem can be treated as follows.

Extracting the x^i coefficients, by applying the binomial theorem, the expression for $T(m,4)$ (note that we fixed m to be odd) can be rewritten as

$$EO(m) := 4 \sum_{j=0}^{m-1} (-1)^j \binom{m-1}{j} \binom{2m-1}{2j} \binom{4m-1}{4j}^{-1} \frac{2m-4j-1}{4m-4j-1}.$$

Andrews translated this sum into a terminating hypergeometric ${}_4F_3$ series and applied several nontrivial transformations in order to arrive at the desired closed form expression.

Summarizing, we have for m odd that $T(m,4) = EO(m)$, and for even m that $T(m,4) = 0$. In addition, one observes that $EO(m)$ has a non-zero closed form evaluation also for *even* m, as we shall see below. This fact was mentioned neither by Jackson nor by Andrews.

Inputting the sum $EO(m)$ into Zeilberger's algorithm one obtains as the recurrence of minimal order which can be found:

```
In[12]:= Zb[4 (-1)^j Binomial[m-1,j] Binomial[2m-1,2j] *
         Binomial[4m-1,4j]^(-1) (2m-4j-1)/(4m-4j-1) , {j,0,m-1}, m, 2]
                        2
Out[12]= {64 (1 + m)  (2 + m) (1 + 2 m) (3 + 2 m) SUM[m] +

>       8 (2 + m) (3 + 2 m) (1 + 4 m) (3 + 4 m) SUM[1 + m] -

>       (3 + m) (1 + 4 m) (3 + 4 m) (5 + 4 m) (7 + 4 m) SUM[2 + m] == 0}
```

If we extend the definition of $EO(m)$ by $EO(0) := 1$, it is easily checked that the recurrence holds from $m = 0$ on.

Now the goal is to find a closed form solution, i.e. we consider $EO(m)$ as the unknown satisfying the recursion for $m \geq 0$ and with initial values $EO(0) = 1, EO(1) = 4/3$.

The extremely nice form of the recurrence allows to modify the problem. First, the occurrence of the constant factors 64, 8, 1 suggests to multiply both sides by 8^{-m}. By this standard technique we get a new recurrence, now in $S(m) := 8^{-m} EO(m)$ with $S(0) = 1$ and $S(1) = 1/6$, and with the same coefficient polynomials but without the powers of 8. So far we haven't gained that much, but more can be done. Dividing both sides of the new recurrence in $S(n)$ by $(1+m)(2+m)(1+2m)(3+2m)$ results in

$$(1+m)S(m) + rat(m)S(m+1) - (3+m)rat(m+1)rat(m)S(m+2) = 0,$$

where $rat(m) = (1+4m)(3+4m)/((1+m)(1+2m))$ is the rational certificate of the hypergeometric sequence $a_m = 2^{-m}\binom{4m}{2m}$, i.e. $a_{m+1}/a_m = rat(m)$. Hence multiplying both sides by a_m and setting $T(m) := a_m\, S(m)$ results in

$$(1+m)T(m) + T(m+1) - (3+m)T(m+2) = 0.$$

Note that now $EO(m) = 2^{4m}\binom{4m}{2m}^{-1} T(m)$ for $m \geq 0$ and with $T(0) := 1, T(1) := 1/2$.

We want to remark that if one is not able to reduce a recurrence by inspection, as we did above, in general one could try to extract the hypergeometric factor heuristically.

For instance, having a computer algebra system at hand, one could examine the integer factorizations of the ratios $EO(m+1)/EO(m)$, or $EO(m+2)/EO(m)$ etc., for some small values of m and then look out for a pattern. Here programs like the *Maple* procedure `gfun` written by Salvy and Zimmermann, see e.g. Salvy and Zimmermann (1993), might be helpful. Suppose our guess is, as above, $2^{4m}\binom{4m}{2m}^{-1}$ then using Zeilberger's algorithm it is easy to check whether the recursion boils down or not. For instance:

```
In[13]:= Zb[2^(-4m) Binomial[4m,2m] 4 (-1)^j Binomial[m-1,j] *
         Binomial[2m-1,2j] Binomial[4m-1,4j]^(-1) (2m-4j-1)/(4m-4j-1),
         {j,0,m-1}, m, 2]
Out[13]= {(1 + m) SUM[m] + SUM[1 + m] + (-3 - m) SUM[2 + m] == 0}
```

This recursion finds no hypergeometric solution $T(m)$ which can be seen, for instance, by applying Petkovšek's (1992) algorithm. Consequently, the term $2^{4m}\binom{4m}{2m}^{-1}$ we extracted constitutes the hypergeometric part of $EO(m)$. Nevertheless, the remainder $T(m)$ is "almost" hypergeometric which is made more precise as follows.

Denoting $TE(n) := T(2n)$ for $n \geq 0$, one gets

```
In[14]:= Zb[2^(-8n) Binomial[8n,4n] 4 (-1)^j Binomial[2n-1,j] *
          Binomial[4n-1,2j] Binomial[8n-1,4j]^(-1) (4n-4j-1)/(8n-4j-1),
          j, n, 2]
                                                      2
Out[14]= {-2 (1 + n) (1 + 2 n) SUM[n] + (19 + 24 n + 8 n ) SUM[1 + n] -

>       2 (2 + n) (5 + 2 n) SUM[2 + n] == 0}
```

as the recurrence for $TE(n)$.

Remark: Certainly the same recursion can be obtained without Zeilberger's algorithm by standard transformation of the recurrence to step-width 2.

NOW Petkovšek's algorithm finds as the unique hypergeometric solution, with respect to the initial values $TE(0) = T(0) = 1, TE(1) = T(2) = 1/2$, that $TE(n) = 2^{-2n}\binom{2n}{n}$. Analogously, for $TO(n) := S(2n+1)$ $(n \geq 0)$ the unique hypergeometric solution of the corresponding recurrence, with initial values $TO(0) = T(1) = 1/2, TO(1) = T(3) = 3/8$, is found to be $TO(n) = 2^{-2n-1}\binom{2n+1}{n}$. Both solutions combined into one gives for $m \geq 0$:

$$EO(m) = 8^m \binom{m}{\lfloor m/2 \rfloor}\binom{4m}{2m}^{-1},$$

which proves the above proposition for odd m. In view of Jackson's work one might ask whether the closed form evaluation in case of even m has any combinatorial significance.

We conclude this subsection by showing an evaluation "in one stroke". Instead of rewriting the sum expression $T(m,4)$ only for the case m odd, one can check that for $m \geq 0$:

$$T(m,4) = 4\sum_{k=0}^{m-1}\binom{m-1}{k}\binom{2m-1}{2k}\binom{4m-1}{4k}^{-1}\frac{4m^2+16k^2-16km+16k-6m+3}{(4m-4k-3)(4m-4k-1)}.$$

Applying Zeilberger's algorithm yields

```
In[15]:= Zb[(-1)^k Binomial[m-1,k] Binomial[2m-1,2k] *
         Binomial[4m-1,4k]^(-1) (4 m^2+16 k^2-16 k m+16k-6m+3) *
         1/( (4m-4k-3)(4m-4k-1) ), k, m, 2]

                                     2
Out[15]= {-64 (1 + m) (2 + m)  (1 + 2 m) (3 + 2 m) SUM[m] +

>       (3 + m) (1 + 4 m) (3 + 4 m) (5 + 4 m) (7 + 4 m) SUM[2 + m] == 0}
```

($SUM[m] = T(m,4)$). It's a surprise indeed that the form of this recursion, with respect to a sum being even more complicated than the $EO(m)$-sum, see the *irreducible* polynomial in the numerator, is more elegant, and simple enough to being solved by hand. Observing that $T(0,4) = 0$ and $T(1,4) = 4/3$, it is an easy exercise to check the corresponding evaluations given above for even and odd m.

4.3. The Zb-recurrence is not always minimal

Studying a huge number of practical applications one is tempted to conjecture that Zeilberger's algorithm always returns the recurrence with minimal order. This would imply the algorithmic solution of the canonical form problem for binomial sums, with summands specified as above, by taking the minimal recurrence together with the initial values as the canonical certificate. Unfortunately, that is not true.

Consider the sum

$$S_d(n) := \sum_{k=0}^{n} (-1)^k \binom{n}{k} \binom{d\,k}{n}$$

with positive integers d, n. Using standard technique the evaluation $S_d(n) = (-d)^n$ is almost trivial to prove, for instance, see Graham, Knuth and Patashnik (1989), eq. (5.33). Let us investigate the problem equipped with Zeilberger's algorithm . For $d = 1$ and $d = 2$ everything is fine:

```
In[16]:= Zb[(-1)^k Binomial[n,k] Binomial[k,n], k,n,1]
Out[16]= {SUM[n] + SUM[1 + n] == 0}

In[17]:= Zb[(-1)^k Binomial[n,k] Binomial[2k,n], k,n,1]
Out[17]= {2 SUM[n] + SUM[1 + n] == 0}
```

But checking $d = 3$ invokes a surprise:

```
In[18]:= Zb[(-1)^k Binomial[n,k] Binomial[3k,n], k,n,1]
Out[18]= Try higher order

In[19]:= Zb[(-1)^k Binomial[n,k] Binomial[3k,n], k,n,2]
Out[19]= {9 (1 + n) SUM[n] + 3 (7 + 5 n) SUM[1 + n] +

>       2 (3 + 2 n) SUM[2 + n] == 0}
```

We see that in this instance Zeilberger's algorithm does not deliver the minimal recurrence. The explanation is that the corresponding operator factors:

$$2(3+2n)N^2+3(7+5n)N+9(1+n)I=(2(2n+3)N+3(n+1)I)\ (N+3),$$

where N is the shift-operator with respect to n, i.e. $Nf(n)=f(n+1)$. The detection of this factorization lies outside the scope of the algorithm. More generally, for the sum $S_d(n)$, where $d\geq 3$, Zeilberger's algorithm cannot find a recurrence with respect to order $j\leq d-2$, and the first recurrence it will deliver is of order $d-1$. The proof is left to the reader. In other words, one can prove that there is no linear operator $Op(N)$ of order j with $1\leq j\leq d-2$ and with coefficients being polynomials in n, and no hypergeometric sequence $(g(n,k))_{k\geq 0}$ such that

$$Op(N)(-1)^k\binom{n}{k}\binom{d\ k}{n}=g(n,k+1)-g(n,k).$$

The minimal order j for an operator $Op(N)$ with this property is $j=d-1$. For that choice it is guaranteed that Gosper's algorithm will find the corresponding sequence $(g(n,k))_{k\geq 0}$. From the closed form evaluation $S_d(n)=(-d)^n$, i.e. $NS_d(n)+d\ S_d(n)=0$, we know that this operator $Op(N)$ of order $d-1$ factors as

$$Op(N)=(p_{d-2}(n)N^{d-2}+\cdots+p_0(n)I)(N+dI).$$

Without knowing the closed form evaluation a priori, applying Zeilberger's algorithm the right-factor $N+dI$ has to be determined by *factoring* the output-recursion. Nevertheless, we again want to emphasize that in applications non-minimality quite rarely occurs.

4.4. Summation bounds

In this subsection we are going to illustrate how useful it is to be able to handle sums where the summation bounds are *not* naturally induced by the summand. As briefly described in section 2 the general case leads to an inhomogeneous recurrence.

Consider Graham, Knuth and Patashnik (1989), exam problem (7.46): "Evaluate

$$\sum_{k=0}^{\lfloor n/2\rfloor}\binom{n-2k}{k}(-\frac{4}{27})^k$$

in closed form. *Hint:* $z^3-z^2+\frac{4}{27}=(z+\frac{1}{3})(z-\frac{2}{3})^2$".

In the solution it is pointed out that if $S_n(a):=\sum_{k=0}^{\lfloor n/2\rfloor}\binom{n-2k}{k}a^k$ then $S_n(a)=S_{n-1}(a)+a\ S_{n-3}(a)+(n=0)$, and its generating function is $1/(1-z-a\ z^3)$. When $a=-4/27$ the hint tells the nice factorization of the generating function which leads to the closed form evaluation $S_n(-4/27)=(2n/3+8/9)(2/3)^n+1/9(-1/3)^n$.

Using the feature of admitting general summation bounds, as specified in section 3.3.3, the generating function can be obtained by computer, for instance, as follows. Due to the floor function in the upper bound we distinguish between even and odd subsequences.

The smallest recurrences we find are of order 3:

```
In[20]:= Simplify[Zb[Binomial[2m-2k,k] a^k,{k,0,m}, m, 3]]
                2
Out[20]= {-(a  SUM[m]) - 2 a SUM[1 + m] - SUM[2 + m] + SUM[3 + m] ==
```

```
             3 + m
            a
>     -----------------}
      (-3 - m)! (3 + m)!

In[21]:= Simplify[Zb[Binomial[2m+1-2k,k] a^k,{k,0,m}, m, 3]]
             2
Out[21]= {-(a  SUM[m]) - 2 a SUM[1 + m] - SUM[2 + m] + SUM[3 + m] ==

             3 + m
            a
>     -----------------}
      (-2 - m)! (3 + m)!
```

Due to the properties of the Gamma function, for $m \geq 0$ both inhomogeneous parts evaluate to zero. Thus for both the even and the odd subsequences we get the same *homogeneous* recurrence which can be combined into one as follows. For $n \geq 0$:

$$(N^6 - N^4 - 2aN^2 - a^2 I)S_n(a) = 0,$$

where $NS_n(a) = S_{n+1}(a)$ is the shift operator in n. Using the `Interlace` function of a *Mathematica* package written by Nemes and Petkovšek (1994) one does not get this operator but the minimal annihilating one which is a factor of the operator above:

$$(N^3 - N^2 - aI)S_n(a) = 0.$$

Checking the initial values, the generating function can be read off directly.

But even more can be done knowing that $a = -4/27$.

```
In[22]:= Simplify[Zb[Binomial[n-2k,k] (-4/27)^k,{k,0,m}, n,2]]
Out[22]= {2 (3 + n) SUM[n] + 3 (4 + n) SUM[1 + n] - 9 (2 + n) SUM[2 + n] ==

             m  4  m
      -4 (-1)  (--)  (1 + m) (-2 m + n)!
                27
>     ---------------------------------}
           (1 + m)! (-1 - 3 m + n)!
```

Note that for setting $m = \lfloor n/2 \rfloor$ the right hand side is equal to $(-4)\ \delta_{n,1}$ where $\delta_{n,1}$ is the Kronecker symbol. Note also that

$$\mathtt{SUM}[n] = \sum_{k=0}^{m} \binom{n-2k}{k} (-\frac{4}{27})^k.$$

Therefore, in extracting a recurrence for $S_n(-4/27)$ one has to be a little bit careful. More precisely, one has to consider the dependency on m in the upper bound, a task which, in principle, also can be done by computer. Indicating the dependency on m more explicitly by writing $\mathtt{SUM}[n] = \mathtt{SUM}[n, m]$ one can check that

$$\mathtt{SUM}[n, \lfloor \frac{n}{2} \rfloor] = S_n(a_0), \quad \mathtt{SUM}[n+1, \lfloor \frac{n}{2} \rfloor] = S_{n+1}(a_0),$$

and

$$\texttt{SUM}[n+2, \lfloor \frac{n}{2} \rfloor] = S_{n+2}(a_0) - a_0\ \delta_{n,1},$$

where $a_0 = -4/27$. Now the above recursion turns into

$$2(n+3)S_n(a_0) + 3(n+4)S_{n+1}(a_0) - 9(n+2)(S_{n+2}(a_0) - a_0\ \delta_{n,1}) = (-4)\delta_{n,1}$$

which simplifies to

$$2(n+3)S_n(a_0) + 3(n+4)S_{n+1}(a_0) - 9(n+2)S_{n+2}(a_0) = 0$$

for $n \geq 0$.

By Petkovšek's algorithm(or by hand computation) we see that this homogeneous recursion possesses two independent hypergeometric solutions which combine to the closed form evaluation of $S_n(-4/27)$ as above.

The explanation for obtaining a recurrence of order 2 lies in the fact that the operator obtained for $S_n(a)$ factors exactly for $a = a_0 = 4/27$ into the form $\alpha(z+\beta)(z+\gamma)^2$. Certainly, one can select the parameter a such that $S_n(a)$ finds a solution as a linear combination of *three* different powers of n. In this case the order of the minimal recurrence is indeed 3.

We want to remark that not all examples with non-naturally induced summation bounds necessarily end up in an inhomogeneous recurrence. The inhomogeneous part might evaluate to zero. For example, see Wilf (1993):

```
In[23]:= Zb[Binomial[n+k,k] 2^(-k),{k,0,n},n,1]
Out[23]= {2 SUM[n] - SUM[1 + n] == 0}
```

From

```
In[24]:= Simplify[Zb[Binomial[n+k,k] a^k,{k,0,n},n,1]]
Out[24]= {SUM[n] + (-1 + a) SUM[1 + n] ==

            1 + n
           a      (-1 + 2 a - n + 2 a n) (1 + 2 n)!
>          ----------------------------------------}
                                     2
                              (1 + n)!
```

it is immediately clear, that $a = 1/2$ is the only possible choice.

A more involved example concerns ${}_2F_1$-summation, in binomial notation

$${}_2F_1\left(\begin{matrix} a, b \\ c \end{matrix}; z\right) = \sum_{k=0}^{\infty} \binom{a+k-1}{k}\binom{b+k-1}{k}\binom{c+k-1}{k}^{-1} z^k,$$

assuming appropriate conditions for convergency (see e.g. Slater (1966)). Let's take the n-th partial sum and ask for a recurrence in c:

```
In[25]:= Zb[Binomial[a+k-1,k] Binomial[b+k-1,k]*
         Binomial[c+k-1,k]^(-1) z^k, {k,0,n}, c, 2]
Out[25]= {c (1 + c) (-1 + z) SUM[c] -
```

```
>       (1 + c) (-c + z - a z - b z + 2 c z) SUM[1 + c] +

>       (1 - a + c) (1 - b + c) z SUM[2 + c] ==

         1 + n
        z       (1 + c)! (a + n)! (b + n)!
>      ----------------------------------}
       (-1 + a)! (-1 + b)! n! (1 + c + n)!
```

Using well-known properties of the Gamma function, in the limit $n \to \infty$ the inhomogeneous part becomes 0. As a consequence we derived an instance of what is called "Gaussian contiguous relations", see e.g. Slater (1966). Substituting $z = 1$ and iterating the resulting recurrence is one of the standard proofs of Gauss' summation theorem,

$$
{}_2F_1\left(\begin{array}{c} a, b \\ c \end{array}; 1\right) = \frac{\Gamma(c-a-b)\Gamma(c)}{\Gamma(c-a)\Gamma(c-b)},
$$

again under appropriate conditions for convergency.

Another contiguous relation can be derived, for instance, with respect to a:

```
In[26]:= Zb[Binomial[a+k-1,k] Binomial[b+k-1,k]*
         Binomial[c+k-1,k]^(-1) z^k, {k,0,n}, a, 2]
Out[26]= {(-1 - a + c) SUM[a] + (2 + 2 a - c - z - a z + b z) SUM[1 + a] +

                                            1 + n
                                           z       (-1 + c)! (a + n)! (b + n)!
>       (1 + a) (-1 + z) SUM[2 + a] == ---------------------------------},
                                            a! (-1 + b)! n! (-1 + c + n)!
```

and taking the limit $n \to \infty$. But for the resulting homogeneous relation it is not at all obvious, as for Gauss' summation, that specializing $c = 1 - a + b$ and $z = -1$ leads to a recurrence of order 1. This choice of parameters corresponds to the classical Kummer ${}_2F_1$-summation:

$$
{}_2F_1\left(\begin{array}{c} a, b \\ 1-a+b \end{array}; -1\right) = \frac{\Gamma(b/2+1)\Gamma(b-a+1)}{\Gamma(b+1)\Gamma(b/2-a+1)},
$$

see e.g. Slater (1966). But once we *know how to choose the parameters*, the desired recurrence is delivered immediately by:

```
In[27]:= Zb[Binomial[a+k-1,k] Binomial[b+k-1,k]*
         Binomial[1-a+b+k-1,k]^(-1) (-1)^k, {k,0,n}, a, 1]
Out[27]= {(-2 a + b) SUM[a] - 2 (-a + b) SUM[1 + a] ==

           1 + n
       (-1)       (-a + b)! (a + n)! (b + n)!
>      ------------------------------------}
           a! (-1 + b)! n! (-a + b + n)!
```

Here the explanation is that exactly under Kummer's substitution the operator resulting from the homogeneous version of the contiguous relation in a above,

$$-2(1+a)A^2 + 2(1+2a-b)A + (-2a+b)I,$$

factors as

$$\left(\frac{1+a}{-a-1+b}A + I\right)(-2(-a+b)A + (-2a+b)I),$$

where A is the shift-operator with respect to a. Consequently, an appropriate factorization algorithm for shift operators would serve as a powerful tool for a systematic algorithmic study of hypergeometric summation theorems.

We conclude this section by presenting an example where general bounds arise in a *transformation* problem. (This problem was kindly communicated to us by I. Nemes.) Problem 1394 of Math. Magazine (April 1992, proposer David Callan) asked for a proof of the identity

$$\sum_{k=0}^{n} \frac{1}{2k+1}\binom{2n+1}{n-k} = \sum_{k=0}^{n} \frac{4^k}{2k+1}\binom{2n-2k}{n-k}.$$

Note that the summation bounds are not naturally induced (this time for negative k). Consequently one has to deal with inhomogeneous recurrences:

```
In[28]:= Zb[1/(2k+1) Binomial[2n+1,n-k], {k,0,n}, n,1]
                                                      (2 + 2 n)!
Out[28]= {8 (1 + n) SUM[n] + (-3 - 2 n) SUM[1 + n] == -(----------)}
                                                            2
                                                      (1 + n)!

In[29]:= Zb[4^k/(2k+1) Binomial[2n-2k,n-k], {k,0,n}, n,1]
                                                      (2 + 2 n)!
Out[29]= {8 (1 + n) SUM[n] + (-3 - 2 n) SUM[1 + n] == -(----------)}
                                                            2
                                                      (1 + n)!
```

The proof is completed by checking the initial values of both sums for $n = 0$.

4.5. Transformations and double sums

Another application aspect of fundamental importance is based on the fact that Zeilberger's algorithm can be used not only for proving (and finding) summation identities, but also for handling *transformations*, as in the last example of the previous section. Roughly spoken, by transformations we mean identities of the form $\sum \ldots = \sum \ldots$, whereas *summations* point to the type $\sum \ldots = closedform$. A more prominent hypergeometric example is Pfaff's *reflection law*, see e.g. Graham, Knuth and Patashnik (1989), which we prove in its terminating form:

```
In[30]:= Zb[(1-z)^n Binomial[-n+k-1,k] Binomial[b+k-1,k]*
         Binomial[c+k-1,k]^(-1) (z/(z-1))^k, k, n, 2]
Out[30]= {(1 + n) (-1 + z) SUM[n] +
```

```
>      (2 + c + 2 n - z + b z - c z - n z) SUM[1 + n] +

>      (-1 - c - n) SUM[2 + n] == 0}
```

and

```
In[31]:= Zb[Binomial[-n+k-1,k] Binomial[c-b+k-1,k]*
         Binomial[c+k-1,k]^(-1) z^k, k, n, 2]
Out[31]= {(1 + n) (-1 + z) SUM[n] +

>      (2 + c + 2 n - z + b z - c z - n z) SUM[1 + n] +

>      (-1 - c - n) SUM[2 + n] == 0}
```

Equality of the corresponding sums follows by checking the initial values. Some more transformation examples will be given below, but in this paper we do not focus on this important topic.

Wilf and Zeilberger (1992) described a general approach for an algorithmic treatment of multiple binomial sums. It is the goal of this section to illustrate that for many problems arising in practice just Zeilberger's algorithm for *single* sums might be able to do the job.

For nonnegative integers m, n consider

$$\sum_{j,k}(-1)^{j+k}\binom{j+k}{j}\binom{r}{j}\binom{n}{k}\binom{m+n-j-k}{m-j}=\binom{n+r}{n}\binom{m-r}{m-n}.$$

(Note that both sides can be viewed as polynomials in r, thus w.l.o.g. we restrict r to be a nonnegative integer.) To that identity Graham, Knuth and Patashnik (1989), bonus problem 5.83, remark that it "even has a chance of arising in practical applications" and ask for a "substantially shorter proof" than that one given in the solution section.

Equipped with Zeilberger's algorithm one could try whether one of the inner sums finds a closed form evaluation, i.e. a recurrence of order 1, and then apply Zeilberger's algorithm again to complete. Despite the fact that for a good deal of problems this might work quite well, this approach fails here. What is needed is a small portion of human preprocessing, then the computer is able to complete the work.

The "human step" is the idea to use the well-known elementary inversion formula

$$g(r)=\sum_{j}(-1)^j\binom{r}{j}f(j)\Leftrightarrow f(r)=\sum_{j}(-1)^j\binom{r}{j}g(j),$$

see e.g. Graham, Knuth and Patashnik (1989), eq. (5.58). In view of that it is immediate that proving the double-sum is equivalent to proving

$$\sum_{k}(-1)^k\binom{n}{k}\binom{r+k}{r}\binom{m+n-r-k}{m-r}=\sum_{j}(-1)^j\binom{r}{j}\binom{n+j}{n}\binom{m-j}{m-n}.$$

Apart from checking the initial values, in this form Zeilberger's algorithm generates as a "one line" proof:

```
In[32]:= Zb[(-1)^k*Binomial[n,k]*Binomial[k+r,k]*
```

```
         Binomial[m+n-r-k,m-r], k, n, 2]
Out[32]= {(-m + n) (2 + m + n) SUM[n] - (3 + 2 n) (-m + 2 r) SUM[1 + n] -

                2
>       (2 + n)  SUM[2 + n] == 0}

In[33]:= Zb[(-1)^j*Binomial[r,j]*Binomial[n+j,n]*
         Binomial[m-j,m-n], j, n, 2]
Out[33]= {-((-m + n) (2 + m + n) SUM[n]) + (3 + 2 n) (-m + 2 r) SUM[1 + n] +

                2
>       (2 + n)  SUM[2 + n] == 0}
```

This *inversion-technique* can be applied in various situations. A beautiful example is provided by a transformation which was studied extensively by Strehl (1993), where also reference to its significant number-theoretic relevance can be found:

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

As Strehl showed, in this instance another inversion pair, namely Legendre-inversion, see e.g. Riordan (1968), can be successfully applied.

In order to demonstrate the usefulness of the inversion-method, we discuss another example which also stems from Strehl (1993), but was proven there by different means:

$$\sum_{k=0}^{n} \binom{n}{k}^2 \binom{2k}{k} = \sum_{k=0}^{n} \binom{n}{k} \sum_{j=0}^{k} \binom{k}{j}^3.$$

Because again none of the single sums finds a recurrence of order 1, we modify the problem by applying the inversion technique. Using the same inversion pair as above the problem is equivalent to prove

$$(-1)^n \sum_{k=0}^{n} \binom{n}{k}^3 = \sum_{k=0}^{n} (-1)^k \binom{n}{k} \sum_{j=0}^{k} \binom{k}{j}^2 \binom{2j}{j}.$$

Now the right hand side can be rewritten as $\sum_{j=0}^{n} \binom{2j}{j}\ S(n)$ where

$$S(n) := \sum_{k=j}^{n} (-1)^k \binom{n}{k} \binom{k}{j}^2.$$

This time we are lucky:

```
In[34]:= Zb[(-1)^k Binomial[n,k] Binomial[k,j]^2, k, n, 1]
                                                    2
Out[34]= {(1 + n) (-2 j + n) SUM[n] - (1 - j + n)  SUM[1 + n] == 0}
```

which means that $S(n) = (-1)^n \binom{n}{j}\binom{j}{n-j}$. Consequently the original problem is equivalent to prove

$$\sum_{k=0}^{n} \binom{n}{k}^3 = \sum_{k=0}^{n} \binom{n}{k}^2 \binom{2k}{n}$$

which again can be done by Zeilberger's algorithm:

```
In[35]:= Zb[Binomial[n,k]^3, k, n, 2]
                          2                        2
Out[35]= {-8 (1 + n)  SUM[n] + (-16 - 21 n - 7 n ) SUM[1 + n] +

                2
>       (2 + n)  SUM[2 + n] == 0}

In[36]:= Zb[Binomial[n,k]^2 Binomial[2k,n], k, n, 2]
                          2                        2
Out[36]= {-8 (1 + n)  SUM[n] + (-16 - 21 n - 7 n ) SUM[1 + n] +

                2
>       (2 + n)  SUM[2 + n] == 0}
```

Of course, not for all applications an appropriate inversion-pair is at hand. Nevertheless, there are other possibilities to transform the problem into a "Zeilberger-tractable" one. A more detailed discussion is in preparation. See also Paule (1992*b*).

4.6. Asymptotics

In case a sum is not evaluable in closed form, for instance, as a hypergeometric term, the next question in applications might be to ask for an appropriate asymptotic estimate. In solving this problem, the recurrence certificate, despite being of order greater than 1, might be of great help.

Consider the sum

$$S(n) := \sum_{k=0}^{n} \binom{3n}{k}.$$

The recurrence tells us that no hypergeometric evaluation is to expect:

```
In[37]:= Simplify[Zb[Binomial[3m,k], {k,0,m}, m, 1]]
                                                      2
                                     (4 + 11 m + 5 m ) (3 m)!
Out[37]= {8 SUM[m] - SUM[1 + m] == ------------------------}
                                       (1 + m)! (1 + 2 m)!
```

Multiplying both sides with 8^{-m-1}, and summing both sides for m from 0 to $n-1$, we get after telescoping

$$8^{-n} S(n) = 1 - \frac{2}{3} \sum_{m=1}^{n} 8^{-m} \frac{5m^2 + m - 2}{(3m-1)(3m-2)} \binom{3m}{m}.$$

As explained by Wilf (1993), it is possible to treat the right hand side by elegant but elementary means in order to extract asymptotic information, and to arrive at

$$S(n) \sim 2\binom{3n}{n} \quad \text{for} \quad n \to \infty.$$

We conclude this section with an example arising from Padé approximation of irrational numbers, see D. Chudnovsky and R. Chudnovsky (1989).

```
In[38]:= Zb[Binomial[3n,k] Binomial[3n-1/2,4n-k] (3/4)^k, k, n, 2]
Out[38]= {27 (1 + n) (1 + 2 n) (1 + 3 n) (2 + 3 n) (1 + 6 n) (5 + 6 n)

                                      2          3
>       (117000 + 217676 n + 134001 n  + 27279 n ) SUM[n] +

>      384 (1 + 4 n) (3 + 4 n) (16022087856 + 128021157420 n +

                         2                  3                  4
>        415975459648 n  + 715282318379 n  + 706125904254 n  +

                         5                  6                 7
>        401859218160 n  + 122518066482 n  + 15484624281 n ) SUM[1 + n] -

>      262144 (2 + n) (3 + 2 n) (1 + 4 n) (3 + 4 n) (5 + 4 n) (7 + 4 n)

                                  2          3
>       (6046 + 31511 n + 52164 n  + 27279 n ) SUM[2 + n] == 0}
```

In D. Chudnovsky and R. Chudnovsky (1989) this recursion, which is used for deriving asymptotic information, has been found by a suitable integral representation of the sum in question. It is interesting to note that the innocent looking summand gives rise to relatively large integer coefficients in the recursion. This is also reflected in the computing time which is around 100 seconds on an Apollo 4500 workstation. All other examples in this paper are in the range of a few seconds.

5. Symbolic Finite Summation in the MATHEMATICA System

In this section we briefly comment on the built-in *Mathematica* facilities for handling *finite* hypergeometric summation.

The first important issue is that without loading `SymbolicSum.m`, a *Mathematica* package, symbolic summation is not possible. For the following discussion assume that the package (February 1991), delivered with *Mathematica version 2.1* or *2.2*, has been loaded.

Within the corresponding file the scope of the package is described as follows:

```
(*:Limitations:
    This package can evaluate symbolic sums of the following type
              Sum[ a[k],{k,min,max} ] ,
    where
    a[k+1]/a[k] is a rational function.
```

```
*)
```

But the impression that something like a Gosper algorithm has been implemented turns out to be entirely false, once trying out one of the nontrivial but elementary examples:

```
In[39]:= Sum[k k!, {k,0,n}]
Out[39]= Sum[k k!, {k, 0, n}]
```

As we shall see later, the package `SymbolicSum.m` works differently in *version 2.2*, but for this example the behaviour is just the same with the exception of echoing an additional message:

```
In[2]:= Sum[k k!, {k,0,n}]

HypergeometricPFQ::hdiv:
    Warning: Divergent generalized hypergeometric series
    HypergeometricPFQ[{1, 2 + n, 2 + n}, {1 + n}, 1].

Out[2]= Sum[k k!, {k, 0, n}]
```

The strategy will become more clear trying out rational summation. One gets in *version 2.1*:

```
In[40]:= Sum[1/(k^2-1), {k,2,n}]
Out[40]= (2 n HypergeometricPFQ[{1, 1, 3}, {2, 4}, 1] +

          2
>        n  HypergeometricPFQ[{1, 1, 3}, {2, 4}, 1] -

>        3 HypergeometricPFQ[{1, n, 2 + n}, {1 + n, 3 + n}, 1]) /

>        (3 n (2 + n))
```

From this, one conjectures that the package translates finite sums over hypergeometric sequences into infinite hypergeometric series in classic ${}_pF_q$-notation. Nevertheless, in following this strategy one would need a tool for finding closed form evaluations in case they exist. For instance, one would like to have *algorithms* like Gosper's or Zeilberger's, or at least a facility for assisting a *table-lookup*, see e.g. the paper by Krattenthaler (this issue). At the first glance, this problem seems to be solved using `SymbolicSum.m` in *version 2.2*:

```
In[3]:= Sum[1/(k^2-1), {k,2,n}]
         (-1 + n) (2 + 3 n)
Out[3]= ------------------
            4 n (1 + n)
```

But this impression turns out to be wrong as we shall see later. Alternatively, our Gosper package solves the problem as:

```
In[41]:= Gosper[1/(k^2-1), {k,2,n}]
                        1                         3   -1 + 2 (1 + n)
Out[41]= {SUM[----------------, {k, 2, n}] == - - --------------}
              (-1 + k) (1 + k)                    4    2 n (1 + n)
```

Let us consider a slightly more involved example:

```
In[4]:= ex = Factor[(1/(k^2+1) /.k->k+1)- 1/(k^2+1)]
                   1 + 2 k
Out[4]= -(-----------------------)
                 2              2
         (1 + k ) (2 + 2 k + k )
In[5]:= Sum[ex, {k,0,n}]

$RecursionLimit::reclim: Recursion depth of 256 exceeded.
...
$IterationLimit::itlim: Iteration limit of 4096 exceeded.
...
Interrupt> a
Out[5]= $Aborted
```

Thus in *version 2.2* we are faced with an infinite recursion. (The same happens in *version 2.1*.) But on this example also our package will fail, due to the irreducible non-linear factors of the denominator polynomial, cf. 3.3.2. Here, one might apply, for instance, a *rational* summation algorithm, see e.g. Paule (1992*a*), or a Gosper implementation admitting input of more general type. For instance, applying a corrected version of the *Mathematica* package `GosperSum.m`, delivered with *version 1.2*, which uses resultant computation yields:

```
In[42]:= PrivateGosperSum[ex, {k,0,n}]
                                             2      3
          1    (1 + 2 n) (3 + 2 n + 3 n  + 2 n )
Out[42]= -(-) - ---------------------------------
          5                 2              2
                     5 (1 + n ) (2 + 2 n + n )
```

Now we change from rational summation to binomial summation. Here the *Mathematica* strategy works even more unsatisfactory.

In *version 2.1* and *2.2*, for instance,

```
In[43]:= Sum[(-1)^k Binomial[n,k]^(-1), {k,0,n}]
                        k
                    (-1)
Out[43]= Sum[--------------, {k, 0, n}]
             Binomial[n, k]
```

I.e., the package fails also in the translation to classic hypergeometric notation. A fact which is disturbing especially in an example like this, because it turns out to be Gosper-summable:

```
In[44]:= Simplify[Gosper[(-1)^k Binomial[n,k]^(-1), {k,0,n}]]
                 k                                       n
             (-1)  k! (-k + n)!                (1 + (-1) ) (1 + n)!
Out[44]= {SUM[------------------, {k, 0, n}] == --------------------}
                     n!                              (2 + n) n!
```

Finally we want to point out that the same is true with respect to handling hypergeometric *transformations*. For example, without applying hypergeometric transformations to the following output representations, it is not possible to extract further information about the relation of the corresponding sums:

```
In[6]:= Sum[Binomial[n,k] Binomial[x,k] 2^k, {k,0,n}]
Out[6]= Hypergeometric2F1[-n, -x, 1, 2]

In[7]:= Sum[Binomial[n,k] Binomial[x+k,n], {k,0,n}]
        Gamma[1 + x] Hypergeometric2F1[-n, 1 + x, 1 - n + x, -1]
Out[7]= --------------------------------------------------------
                     Gamma[1 + n] Gamma[1 - n + x]
```

In *version 2.1* the output is essentially the same but with slight differences in the syntax.

But using Zeilberger's algorithm one gets

```
In[45]:= Zb[Binomial[n,k] Binomial[x,k] 2^k, {k,0,n},n,2]
Out[45]= {(1 + n) SUM[n] + (1 + 2 x) SUM[1 + n] + (-2 - n) SUM[2 + n] == 0}

In[46]:= Zb[Binomial[n,k] Binomial[x+k,n] , {k,0,n},n,2]
Out[46]= {(1 + n) SUM[n] + (1 + 2 x) SUM[1 + n] + (-2 - n) SUM[2 + n] == 0}
```

and equality follows by checking the initial values which could be done also by computer:

```
In[47]:= L[n_]:=Sum[Binomial[n,k] Binomial[x,k] 2^k, {k,0,n}]

In[48]:= R[n_]:=Sum[Binomial[n,k] Binomial[x+k,n] , {k,0,n}]

In[49]:= L[0]
Out[49]= 1

In[50]:= R[0]
Out[50]= 1

In[51]:= L[1]
Out[51]= 1 + 2 x

In[52]:= R[1]
Out[52]= 1 + 2 x
```

6. Conclusion

The huge potential of applicability of Zeilberger's algorithm can be hardly overestimated. For extensions and variants of the method, for instance the "WZ-method" or "WZ-pairs", see e.g. Wilf and Zeilberger (1992) or Wilf (1993). These papers and their references sections also provide the source for many further nontrivial examples.

In Koornwinder (1993), besides discussing Zeilberger's algorithm, a *Maple* implementation of a q-analogue is described. Based on recent work, Paule (1992a), the first named author outlined a q-Zeilberger algorithm which currently is going to be implemented by A. Riese in the *Mathematica* system.

We want to conclude by remarking that, besides those already mentioned in section 3, several further updates of our packages are planned. One issue is that it should be brought into *Mathematica Notebook* form, another concerns the speed-up of the linear equation solver. Other improvements envisioned concern, for instance, simplification or the evaluation of hypergeometric terms. We also want to emphasize that any kind of comments or criticisms of users of our packages are highly appreciated.

The work of the first named author was partially supported by the Austrian FWF grant P7220.

References

Andrews, G.E. (1988). Application of Scratchpad to Problems in Special Functions and Combinatorics. In *Trends in Computer Algebra, R. Janssen (Ed.), Lecture Notes in Computer Science* **296**, 158–166. Springer-Verlag Berlin Heidelberg.

Chudnovsky D.V., Chudnovsky G.V. (1989). Use of computer algebra for diophantine and differential equations. In *Computer Algebra, D.V. Chudnovsky and R.D. Jenks (Eds.), Lecture Notes in Pure and Applied Mathematics* **113**, 1–81. Marcel Dekker New York and Basel.

Fasenmyer, M.C. (1949). A note on pure recurrence relations. *Amer. Math. Monthly* **56**, 14–17.

Gosper, R.W. (1978). Decision procedures for indefinite hypergeometric summation. *Proc. Natl. Acad. Sci. U.S.A.* **75**, 40–42.

Gould, H.W. (1972). Combinatorial Identities - A Standardized Set of Tables Listing 500 Binomial Coefficient Summations. Revised Edition. Morgantown, West Virginia University.

Graham, R.L., Knuth, D.E., Patashnik, O. (1989). *Concrete Mathematics, A Foundation for Computer Science*. Reading: Addison-Wesley.

Jackson, D.M. (1988). Counting semiregular permutations which are products of a full cycle an an involution. *Trans. Amer. Math. Soc.* **305**, 317–331.

Knuth, D.E. (1969) *The Art of Computer Programming, volume 2 / Seminumerical Algorithms* , 2nd edition, Addison-Wesley Reading.

Koornwinder, T.H. (1993). On Zeilberger's Algorithm and its q-analogue. *J. of Comput. and Appl. Math.* **48**, 91–111.

Krattenthaler, C. (this issue). HYP and HYPQ - Mathematica packages for the manipulation of binomial sums and hypergeometric series, respectively q-binomial sums and basic hypergeometric series.

Lafon, J.C. (1983). Summation in finite terms. In *Computer Algebra and Symbolic Computation, 2nd ed.* 71–77. Springer-Verlag Wien.

Paule, P. (1992*a*). Greatest-Factorial Factorization and Symbolic Summation I. *RISC-Linz Report Series No. 93-02* (submitted to J. Symb. Comp.).

Paule, P. (1992*b*) Solution of a Seminaire Homework Example (28th SLC). *RISC-Linz Report Series No. 92-59*

Paule, P., Schorn M. (1993). A Mathematica version of Zeilberger's algorithm for proving binomial coefficient identities - A description how to use it. *RISC-LINZ Report Series No.93-36.*

Petkovšek, M. (1992). Hypergeometric solutions of linear recurrences with polynomial coefficients. *J. Symb. Comp.* **14**, 243–264.

Riordan, J. (1968) *Combinatorial Identities*. John Wiley.

Salvy, B., and Zimmermann, P. (1993). Gfun: A Maple package for the manipulation of generating and holonomic functions in one variable. *to appear in: ACM Transactions on Mathematical Software.*

Slater, L.J. (1966). *Generalized Hypergeometric Functions*. Cambridge University Press.

Strehl, V. (1993). Binomial identities - combinatorial and algorithmic aspects. *Lehrstuhl I fuer Informatik, Interner Bericht Nr. 93*. Institut fuer Mathematische Maschinen und Datenverarbeitung der Friedrich-Alexander-Universitaet Erlangen-Nuernberg.

Verbaeten, P. (1974). The automatic construction of pure recurrence relations. *Proc. EUROSAM'74, ACM-SIGSAM Bulletin* **8**, 96–98.

Wilf, H.S., Zeilberger, D. (1992). An algorithmic proof theory for hypergeometric (ordinary and "q") multisum/integral identities. *Invent. Math.* **108**, 575–633.

Wilf H.S. (1993). *Identities and their Computer Proofs*. "SPICE" lecture notes, available via anonymous `ftp` of the file `pub/wilf/lecnotes.ps` at the site `ftp.cis.upenn.edu`.

Zeilberger, D. (1990*a*). A holonomic systems approach to special function identities. *J. Comp. Appl. Math.* **32**, 321–368.

Zeilberger, D. (1990*b*). A fast algorithm for proving terminating hypergeometric identities. *Discr. Math.* **80**, 207–211.

Zeilberger, D. (1991). A Maple program for proving hypergeometric identities. *SIGSAM Bulletin, ACM Press* **25**, 4–13.

Zeilberger, D. (1993). Three recitations on holonomic systems and hypergeometric series. *Proc. of the 24th Séminaire Lotharingien, D. Foata (ed.), Publ. I.R.M.A. Strasbourg*, 5–37. (Reprinted in this special issue of JSC).