Fields Institute Communications
Volume **00**, 0000

A Mathematica q-Analogue of Zeilberger's Algorithm Based on an Algebraically Motivated Approach to q-Hypergeometric Telescoping

Peter Paule
Research Institute for Symbolic Computation
J. Kepler University Linz
A–4040 Linz, Austria
`Peter.Paule@risc.uni-linz.ac.at`

Axel Riese
Research Institute for Symbolic Computation
J. Kepler University Linz
A–4040 Linz, Austria
`Axel.Riese@risc.uni-linz.ac.at`

Abstract. *Mathematica* implementations, available by email request, of q-analogues of Gosper's and Zeilberger's algorithm are described. Nontrivial examples are given in order to illustrate the usage of these packages. The algorithms are based on a new approach to q-hypergeometric telescoping in which a new algebraic concept, q-greatest factorial factorization (qGFF), plays a fundamental role.

1 Introduction

Based on Gosper's [1978] algorithm for indefinite hypergeometric summation, Zeilberger's algorithm for proving definite hypergeometric summation and transformation formulae constitutes a recent breakthrough in symbolic computation. An excellent and detailed account of this theory can be found in the book of Petkovšek, Wilf and Zeilberger [1996].

Zeilberger also was the first who observed that these algorithms can be carried over to the q-hypergeometric case. Despite the fact that there already exist implementations in Maple, written by Zeilberger (cf. Petkovšek, Wilf and Zeilberger [1996]) and Koornwinder [1993], we felt the need to come up with another implementation `qZeil`, in Mathematica, which is able to deal with various important features not covered by the ones mentioned. A similar project was carried out for the ordinary hypergeometric case; see the Mathematica implementation by Paule and Schorn [1993].

1991 *Mathematics Subject Classification.* Primary 33D15, 33D20, 68Q40; Secondary 05A30, 68R05.

The work of the first author was supported in part by grant P7220 of the Austrian FWF.

Another aspect concerns the theoretical foundation of a q-analogue of Gosper's algorithm. Besides Karr's [1991] approach which covers indefinite q-hypergeometric summation in the general frame of his theory of difference field extensions, up to now it had been kind of a surprise that Gosper's algorithm can be carried over to the q-case almost word by word; cf. Koornwinder [1993]. It turns out that the new algebraic concept of "greatest factorial factorization", introduced by Paule [1995], provides an algebraic explanation not only of Gosper's algorithm, but also of its analogue for q-hypergeometric telescoping.

Summarizing, the general goals of the paper are: (i) to introduce our Mathematica implementation `qZeil` of a q-analogue of Zeilberger's (and Gosper's) algorithm to potential users; (ii) to explain the usage of `qZeil` by illustrating examples that range from simple to nontrivial; (iii) to present an algebraically motivated approach to q-hypergeometric telescoping the q-Zeilberger algorithm is based on. Since the sections to some extent can be read independently from each other, we shortly comment on the table of contents.

In Section 2, "q-greatest factorial factorization" (qGFF) of polynomials is introduced. It is a q-analogue of the concept introduced by Paule [1995] and provides the crucial tool for analyzing q-hypergeometric telescoping.

In Section 3, qGFF is used to explain the mechanism of the algorithm for solving the q-hypergeometric telescoping problem, i.e., finding a q-hypergeometric sequence $\langle g_k \rangle$ such that $f_k = g_{k+1} - g_k$ for a given q-hypergeometric sequence $\langle f_k \rangle$. In our Mathematica implementation this algorithm is called "qTelescope".

In Section 4, a detailed description of the usage of the Mathematica package `qZeil` is given. All notions needed for the understanding of the various program features are described in short, for instance: qWZ-certification, qWZ-pairs, qWZ-dualization and qWZ-companion identities. The section concludes with a brief comparison with Koornwinder's implementation.

Section 5 deals with nontrivial applications. As in Paule and Schorn [1993], a special emphasis is put on illustrating how to apply the package to concrete problems. Besides the examples one finds in Section 4, in this section we present: an example where a computer gave the first proof of a (human) conjecture, a detailed discussion of "creative symmetrizing" for reducing the order of the q-Zeilberger output recurrence, the observation that L.J. Rogers' classical finite version of Euler's pentagonal number theorem is nothing but the dual identity of a limiting case of q-Chu-Vandermonde, and a computer proof of an infinite q-series identity recently discovered by R.J. McIntosh.

2 q-Greatest Factorial Factorization

In this section, "q-greatest factorial factorization" (qGFF) of polynomials is introduced. It is a q-analogue of a new canonical form representation (GFF) introduced by Paule [1995], defined with respect to the q-shift operator ϵ instead of the shift $(Ep)(x) = p(x+1)$ as for GFF.

2.1 Basic Definitions. By $\mathbf{N}$ we understand the set of all nonnegative integers. We assume $\mathbf{K}$ to be a *field* of *characteristic zero*, for instance, the complex numbers. Its transcendental extension by the indeterminate q is denoted by $\mathbf{F}$, i.e., $\mathbf{F} = \mathbf{K}(q)$. For implementations and for algorithm specification, $\mathbf{K}$ is assumed to be effectively computable.

As usual we shall assume the result of any gcd (greatest common divisor) computation in $\mathbf{K}[x]$ or $\mathbf{F}[x]$ as being normalized to a *monic* polynomial p, i.e., the leading coefficient $\mathrm{lcf}(p) = 1$.

By ϵ we denote the q-shift operator on $\mathbf{F}[x]$, i.e., $(\epsilon p)(x) = p(qx)$ for any $p \in \mathbf{F}[x]$. The extension of this shift operator to the rational function field $\mathbf{F}(x)$, the quotient field of $\mathbf{F}[x]$, will be also denoted by ϵ.

A polynomial $p \in \mathbf{F}[x]$ is said to be *q-monic* if $p(0) = 1$. Note that this property is invariant with respect to the q-shift operator, i.e., $(\epsilon p)(0) = p(0) = 1$, whereas usual monicness is invariant with respect to the shift operator E, i.e., $\mathrm{lcf}(Ep) = \mathrm{lcf}(p) = 1$.

The polynomial degree (in x) of any $p \in \mathbf{F}[x]$, $p \neq 0$, is denoted by $\deg(p)$. We define $\deg(0) := -1$.

Evidently, any polynomial $p \in \mathbf{F}[x]$ has a unique factorization, the *q-monic decomposition*, in the form

$$p = z \cdot x^{\alpha} \cdot \widehat{p}$$

where $z \in \mathbf{F}$, $\alpha \in \mathbf{N}$, and $\widehat{p} \in \mathbf{F}[x]$ q-monic. The q-monic decomposition can be computed easily; note also that α simply is the multiplicity of the root 0.

Besides the usual normalization with respect to monicness, it will be convenient to introduce normalization also with respect to q-monicness. In these instances we write "gcd_q" instead of "gcd", indicating that the gcd_q of two q-monic polynomials is understood to be q-monic.

More generally, if $p_1 = z_1 \cdot x^{\alpha_1} \cdot \widehat{p}_1$ and $p_2 = z_2 \cdot x^{\alpha_2} \cdot \widehat{p}_2$ are the q-monic decompositions of $p_1, p_2 \in \mathbf{F}[x]$, we define

$$\mathrm{gcd}_q(p_1, p_2) := \mathrm{gcd}(x^{\alpha_1}, x^{\alpha_2}) \cdot \mathrm{gcd}_q(\widehat{p}_1, \widehat{p}_2),$$

where the q-monic part is treated as described above.

Example. Let $p_1 = q\, x^2\, (1-x)(1-qx)$ and $p_2 = \epsilon p_1 = q^3\, x^2\, (1-qx)(1-q^2x)$, then

$$\mathrm{gcd}_q(p_1, p_2) = x^2(1-qx),$$

whereas $\mathrm{gcd}(p_1, p_2) = x^2(x - 1/q)$. □

Note that for any $p_1, p_2 \in \mathbf{F}[x]$:

$$\mathrm{gcd}(p_1, p_2) = 1 \Longleftrightarrow \mathrm{gcd}_q(p_1, p_2) = 1.$$

Definition. For any q-monic polynomial $p \in \mathbf{F}[x]$ and $k \in \mathbf{N}$ the *k-th falling q-factorial* $[p]_q^{\underline{k}}$ of p is defined as

$$[p]_q^{\underline{k}} := \prod_{i=0}^{k-1} \epsilon^{-i} p.$$

Note that by the null convention $\prod_{i \in \emptyset} p_i = 1$ we have $[p]_q^{\underline{0}} = 1$. In view of the commonly used q-shifted factorial, i.e., $(x;q)_k = [1 - q^{k-1}x]_q^{\underline{k}}$ it would be more natural to introduce the product in terms of rising powers of ϵ. The reason why we still stick to the "reverse" definition is that Paule's [1995] theory of greatest factorial factorization was presented exactly this way with E instead of ϵ, and we want to keep the parallels to the q-case as close as possible.

Polynomials arising in q-hypergeometric summation, in general have several different representations in terms of q-factorials.

Example. Let $p = (1-q^3x)(1-q^2x)(1-qx)^2(1-x)$, then $p = [1-qx]_q^{\underline{1}}\,[(1-q^3x)(1-qx)]_q^{\underline{2}} = [(1-qx)(1-x)]_q^{\underline{1}}\,[1-q^3x]_q^{\underline{3}} = [(1-q^3x)(1-qx)]_q^{\underline{1}}\,[1-q^2x]_q^{\underline{3}} = [1-qx]_q^{\underline{2}}\,[1-q^3x]_q^{\underline{3}} = [1-qx]_q^{\underline{1}}\,[1-q^3x]_q^{\underline{4}}$, etc. □

From all these possibilities the last one, which takes care of maximal chains is of particular importance. Informally, it can be obtained as follows: One selects irreducible factors of p in such a way that their product, say

$$q_1(x)\ q_1(q^{-1}x)\cdots q_1(q^{-k+1}x),$$

forms a falling q-factorial $[q_1]_q^{\underline{k}}$ of maximal length k. For the remaining irreducible factors of p this procedure is applied again in order to find all k-th falling q-factorial divisors $[q_1]_q^{\underline{k}}$, $[q_2]_q^{\underline{k}}$, etc., of that type. Then $[q_1\cdot q_2\cdots]_q^{\underline{k}}$ forms the q-factorial factor of p of maximal length k. Iterating this procedure one gets a factorization of p in terms of "greatest" q-factorial factors.

Definition. We say that $\langle p_1,\ldots,p_k\rangle$, $p_i \in \mathbf{F}[x]$, is a *qGFF-form* of a q-monic polynomial $p \in \mathbf{F}[x]$ if the following conditions hold:
(qGFF1) $p = [p_1]_q^{\underline{1}}\cdots[p_k]_q^{\underline{k}}$,
(qGFF2) each p_i is q-monic, and $k>0$ implies $\deg(p_k)>0$,
(qGFF3) $i\le j \Rightarrow \gcd_q([p_i]_q^{\underline{i}}, \epsilon p_j) = 1 = \gcd_q([p_i]_q^{\underline{i}}, \epsilon^{-j}p_j)$.

Note that, due to the null convention, $\langle\rangle$ is the qGFF-form of $1 \in \mathbf{F}[x]$. Condition (qGFF3) intuitively can be understood as prohibiting "overlaps" of q-factorials that violate length maximality.

The following theorem explicitly states the fact that the qGFF-form provides a canonical form. For instance, $\langle 1-qx, 1, 1, 1-q^3x\rangle$ is *the* qGFF-form of the polynomial p from the example above.

Theorem 1 *If $\langle p_1,\ldots,p_k\rangle$ and $\langle q_1,\ldots,q_l\rangle$ are* qGFF*-forms of a q-monic $p \in \mathbf{F}[x]$ then $k=l$ and $p_i = q_i$ for all $i \in \{1,\ldots,k\}$.*

Proof. The proof is entirely analogous to that of Theorem 2.1 in Paule [1995], and is thus left to the reader. □

From algorithmic point of view it is important to note that the qGFF form can be computed in an iterative manner essentially involving only gcd computations; see Section 3.2.

If $\langle p_1,\ldots,p_k\rangle$ is the qGFF-form of a q-monic $p \in \mathbf{F}[x]$ we also denote this fact for short by $q\mathrm{GFF}(p) = \langle p_1,\ldots,p_k\rangle$.

2.2 The Fundamental qGFF Lemma.

In hypergeometric summation (i.e., $q=1$) the gcd of a polynomial p and its shift Ep plays a fundamental role. The same is true for q-hypergeometric summation with respect to the q-shift operator ϵ instead of E. The qGFF-concept, as GFF in case $q=1$, takes special care of that observation. The mathematical and algorithmic essence lies in the following lemma.

Lemma 1 ("Fundamental qGFF Lemma") *Given a q-monic polynomial $p \in \mathbf{F}[x]$ with* qGFF*-form $\langle p_1,\ldots,p_k\rangle$. Then*

$$\gcd_q(p, \epsilon p) = [p_1]_q^{\underline{0}}\cdots[p_k]_q^{\underline{k-1}}.$$

Proof. Proceeding by induction on k the case $k = 0$ is trivial. For $k > 0$,

$$\gcd_q(p, \epsilon p) =$$
$$[p_k]_q^{\underline{k-1}} \cdot \gcd_q([p_1]_q^{\underline{1}} \cdots [p_{k-1}]_q^{\underline{k-1}} \cdot \epsilon^{-k+1} p_k, \epsilon([p_1]_q^{\underline{1}} \cdots [p_{k-1}]_q^{\underline{k-1}} \cdot p_k)) =$$
$$[p_k]_q^{\underline{k-1}} \cdot \gcd_q([p_1]_q^{\underline{1}} \cdots [p_{k-1}]_q^{\underline{k-1}}, \epsilon([p_1]_q^{\underline{1}} \cdots [p_{k-1}]_q^{\underline{k-1}})).$$

The first equality is obvious, the second is a consequence of (qGFF3) because for $i < k$ we have

$$\gcd_q([p_i]_q^{\underline{i}}, \epsilon p_k) = \gcd_q(\epsilon^{-k+1} p_k, \epsilon[p_i]_q^{\underline{i}}) = \epsilon(\gcd_q(\epsilon^{-k} p_k, [p_i]_q^{\underline{i}})) = 1$$

together with $\gcd_q(\epsilon^{-k+1} p_k, \epsilon p_k) | \gcd_q([p_k]_q^{\underline{k}}, \epsilon p_k) = 1$. The rest follows from applying the induction hypothesis. □

In other words, from the qGFF-form of p, i.e., $q\text{GFF}(p) = \langle p_1, \ldots, p_k \rangle$ one directly can extract the qGFF-form of its "$\gcd_q$-shift", i.e., $q\text{GFF}(\gcd_q(p, \epsilon p)) = \langle p_2, \ldots, p_k \rangle$.

Example. From $q\text{GFF}(p) = \langle 1 - qx, 1, 1, 1 - q^3 x \rangle$ one immediately gets by Lemma 1 that $\gcd_q(p, \epsilon p) = [1 - q^3 x]_q^{\underline{3}}$ and $q\text{GFF}(\gcd_q(p, \epsilon p)) = \langle 1, 1, 1 - q^3 x \rangle$. □

Note that dividing p with $q\text{GFF}(p) = \langle p_1, \ldots, p_k \rangle$ by $\epsilon^{-1} \gcd_q(p, \epsilon p)$ or $\gcd_q(p, \epsilon p)$ results in separating the product of the first, respectively last, falling q-factorial entries:

$$\frac{p}{\epsilon^{-1} \gcd_q(p, \epsilon p)} = p_1 p_2 \cdots p_k \quad \text{and} \quad \frac{p}{\gcd_q(p, \epsilon p)} = p_1 (\epsilon^{-1} p_2) \cdots (\epsilon^{-k+1} p_k).$$

Remark. For $q = 1$ the Fundamental GFF Lemma is formulated with respect to the shift operator E. In this version (see Paule [1995]) it forms a perfect analogue to the fundamental property

$$\gcd(p, Dp) = p_1^0 \, p_2^1 \cdots p_k^{k-1}$$

of the derivation operator D used in square-free factorization, i.e., for computing $p = p_1^1 \, p_2^2 \cdots p_k^k$, the square-free factorization of $p \in \mathbf{K}[x]$. □

3 q-Hypergeometric Telescoping

In this section we consider q-hypergeometric telescoping. In the general frame of his theory of difference field extensions, Karr [1981] was the first who provided an algorithmic answer to the problem. However, the theoretical setting is mighty and complex enough to deal also with other, non-hypergeometric types of indefinite summation problems, and, up to now, no working implementation is available.

Zeilberger (e.g., Petkovšek, Wilf and Zeilberger [1996]) has been the first who observed that the problem can be treated along the same lines as Gosper's algorithm. This approach was followed by Koornwinder [1993], who remarked "Surprisingly, Gosper's and Zeilberger's algorithms can be carried over to the q-case almost unchanged". Using GFF, defined with respect to the shift operator E, we presented a new and algebraically motivated approach to Gosper's algorithm; see Paule [1995]. Defining this type of polynomial factorization with respect to the q-shift operator ϵ, one is led to qGFF which enables to apply essentially the same argumentation, resulting in Algorithm qTelescope, now solving the problem of q-hypergeometric telescoping. In other words, the GFF concept, together with the translation into its q-version qGFF, somehow spoils Koornwinder's surprise, at least with respect to Gosper's algorithm.

3.1 The Algorithm qTelescope. In this section we present the algorithm qTelescope which can be viewed as a q-analogue of Gosper's algorithm. After specifying the problem, we demonstrate that all what is needed consists only in a slight adoption of the arguments presented in Section 5 of Paule [1995].

A sequence $\langle f_k\rangle_{k\geq 0}$ is called *q-hypergeometric* over $\mathbf{F}$ if there exists a rational function $\rho\in\mathbf{F}(x)$ such that $f_{k+1}/f_k=\rho(q^k)$ for all $k\in\mathbf{N}$. Given q-hypergeometric $\langle f_k\rangle_{k\geq 0}$, the problem of *$q$-hypergeometric telescoping* is to find a q-hypergeometric solution $\langle g_k\rangle_{k\geq 0}$ of

$$g_{k+1}-g_k=f_k. \tag{1}$$

Assume that a q-hypergeometric solution $\langle g_k\rangle_{k\geq 0}$ of (1) exists. Let $\sigma\in\mathbf{F}(x)$ be such that $g_{k+1}/g_k=\sigma(q^k)$ for all $k\in\mathbf{N}$, then evidently

$$g_k=\tau(q^k)\cdot f_k, \tag{2}$$

where $\tau(x)=1/(\sigma(x)-1)\in\mathbf{F}(x)$.

For any integer α we define $\alpha_+:=\max(\alpha,0)$, and $\alpha_-:=\max(-\alpha,0)$.

By relation (2), eq. (1) is equivalent to

$$z\cdot x^{\alpha_+}\cdot a\cdot\epsilon\tau-x^{\alpha_-}\cdot b\cdot\tau=x^{\alpha_-}\cdot b, \tag{3}$$

where $\rho=z\cdot x^{\alpha}\cdot a/b$ with $z\in\mathbf{F}$, α integer, and $a,b\in\mathbf{F}[x]$ relatively prime and q-monic.

Vice versa, any rational solution $\tau\in\mathbf{F}(x)$ of (3) gives rise to a q-hypergeometric solution $g_k:=\tau(q^k)\cdot f_k$ of (1). This means, q-hypergeometric telescoping is equivalent to finding a *rational* solution τ of (3).

Any $\tau\in\mathbf{F}(x)$ can be represented as the quotient of relatively prime polynomials in the form $\tau=u/v$ where $u,v\in\mathbf{F}[x]$ with $v=x^{\beta}\cdot\widehat{v}$ the q-monic decomposition of v. In case such a solution τ of (3) exists, assume we know v or a multiple $V\in\mathbf{F}[x]$ of v. Then by clearing denominators in

$$z\cdot x^{\alpha_+}\cdot a\cdot\frac{\epsilon U}{\epsilon V}-x^{\alpha_-}\cdot b\cdot\frac{U}{V}=x^{\alpha_-}\cdot b,$$

the problem reduces further to finding a polynomial solution $U\in\mathbf{F}[x]$ of the resulting difference equation with polynomial coefficients,

$$z\cdot x^{\alpha_+}\cdot a\cdot V\cdot\epsilon U-x^{\alpha_-}\cdot b\cdot(\epsilon V)\cdot U=x^{\alpha_-}\cdot b\cdot V\cdot\epsilon V. \tag{4}$$

(Note that at least one polynomial solution, namely $U=u\cdot V/v$, exists.) Furthermore, equations of that type simplify by canceling $\gcd_q$'s. For instance, in order to get more information about the denominator v, let $v_i:=\epsilon^i v/\gcd_q(v,\epsilon v)$, $i\in\{0,1\}$. Then (3) is equivalent to

$$z\cdot x^{\alpha_+}\cdot a\cdot v_0\cdot\epsilon u-x^{\alpha_-}\cdot b\cdot v_1\cdot u=x^{\alpha_-}\cdot b\cdot v_0\cdot v_1\cdot\gcd_q(v,\epsilon v). \tag{5}$$

Now, if $\langle p_1,\ldots,p_m\rangle$, $m\geq 0$, is the qGFF-form of $\widehat{v}$, it follows from $\gcd(u,v)=1=\gcd(v_0,v_1)$ and the Fundamental qGFF Lemma that

$$v_0=(\epsilon^0 p_1)\cdots(\epsilon^{-m+1}p_m)|b \quad\text{and}\quad v_1=q^{\beta}\cdot(\epsilon p_1)\cdots(\epsilon p_m)|a.$$

This observation gives rise to a simple and straightforward algorithm for computing a multiple $\widehat{V}:=[P_1]_q^{\underline{1}}\cdots[P_n]_q^{\underline{n}}$ of $\widehat{v}$. For instance, if $P_1:=\gcd_q(\epsilon^{-1}a,b)$ then obviously $p_1|P_1$. Indeed, we shall see below that by exploiting properties (qGFF1), (qGFF2), and (qGFF3) from Section 2.1, one can iteratively extract q-monic p_i-multiples P_i such that $\epsilon P_i|a$ and $\epsilon^{-i+1}P_i|b$:

Algorithm $\widehat{\mathrm{V}}$MULT. INPUT: relatively prime and q-monic polynomials $a, b \in \mathbf{F}[x]$ that constitute the q-monic quotient of $\rho = z \cdot x^{\alpha} \cdot a/b \in \mathbf{F}(x)$; OUTPUT: q-monic polynomials $\langle P_1, \ldots, P_n \rangle$ such that $\widehat{V} := [P_1]^{\frac{1}{q}} \cdots [P_n]^{\frac{n}{q}}$ is a multiple of $\widehat{v}$, the q-monic part of the denominator $v = x^{\beta} \cdot \widehat{v}$ of $\tau \in \mathbf{F}(x)$.

(i) Compute $n = \min\{j \in \mathbf{N} | \gcd(\epsilon^{-1}a, \epsilon^{k-1}b) = 1 \text{ for all integers } k > j\}$.

(ii) Set $a_0 = a$, $b_0 = b$, and compute for i from 1 to n:

$$\begin{aligned} P_i &= \gcd_q(\epsilon^{-1}a_{i-1}, \epsilon^{i-1}b_{i-1}), \\ a_i &= a_{i-1}/\epsilon P_i, \\ b_i &= b_{i-1}/\epsilon^{-i+1}P_i. \end{aligned}$$

The following lemma tells that the output polynomials P_i indeed are multiples of the p_i's.

Lemma 2 *Let $\tau \in \mathbf{F}(x)$ be a rational function solution of eq.* (3) *such that $\tau = u/v$ with relatively prime $u, v \in \mathbf{F}[x]$ and $v = x^{\beta} \cdot \widehat{v}$ the q-monic decomposition of v. Let $q\mathrm{GFF}(\widehat{v}) = \langle p_1, \ldots, p_m \rangle$. If n and $\langle P_1, \ldots, P_n \rangle$ are computed as in Algorithm* $\widehat{\mathrm{V}}$MULT*, then:*

$$n \geq m \text{ and } p_i | P_i \text{ for all } i \in \{1, \ldots, m\}.$$

Proof. In its essence the proof only uses properties (qGFF1), (qGFF2), and (qGFF3); almost word by word the argumentation can be carried over from the case $q = 1$. For that see Algorithm VMULT and the proof of Lemma 5.1 in Paule [1995]. Hence we leave the steps of the verification to the reader. □

With the multiple $\widehat{V}$ of $\widehat{v}$ in hands, all what is left for solving (3), and thus the q-hypergeometric telescoping problem (1), is to determine an appropriate multiplicity γ such that

$$V = x^{\gamma} \cdot \widehat{V} \text{ is a multiple of } v = x^{\beta} \cdot \widehat{v}.$$

For that we consider eq. (5) again: (i) Assume that either $\alpha_- \neq 0$ or $\alpha_+ \neq 0$. In the first case we have $\alpha_+ = 0$ and $x^{\alpha_-}|u$, hence β must be 0 because of $\gcd(u, v) = 1$. This means, we can choose $\gamma := 0$. In the second case we have $\alpha_- = 0$ and $x^{min(\alpha_+, \beta)}|u$, because of $\gcd_q(v, \epsilon v) = x^{\beta} \cdot \gcd_q(\widehat{v}, \epsilon\widehat{v})$. Again β must be 0, and again we can choose $\gamma := 0$. (ii) Assume that $\alpha = 0$. In this case eq. (5) evaluated at $x = 0$ turns into

$$(z - q^{\beta})\, u(0) = q^{\beta} \cdot \delta_{0,\beta},$$

because $v_1 = q^{\beta} \cdot (\epsilon p_1) \cdots (\epsilon p_m)$; $\delta_{0,\beta}$ denotes the Kronecker symbol. This means, if $\beta > 0$ we obtain, observing that $u(0) \neq 0$ in this case, as a condition for β that $z = q^{\beta}$. Hence in case $\alpha = 0$, we choose $\gamma := \log_q(z)$ if z is a positive integer power of q, or $\gamma := 0$ otherwise.

Summarizing, q-hypergeometric telescoping can be decided constructively as follows:

Algorithm qTelescope. INPUT: $\rho \in \mathbf{F}(x)$ such that $f_{k+1}/f_k = \rho(q^k)$ for all $k \in \mathbf{N}$; OUTPUT: a q-hypergeometric solution $\langle g_k \rangle_{k \geq 0}$ of (1); in case such a solution does not exist, the algorithm stops.

(i) Decompose ρ into the form $\rho = z \cdot x^{\alpha} \cdot a/b$ such that $z \in \mathbf{F}$, α integer, and $a, b \in \mathbf{F}[x]$ relatively prime and q-monic.

(ii) With respect to input a, b compute q-monic polynomials $\langle P_1, \ldots, P_n \rangle$ by Algorithm $\widehat{\mathrm{V}}$MULT and set $\widehat{V} := [P_1]^{\frac{1}{q}} \cdots [P_n]^{\frac{n}{q}}$.

(iii) Determine the zero-root multiplicity γ as follows:

$$\gamma := \begin{cases} \log_q(z) & \text{if } \alpha = 0 \text{ and } z \text{ a positive integer power of } q, \\ 0 & \text{otherwise.} \end{cases}$$

(iv) Take $V := x^\gamma \cdot \widehat{V}$. If eq. (4) can be solved for a polynomial $U \in \mathbf{F}[x]$ then $g_k := f_k \cdot U(q^k)/V(q^k)$ solves (1), if eq. (4) admits no polynomial solution U then no q-hypergeometric solution of (1) exists.

We want to note that in practical computations eq. (4) is solved in a more economical but equivalent form, which is obtained by dividing out gcd's; see eq. (7). A standard way to solve, for $Y \in \mathbf{F}[x]$, such a q-difference equation $Q \cdot \epsilon Y - R \cdot Y = P$, with polynomial coefficients from $\mathbf{F}[x]$, is the method of undetermined coefficients. For that one needs a careful analysis of possible degree bounds for Y which has been done for the Mathematica implementation of qTelescope.

3.2 Remarks on Canonical Forms. Algorithm $\widehat{\text{V}}$MULT is the key ingredient of Algorithm qTelescope. The $q = 1$ version of $\widehat{\text{V}}$MULT is essentially the same algorithm Petkovšek [1992] came up with in order to compute a canonical "Gosper-form" representation; the q-version described by Abramov, Paule and Petkovšek [1995] was used to formulate a q-analogue of this form. Lemma 2 focuses on the $p_i|P_i$ property; the following lemma lists additional facts about Algorithm $\widehat{\text{V}}$MULT which can be proved in a similar fashion:

Lemma 3 *Let n, a_n, b_n, and the tuple $\langle P_1, \ldots, P_n\rangle$ be computed as in Algorithm $\widehat{\text{V}}$MULT, then:*

(i) $a = (\epsilon P_1) \cdots (\epsilon P_n) \cdot a_n$,

(ii) $b = (\epsilon^0 P_1) \cdots (\epsilon^{-n+1} P_n) \cdot b_n$,

(iii) $\forall k \in \mathbf{N}$: $\gcd(a_n, \epsilon^k b_n) = 1$,

(iv) $\forall i \in \{1, \ldots, n\}$: $\gcd([P_i]_q^{\frac{i}{q}}, a_n) = 1$,

(v) $\forall i \in \{1, \ldots, n\}$: $\gcd([P_i]_q^{\frac{i}{q}}, \epsilon^{-1} b_n) = 1$,

(vi) $q\text{GFF}([P_1]_q^{\frac{1}{q}} \cdots [P_n]_q^{\frac{n}{q}}) = \langle P_1, \ldots, P_n\rangle$.

Proof. Left to the reader; cf. Paule [1995] or Abramov, Paule and Petkovšek [1995]. □

The intended scope of this paper prohibits a more detailed discussion of the numerous interesting features of Algorithm $\widehat{\text{V}}$MULT; we briefly touch two which are of particular interest.

By Lemma 3 we have

$$\frac{a}{b} = \frac{\epsilon \widehat{V}}{\widehat{V}} \cdot \frac{a_n}{b_n}. \tag{6}$$

Using this representation for a/b, the task to solve eq. (4) for U reduces to solve

$$z \cdot q^{-\gamma} \cdot x^{\alpha_+} \cdot a_n \cdot \epsilon W - x^{\alpha_-} \cdot (\epsilon^{-1} b_n) \cdot W = x^{\gamma + \alpha_-} \cdot \widehat{V} \tag{7}$$

for $W \in \mathbf{F}[x]$; then $U = (\epsilon^{-1} b_n) \cdot W$.

More generally, any nonzero $\rho \in \mathbf{F}(x)$ has a *unique* representation of the form

$$\rho = \frac{\epsilon P}{P} \cdot \frac{Q}{\epsilon R}, \tag{8}$$

where the polynomials $P, Q, R \in \mathbf{F}[x]$, being normalized in a certain way, are such that $\gcd(P, Q) = \gcd(P, R) = 1$, and $\gcd(Q, \epsilon^k R) = 1$ for all $k \geq 1$. A representation of this type which provides a perfect q-analogue of Petkovšek's canonical

Gosper-form, was introduced by Paule and Strehl [1995]; eq. (6) simply is its restriction to the q-monic quotient of ρ. Another normalization of P, Q, and R was chosen by Abramov, Paule and Petkovšek [1995].

Paule and Strehl [1995] described, without giving detailed proofs, that the form (8), as in case $q = 1$ contains all the information needed for solving q-hypergeometric telescoping as follows: If

$$P = Q \cdot \epsilon Y - R \cdot Y \tag{9}$$

admits a solution $Y \in \mathbf{F}[x]$, then $C/D = ((\epsilon Y)/Y) \cdot (Q/R) \in \mathbf{F}(x)$ is the rational representation of the q-hypergeometric solution, i.e.,

$$g_k = \frac{D(q^k)}{C(q^k) - D(q^k)} \cdot f_k$$

solves (1).

One can easily verify that Algorithm qTelescope is equivalent to this procedure; for instance, eq. (7) is equivalent to eq. (9). It is this canonical form version of qTelescope which was taken for our Mathematica implementation, i.e., computing the form (8) with normalization as explained by Paule and Strehl [1995]; see also Section 4.2.7.

We want to conclude this section by pointing to another important computational feature of Algorithm $\widehat{\mathrm{V}}$MULT. Namely, in view of Lemma 3 and representation (6), it can be used to compute the (canonical) qGFF-form $q\mathrm{GFF}(P) = \langle P_1, \ldots, P_n \rangle$ for given q-monic $P \in \mathbf{F}[x]$: simply set $a = \epsilon P / \gcd_q(P, \epsilon P)$ and $b = P / \gcd_q(P, \epsilon P)$.

4 A q-Analogue of Zeilberger's Algorithm

As demonstrated in Wilf and Zeilberger [1992b], Zeilberger's [1990] fast algorithm for proving terminating hypergeometric identities generalizes to the q-case.

In this section we first give a short account of the underlying mechanism, especially of the fundamental notion of "proof certification"; for further details the reader is referred to the beautiful book by Petkovšek, Wilf and Zeilberger [1996]. This is followed by a description of the usage of the package `qZeil`, a q-analogue of Zeilberger's algorithm in the computer algebra system Mathematica. Its implementation, based on q-hypergeometric telescoping introduced in Section 3, was carried out by A. Riese in the frame of a diploma thesis supervised by the first named author of this paper. The section concludes by a brief comparison of `qZeil` with Koornwinder's [1993] Maple package.

4.1 qWZ-Certification. Analogous to Zeilberger's [1990] algorithm its q-analogue takes terminating q-hypergeometric sums as input; the output is a linear recurrence that is satisfied by the input sum, together with a rational function which serves as the proof certificate. It is important to note that the proof certificate enables a verification of the output recurrence merely by checking a rational function identity. This means, the algorithm itself supplies complete information for a correctness check which works independently of the steps in which the output recurrence was manufactured.

Let $f := \langle f_{n,k} \rangle$ be a double-indexed sequence with values in $\mathbf{F}$. We shall consider only sequences where n runs through $\mathbf{N}$, whereas the second parameter k might run through all integers.

The sequence f is called q-hypergeometric in both parameters if both quotients

$$\frac{f_{n,k}}{f_{n-1,k}} \quad \text{and} \quad \frac{f_{n,k}}{f_{n,k-1}}$$

are rational functions in q^n and q^k over $\mathbf{F}$ for all n and k where the quotients are well-defined.

Recall the standard definition of the q-shifted factorial of $a \in \mathbf{F}$:

$$(a;q)_k = \begin{cases} (1-a)(1-aq)\cdots(1-aq^{k-1}), & \text{if } k>0,\\ 1, & \text{if } k=0,\\ 1/((1-aq^{-1})(1-aq^{-2})\cdots(1-aq^{k})), & \text{if } k<0, \end{cases}$$

and

$$(a;q)_\infty = \prod_{k=0}^{\infty}(1-aq^k).$$

Then the sequence of Gaussian polynomials $\begin{bmatrix} n \\ k \end{bmatrix}_q = (q;q)_n/((q;q)_k(q;q)_{n-k})$, if $0 \le k \le n$, and $\begin{bmatrix} n \\ k \end{bmatrix}_q = 0$ otherwise, is q-hypergeometric in n and k.

We say the sequence f has finite support with respect to k, if the following is true for each n fixed: $f_{n,k} \neq 0$ for all k from a finite integer interval I_n, and $f_{n,k} = 0$ for all k outside I_n; for example, $f_{n,k} := \begin{bmatrix} n \\ k \end{bmatrix}_q$ with $I_n = \{0, 1, \ldots, n\}$.

Given $f = \langle f_{n,k} \rangle$ q-hypergeometric in n and k, one can prove under mild side-conditions, as demonstrated in Wilf and Zeilberger [1992b], that for a certain integer $d \ge 0$ and $n \ge d$ there exists a linear recurrence

$$c_0(n) f_{n,k} + c_1(n) f_{n-1,k} + \cdots + c_d(n) f_{n-d,k} = g_{n,k} - g_{n,k-1}, \tag{10}$$

where the coefficients are polynomials in q^n not depending on k, and where $g_{n,k}$ is a rational function multiple of $f_{n,k}$ and thus also q-hypergeometric in n and k. Given the order d, which in general is not known a priori, $g_{n,k}$ and also the coefficient polynomials $c_i(n)$ are determined by q-hypergeometric telescoping, i.e., by Algorithm qTelescope.

Assume that f has finite support with respect to k. Then summing both sides of (10) over all k results in

$$c_0(n) S_n + c_1(n) S_{n-1} + \cdots + c_d(n) S_{n-d} = 0, \tag{11}$$

a recurrence for the sum sequence $S_n := \sum_k f_{n,k}$, a finite sum due to the finite support property. We use the convention that the summation parameter k runs through all the integers, in case the summation range is not specified explicitly.

Now the qWZ-*certificate* (for short: certificate) of recurrence (10) or (11), respectively, by definition is the rational function $cert(n,k)$, rational in q^n and q^k, such that

$$g_{n,k} = cert(n,k)\ f_{n,k}.$$

Evidently, with the certificate in hands the verification of (10), and therefore (11), reduces to checking the *rational function identity*

$$r(n,k) = cert(n,k) - cert(n,k-1)\frac{f_{n,k-1}}{f_{n,k}}, \tag{12}$$

where $r(n,k)$, rational in q^n and q^k, comes from rewriting the left hand side of (10) as $r(n,k)\ f_{n,k}$. The computation of $r(n,k)$ is straightforward, because any $f_{n-i,k}$ can be written as a rational function multiple of $f_{n,k}$, for instance, $f_{n-1,k} = (f_{n-1,k}/f_{n,k}) \cdot f_{n,k}$.

We conclude this section with a remark on the inhomogeneous case which also arises in applications. Assume that f does not have finite support, or, that one is interested in summation with bounds not naturally induced by the finite support, for instance,

$$S_n := \sum_{k=ln+m}^{on+p} f_{n,k}$$

with fixed integers $l, m, o,$ and p. In this case the package delivers an output recurrence which is inhomogeneous of type

$$c_0(n)S_n + c_1(n)S_{n-1} + \cdots + c_d(n)S_{n-d} = g_{n,on+p} - g_{n,ln+m-1} + ct(n),$$

where the corresponding correction term $ct(n)$ is defined as

$$ct(n) := \sum_{j=1}^{d} c_j(n)[ct_1(j,n) - ct_2(j,n)],$$

with

$$ct_1(j,n) := \sum_{k=l(n-j)+m}^{ln+m-1} f_{n-j,k} \quad \text{and} \quad ct_2(j,n) := \sum_{k=o(n-j)+p+1}^{on+p} f_{n-j,k},$$

employing the extended sum-definition

$$\sum_{k=a}^{b} f_k = \begin{cases} f_a + f_{a+1} + \cdots + f_b, & a \leq b, \\ 0, & a = b+1, \\ -[f_{b+1} + f_{b+2} + \cdots + f_{a-1}], & a \geq b+2. \end{cases}$$

An example for a nontrivial application of this "general bounds" feature of `qZeil` is presented in Section 5.4.

4.2 The Package `qZeil`.

4.2.1 *Installation.* The package consists of five files named `qZeil.m`, `qTelescope.m`, `qInput.m`, `qSimplify.m` and `LinSolve.m`, which have to be copied into one directory. The files are available by email request to

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

After starting a Mathematica session from this directory and typing `<<qZeil.m` all files are loaded automatically. In addition to these files containing the code for the algorithm, the ASCII-file `qZeilExamples.txt`, consisting of about 250 identities at the moment, can be used as a source of examples. Most of these identities are terminating summations or transformations from the book by Gasper and Rahman [1990]. The package `qZeil` already helped to improve this outstanding q-hypergeometric database; see Riese [1995].

4.2.2 *Interfaces.* The package has two interfaces. Concerning indefinite summation one can run `qTelescope` for q-hypergeometric telescoping, or `qZeil`, the q-analogue of Zeilberger's algorithm, to come up with a recurrence, homogeneous or inhomogeneous, for a definite q-hypergeometric sum. The corresponding commands are given by

```
qTelescope[SUMMAND, RANGE, <INTCONST>]
```

and

```
qZeil[SUMMAND, RANGE, RECVAR, ORDER, <INTCONST>],
```

where <PARAMETER> denotes an optional argument. Before we will give a detailed description of the parameters, let us first present some illustrating examples.

4.2.3 *Warm-up Examples.*

1. Load the package:

```
In[1]:= <<qZeil.m
Out[1]= Axel Riese's q-Zeilberger implementation version
        1.4 loaded
```

2. Compute the closed form for a special case of the q-Chu-Vandermonde summation formula

$$\sum_{k=0}^{n} \frac{(b;q)_k}{(q;q)_k}\, q^k = \frac{(bq;q)_n}{(q;q)_n}.$$

```
In[2]:= qTelescope[qfac[b,q,k] q^k / qfac[q,q,k], {k,0,n}]
        qfac[b q, q, n]
Out[2]= ---------------
         qfac[q, q, n]
```

3. The so-called Rogers-Szegö polynomials

$$r_n(x,a) := \sum_{k=0}^{n} \begin{bmatrix} n \\ k \end{bmatrix}_q a^{n-k} x^k$$

are known to satisfy a recurrence of order 2:

```
In[3]:= qZeil[qBinomial[n,k,q] a^(n-k) x^k, {k,0,n}, n, 2]
                                   -1 + n
Out[3]= SUM[n] == a (-1 + q      ) x SUM[-2 + n] +

         (a + x) SUM[-1 + n]
```

4. For Jackson's q-analogue of the Pfaff-Saalschütz formula

$${}_3\phi_2(q^{-n}, a, b; c, abc^{-1}q^{1-n}; q, q) = \frac{(c/a, c/b; q)_n}{(c, c/ab; q)_n}$$

we obtain:

```
In[4]:= qZeil[qfac[q^(-n),q,k] qfac[a,q,k] qfac[b,q,k] q^k /
             (qfac[c,q,k] qfac[a b / c q^(1-n),q,k] *
              qfac[q,q,k]), {k,0,n}, n, 1]
                            -1 + n          -1 + n
                         c q             c q
                   (1 - ---------) (1 - ---------) SUM[-1 + n]
                            a               b
Out[4]= SUM[n] == -------------------------------------------
                                                 -1 + n
                                  -1 + n      c q
                        (1 - c q      ) (1 - ---------)
                                                a b
```

4.2.4 *The Input Summand.* As summand we allow any q-hypergeometric term of the form

$$f_{n,k} = \frac{\prod_{r=1}^{rr}(A_r\, q^{(d_r i_r)n+(e_r i_r)k+l_r}; q^{i_r})_{a_r n+b_r k+c_r}}{\prod_{s=1}^{ss}(B_s\, q^{(f_s j_s)n+(g_s j_s)k+m_s}; q^{j_s})_{u_s n+v_s k+w_s}} \cdot R(q^n, q^k)\cdot q^{\alpha\binom{k}{2}+(\beta n+\gamma)k}\cdot z^k, \qquad (13)$$

with

A_r, B_s	power products in $\mathbf{K}$,
a_r, b_r, u_s, v_s	specific integers (i.e., integers free of any parameters),
c_r, w_s	integers, which may depend on parameters free of n and k,
d_r, e_r, f_s, g_s	specific integers,
l_r, m_s	integers free of n and k,
i_r, j_s	specific nonzero integers,
R	a rational function in $\mathbf{F}(q^n, q^k)$ such that the denominator factors completely into a product of terms of the form $(1 - Bq^{fn+gk+m})$,
α, β, γ	specific integers, and
z	a rational function in $\mathbf{F}$.

The q-shifted factorial $(a;q^i)_k$ has to be typed as `qfac[a,q^i,k]`. In addition we allow terms of the form `qBrackets[a,q]` for $[a]_q := (1-q^a)/(1-q)$, `qFactorial[a,q]` for $[a]_q! := [1]_q\,[2]_q \cdots [a]_q$, and `qBinomial[a,b,q]` for $\begin{bmatrix} a \\ b \end{bmatrix}_q$, provided that those expressions can be translated correctly — with respect to (13) — into terms of q-shifted factorials as with `qfac` also for these forms powers `q^i` are admitted.

4.2.5 *The Summation Range.* The range of summation has to be specified in the form

`RANGE := {SUMVAR, LOW, UPP}`.

In `qTelescope`, `LOW` and `UPP` may be arbitrary integers or integer parameters free of `SUMVAR` satisfying `LOW` $\leq$ `UPP`. In `qZeil`, `LOW` and `UPP` are linear integer functions in `RECVAR` being free of `SUMVAR` such that `LOW` $\leq$ `UPP`.

In `qZeil` the user may specify one or both bounds to be $\pm$`Infinity`. In this case, the bounds are assumed to be naturally induced by the finite support. The algorithm runs considerably faster in this *Turbo*-mode, since no inhomogeneous part and no correction terms of the recurrence have to be computed.

4.2.6 *The Optional Argument* `INTCONST`. Since Mathematica is not able to handle typed variables, it is necessary to simulate them by telling the system explicitly which indeterminates should be treated as nonnegative integer constants. If one assigns to the optional argument `INTCONST` a list of Mathematica symbols representing those indeterminates, the program will assume them to be nonnegative integers. This also improves the simplification abilities of the program.

Consider the following example. Suppose we want to find a closed form for the indefinite sum

$$\sum_{k=0}^{n} \begin{bmatrix} m+k \\ k \end{bmatrix}_q q^k.$$

Without any knowledge about m the program is not able to recognize m and $m+k$ in $(q;q)_m$ and $(q;q)_{m+k}$, respectively, as integers. The problem disappears if we make the assignment `INTCONST := {m}`.

```
In[6]:= qTelescope[qBinomial[m+k,k,q] q^k, {k, 0, n}, {m}]
Out[6]= qBinomial[1 + m + n, 1 + m, q]
```

Note that in `qZeil` and `qTelescope` all indeterminates appearing in the bounds as well as the recursion variable `RECVAR` (in `qZeil`) are assumed to be elements of `INTCONST` automatically.

4.2.7 *The Certificate and other Global Variables.* The (simplified) certificate $cert(n,k)$, i.e., the rational function from Section 4.1 such that $g_{n,k} = cert(n,k) \cdot f_{n,k}$, is delivered by calling the function `Cert` without any parameters. For example, after obtaining `Out[4]` type

```
In[5]:= Cert
           1 - k + n          k           k    k    n
        c q          (-1 + a q ) (-1 + b q ) (q  - q )
Out[5]= ----------------------------------------------
                 n          n           1 + k       n
         (-1 + q ) (-q + c q ) (-(a b q     ) + c q )
```

and the proof of Jackson's q-analogue of the Pfaff-Saalschütz formula reduces to checking the initial values at $n = 0$ and verifying the rational function equation (12) for this value of $cert(n,k)$.

The output behavior of the program can be influenced by the global Boolean variables `Talk` and `Output`.

If `Talk` is set to `True`, the user can see explicitly which step of the algorithm is executed at the moment. This is mainly thought for time-consuming examples. Default value for `Talk` is `False`.

If `Output` is set to `True`, then running `qTelescope` or `qZeil` generates the file `GoOut`, where some intermediate results of the actual computation are written to. Default value for `Output` is `True`.

The entries `P-factor P_fac(T)`, `Q-numerator Q_num(T)` and `R-denominator R_den(T)` in `GoOut`, where T is used as an abbreviation for q^k, correspond to polynomials $\bar{P}, \bar{Q}$ and $\bar{R}$, respectively, constituting a slightly modified version

$$\frac{f_k}{f_{k-1}} = \frac{\bar{P}}{\epsilon^{-1}\bar{P}} \, \frac{\bar{Q}}{\bar{R}},$$

of the q-analogue (8) of Petkovšek's canonical Gosper-form, normalized as explained in Paule and Strehl [1995]. This is because in practice it turns out to be more convenient to start out with the rational function f_k/f_{k-1}, and to solve $f_k = g_k - g_{k-1}$ instead of the version (1). Hence in the actual implementation of qTelescope, eq. (9) is solved in the correspondingly modified version.

Finally, by setting the global variable `Simp` to `False` one can suppress the automatic simplification of the telescoping solution sequence $g_{n,k}$ and the correction terms. By default, the program applies the rules listed in the file `qSimplify.m` to those expressions. Since the size of the result may grow enormously, this should be done only in case of emergency.

4.2.8 *Computing Companion Identities.* As WZ-pairs in the hypergeometric case (see Petkovšek, Wilf and Zeilberger [1996]), qWZ-pairs play an important role in q-certification. Analogous to case $q = 1$, one can use qWZ-pairs to get new identities "for free", i.e., without too much additional effort. Following Wilf and Zeilberger [1992a], one of these identities "for free" is called the *companion identity.*

It is based on the symmetry of f and g in the qWZ-equation

$$f_{n,k} - f_{n-1,k} = g_{n,k} - g_{n,k-1}.$$

Double-indexed sequences of this type, which in addition are q-hypergeometric in both indices, are called qWZ-pairs. As in case $q = 1$, one has: If f and g form a qWZ-pair satisfying the following conditions:

(F) for each integer k, the limit $f_k := \lim_{n\to\infty} f_{n,k}$ exists and is finite,

(G) $\lim_{k\to-\infty} \sum_{n\geq 0} g_{n+1,k} = 0$,

then the companion identity is given by

$$\sum_{n\geq 0} g_{n+1,k} = \sum_{j\leq k} (f_j - f_{0,j}).$$

The program computes the companion identity, if the global variable `Companion` is set to `True`, and f and g in fact form a qWZ-pair. To compute f_k we might have to make the assumption $|q| < 1$, or, to take the limit w.r.t. sequences of formal power (or Laurent) series. The condition (G) has to be checked by the user. Note that (G) is satisfied automatically if f has finite support with respect to k. Default value for `Companion` is `False`, the result is assigned to the variable `CompId`.

For example, for the q-Chu-Vandermonde identity

$$\sum_{k=0}^{n} \begin{bmatrix} n \\ k \end{bmatrix}_q \begin{bmatrix} b \\ k \end{bmatrix}_q q^{k^2} = \begin{bmatrix} b+n \\ n \end{bmatrix}_q,$$

we obtain the following result:

```
In[7] := Companion = True;
         qZeil[qBinomial[n,k,q] qBinomial[b,k,q] q^(k^2) /
               qBinomial[b+n,n,q], {k,0,n}, n, 1, {b}]
Out[8]= SUM[n] == 1

In[9]:= CompId
                             2
              1 + k + k  + n                                  2
Out[9]= Sum[-((q                 qBinomial[n, k, q] qfac[q, q, b]

          qfac[q, q, n]) / (qfac[q, q, -1 + b - k] qfac[q, q, k]

            qfac[q, q, 1 + b + n])), {n, 0, Infinity}] ==


                            2
                          jj
                         q    qBinomial[b, b - jj, q] qfac[q, q, b]
     -(k >= 0) + Sum[-----------------------------------------,
                                     qfac[q, q, jj]

       {jj, -Infinity, k}]
```

Here we follow the convention that for any true-or-false predicate *pred* we define $(pred) := 1$ if *pred* is true, and $(pred) := 0$ if *pred* is false.

Hence, for $b \geq 0$ and $k \geq 0$ the companion identity reads as (cf. Riese [1995])

$$\frac{q^{1+k+k^2}(1-q^{k+1})}{(1-q^{b+1})} \begin{bmatrix} b \\ k+1 \end{bmatrix}_q \cdot \sum_{n \geq k} \frac{\begin{bmatrix} n \\ k \end{bmatrix}_q}{\begin{bmatrix} n+b+1 \\ n \end{bmatrix}_q} q^n = 1 - \sum_{j=0}^{k} \frac{\begin{bmatrix} b \\ j \end{bmatrix}_q (q;q)_b}{(q;q)_j} q^{j^2}.$$

For the special case $b = n$ of the q-Chu-Vandermonde identity we get the result spelled out in Wilf and Zeilberger [1992b].

4.2.9 *Computing Dual Identities.* Another method for discovering new identities is based on the fact that to any qWZ-pair one can associate a dual pair that may produce new identities.

As in the $q = 1$ case (cf. Wilf and Zeilberger [1992a], or Gessel [1995] who made a systematic investigation of dual identity production), one introduces the operation of *shadowing*. Let, for instance, $a_n = (q;q)_n$ for $n \geq 0$. Then the defining property of a_n is that it satisfies the recurrence equation $a_n = (1-q^n)\, a_{n-1}$ together with the initial condition $a_0 = 1$. Trying to extend this sequence to the "opposite side", one could ask for a sequence $\bar{a}_n$ such that $\bar{a}_n = (1-q^n)\, \bar{a}_{n-1}$ holds for the negative integers. A sequence that satisfies this condition is

$$\bar{a}_n = \frac{(-1)^n q^{\binom{n+1}{2}}}{(q;q)_{-n-1}} \quad \text{for } n \leq -1.$$

We call $\bar{a}_n$ the *shadow* of a_n. More generally, for $a_{n,k} = (\alpha; q^i)_{an+bk+c}$, where $\alpha = Aq^{din+eik+l}$, the shadow is defined by

$$\bar{a}_{n,k} = \frac{(-1)^{an+bk+c} \alpha^{an+bk+c+1} q^{i\left[\binom{an+bk+c}{2}-1\right]}}{(q^{2i}/\alpha; q^i)_{-an-bk-c-1}},$$

with the property that

$$\frac{a_{n,k}}{a_{n-1,k}} = \frac{\bar{a}_{n,k}}{\bar{a}_{n-1,k}} \quad \text{and} \quad \frac{a_{n,k}}{a_{n,k-1}} = \frac{\bar{a}_{n,k}}{\bar{a}_{n,k-1}}.$$

The shadow $\bar{f}_{n,k}$ of a summand term $f_{n,k}$ (see Section 4.2.4) is defined to be the result of formally replacing each factor of the form $(A;q^d)_{an+bk+c}$ in f according to the shadowing rule described above. Since $f_{n,k}/f_{n-1,k} = \bar{f}_{n,k}/\bar{f}_{n-1,k}$ and $f_{n,k}/f_{n,k-1} = \bar{f}_{n,k}/\bar{f}_{n,k-1}$, the sequences f and $\bar{f}$ are essentially equivalent. This means, they only differ in initial values, in their domain of definition, and when they vanish. Thus, it follows that, if f and g form a qWZ-pair, then so do $\bar{f}$ and $\bar{g}$.

Evidently, one is free to shadow only some of the factors of $f_{n,k}$ and $g_{n,k}$ and fixing the others, this way getting different shadow pairs. Following Wilf [1993], a choice that seems to give fruitful results in general, is to shadow only those factors $(A;q^d)_{an+bk+c}$ for which $a + b \neq 0$. Hence, we will apply this kind of "default shadowing" in the algorithm. In this case, to avoid trivial qWZ-pairs like $(0,0)$, etc., we have to cancel all q-shifted factorial expressions in $f_{n,k}$ and $g_{n,k}$ being free of n and k, which again gives us a qWZ-pair.

The final step in dualization is to pass from the shadow pair $(\bar{f}, \bar{g})$ to the dual pair (f', g') by a flip of variables and sequences, transforming the domain of n back to the nonnegative integers. Finally, the dual pair is defined as

$$(f'_{n,k}, g'_{n,k}) := \left(\bar{g}_{-k,-n-1}, \bar{f}_{-k-1,-n}\right),$$

which does not influence the fact that the sequences form a qWZ-pair, but which does alter the certificate via the same change of variables.

Note that in general dualization does not commute with specialization, i.e., the dual identity of some special case of an identity is not the same as the specialization of the dual identity. However, dualization is an involution up to constant factors.

The program computes the dual qWZ-pair, if the global variable `Dual` is set to `True`, and f and g actually form a qWZ-pair. The result is assigned to the variable `DualPair`. Default value for `Dual` is `False`.

Up to now we have not made a systematic investigation of producing new q-identities by dualization, as done for $q = 1$ by Gessel [1995]. We only present a few examples explaining the use of the package; see also Section 5.3.

For the q-Chu-Vandermonde identity above one gets the following:

```
In[12]:= Dual = True;
         qZeil[qBinomial[n,k,q] qBinomial[b,k,q] q^(k^2) /
               qBinomial[b+n,n,q], {k,0,n}, n, 1, {b}]
Out[13]= SUM[n] == 1

In[14]:= DualPair
                                           2                     2
                  k + n  k/2 + b k + k /2 - n/2 - b n - n /2
Out[14]= {((-1)      q

     qBinomial[n, k, q] qfac[q, q, -1 - b + k]

     qfac[q, q, -1 - b - n] qfac[q, q, n]) / qfac[q, q, k],

                                      2                  2
        k + n  b + k/2 + b k + k /2 + n/2 - b n - n /2
  ((-1)      q

     qBinomial[-1 + n, k, q] qfac[q, q, -b + k]

     qfac[q, q, -1 - b - n] qfac[q, q, -1 + n]) / qfac[q, q, k]}
```

Hence, after replacing b by $-b-1$ the dual identity becomes

$$\sum_{k=0}^{n} (-1)^{n+k} q^{(n-k)(2b-k-n+1)/2} \begin{bmatrix} n \\ k \end{bmatrix}_q \begin{bmatrix} b+k \\ k \end{bmatrix}_q = \begin{bmatrix} b \\ n \end{bmatrix}_q,$$

which is the same as the original identity modulo a renaming of the parameters. An identity satisfying this property is called *self-dual*.

As mentioned above, for the special case $b = n$ we do not obtain just the dual identity with b replaced by n, but:

$$\sum_{k=0}^{n} \begin{bmatrix} 2k \\ k \end{bmatrix}_q \begin{bmatrix} n \\ k \end{bmatrix}_q^2 \frac{q^{n-k} + q^n - 2q^k}{1+q^k} = 0$$

presented by Wilf and Zeilberger [1992b].

Next, let us consider the q-Saalschütz identity in the form

$$\sum_{k=0}^{n} \begin{bmatrix} r-s+m \\ k \end{bmatrix}_q \begin{bmatrix} s-r+n \\ n-k \end{bmatrix}_q \begin{bmatrix} s+k \\ m+n \end{bmatrix}_q q^{(n-k)(r-s+m-k)} = \begin{bmatrix} r \\ n \end{bmatrix}_q \begin{bmatrix} s \\ m \end{bmatrix}_q.$$

The program computes the following dual identity (cf. Riese [1995]):

$$\sum_{k=0}^{n} \begin{bmatrix} m+k \\ k \end{bmatrix}_q \begin{bmatrix} s \\ r-k \end{bmatrix}_q \begin{bmatrix} m-s \\ n-k \end{bmatrix}_q q^{(n-k)(r-k)} = \begin{bmatrix} m+r-s \\ n \end{bmatrix}_q \begin{bmatrix} n+s \\ r \end{bmatrix}_q .$$

Renaming the parameters we find the q-Saalschütz identity also to be self-dual.

For the special case $m = n$ and $r = s$, the process of dualization leads to the following result (cf. Riese [1995]):

$$\sum_{k=0}^{n} \frac{1 + q^{-k} - 2q^{n-2k} - 2q^{k-s} + q^{n-s} + q^{n-s-k}}{1+q^k} \cdot \begin{bmatrix} 2k \\ k \end{bmatrix}_q \begin{bmatrix} n \\ k \end{bmatrix}_q^2 \frac{(q;q)_{n+s-2k-1}}{(q;q)_{s-k}^2} = 0,$$

where $s \geq n+1$.

Finally, for the q-Dixon identity

$$\sum_k (-1)^k \begin{bmatrix} n+b \\ n+k \end{bmatrix}_q \begin{bmatrix} n+c \\ c+k \end{bmatrix}_q \begin{bmatrix} b+c \\ b+k \end{bmatrix}_q q^{k(3k-1)/2} = \begin{bmatrix} n+b+c \\ n, b, c \end{bmatrix}_q$$

we get, by "creative symmetrizing" described in Section 5.2, i.e., taking as input,

```
In[15]:= qZeil[(1+q^k)/2 (-1)^k qBinomial[n+b,n+k,q] *
             qBinomial[n+c,c+k,q] qBinomial[b+c,b+k,q] *
             q^(k(3k-1)/2) qfac[q,q,n] qfac[q,q,b] qfac[q,q,c] /
             qfac[q,q,n+b+c],
             {k, -Infinity, Infinity}, n, 1, {b, c}]
Out[15]= SUM[n] == SUM[-1 + n]
```

and then calling `DualPair`, the dual identity

$$\sum_{k=0}^{n} (-1)^k q^{\binom{k+1}{2} - n(1+k+n)} \begin{bmatrix} b+c+k \\ b-1 \end{bmatrix}_q \begin{bmatrix} b+n \\ b+k \end{bmatrix}_q \begin{bmatrix} n+k \\ k \end{bmatrix}_q = \begin{bmatrix} b+c \\ c+n+1 \end{bmatrix}_q \begin{bmatrix} c \\ n \end{bmatrix}_q ,$$

which is nothing but a specialization of the q-Saalschütz identity.

4.2.10 *Some Remarks on Run-Time and History.* Concerning the run-time, the main computational part of Gosper's algorithm and its q-analogue consists in solving a system of homogeneous linear equations with polynomial coefficients. It turned out that the Mathematica functions `NullSpace` and `LinearSolve` are absolutely impracticable even for rather simple applications. To overcome this problem, E. Aichinger wrote a Mathematica function `ENullSpace` based on Gaussian elimination, which does the job excellently for most of the examples. The interface `LinSolve` was written by M. Schorn; see Paule and Schorn [1993]. In the first prototype versions of `qZeil` up to 95 percent of the run-time were spent for solving the system of equations. Meanwhile this amount has decreased to about 30–40 percent in average, mainly due to a preprocessing of the system in which all constant factors with respect to the summation variable are extracted. Furthermore, a lot of considerations had to be put into finding a powerful *and* efficient simplification procedure. As a compromise, the strategy now is based on collecting several rewrite rules into blocks which are applied one after the other.

4.3 A Comparison with Koornwinder's Package. As already mentioned, Koornwinder [1993] implemented Zeilberger's algorithm and its q-analogue in Maple. Furthermore, he gives a rigorous description of the ordinary algorithm and some remarks how to carry it over to the q-case. His program implements q-hypergeometric telescoping for

$$\sum_{k=0}^{n} \frac{(\alpha_1;q)_k\,(\alpha_2;q)_k\cdots(\alpha_r;q)_k}{(q;q)_k\,(\beta_1;q)_k\cdots(\beta_s;q)_k}\,\left((-1)^k q^{\binom{k}{2}}\right)^{1+s-r}\zeta^k$$

and the q-Zeilberger algorithm for

$$\sum_{k=0}^{n} \frac{(q^{-n};q)_k\,(q^{i_2 n}\alpha_2;q)_k\cdots(q^{i_r n}\alpha_r;q)_k}{(q;q)_k\,(q^{j_1 n}\beta_1;q)_k\cdots(q^{j_s n}\beta_s;q)_k}\,\left((-1)^k q^{\binom{k}{2}}\right)^{1+s-r}(q^{n\nu}\zeta)^k\,,$$

where $\alpha_1,\ldots,\alpha_r,\beta_1,\ldots,\beta_s$ and ζ are rational functions over the rational number field in a fixed number of indeterminates including q (but not q^k), and $i_2,\ldots,i_r$, $j_1,\ldots,j_s,\nu$ are integers such that

$\log_q(\beta_t)$ noninteger if $j_t=-1,-2,\ldots$; $\log_q(\beta_t)\neq 0,-1,-2,\ldots$ if $j_t=0$;
$\log_q(\alpha_t)$ noninteger if $i_t=0$; $\zeta\neq 0$.

Since the input specification described above is confined to basic hypergeometric series in base q, which is too restrictive in practice, with our implementation we tried to overcome some of these shortcomings which are:

- The summation range cannot be changed to an interval different from $[0,n]$ like $[-n,n]$ or $[0,2n]$, etc. So, for instance, it is not possible to find a recurrence for Jackson's terminating q-analogue of Dixon's sum

$${}_3\phi_2\left[\begin{matrix} q^{-2n},b,c \\ q^{1-2n}/b,q^{1-2n}/c\end{matrix};q,\frac{q^{2-n}}{bc}\right]=\frac{(b,c;q)_n\,(q,bc;q)_{2n}}{(q,bc;q)_n\,(b,c;q)_{2n}}.$$

- Since $(q^{-n};q)_k$ and $(q;q)_k^{-1}$ have to be factors of the summand, the program always assumes finite support. Therefore no inhomogeneous recurrences can be dealt with.
- Concerning the bases of the q-shifted factorials, no powers of q are accepted. In addition, it is impossible to split q-shifted factorials of the form $(\alpha;q^m)_k$ for $m>1$ into $(\alpha^{(1)},\ldots,\alpha^{(m)};q)_k$, because in general the $\alpha^{(i)}$ are the *complex* roots of α. Furthermore, no rational powers of indeterminates are allowed.
- The index in q-shifted factorial expressions is restricted to be k, and the α's and β's must be free of k. One often has to apply expensive transformations to achieve this form, or even worse, $(\alpha;q)_{2k}$, for instance, cannot be split into $(\alpha;q)_k\,(\alpha q^k;q)_k$, etc.
- No polynomial part can be specified. Hence, "creative symmetrizing" from Section 5.2 in general cannot be applied.

Finally, let us compare the run-time for some typical examples all taken from the Gasper and Rahman [1990] book, which fit also to Koornwinder's input specification:

1. the q-Chu-Vandermonde sum, eq. (II.6) in Appendix II,
2. the q-Dixon sum, eq. (II.13) in Appendix II,
3. the sum of a very-well-poised ${}_6\phi_5$ series, eq. (II.21) in Appendix II,
4. Jackson's q-analogue of Dougall's ${}_7F_6$ sum, eq. (II.22) in Appendix II,
5. Watson's transformation formula, eq. (III.17) in Appendix III,
6. Bailey's ${}_{10}\phi_9$ transformation formula, eq. (III.28) in Appendix III,
7. a transformation formula of a nearly-poised ${}_5\phi_4$ series, eq. (III.25) in Appendix III.

The following table of run-times (in seconds) refers to tests on a Pentium 100 with 16 MB memory using Mathematica 2.2 for Windows and Maple V.2 for Windows, respectively. To have a fair competition, we eliminated a bug in Koornwinder's degree bound computation. Also minor adjustments for the input terms were made due to syntax restrictions of the Maple package. For the entry "Turbo-qZeil" recall the specification of the Turbo-mode from Section 4.2.5.

#	Series	Order	qZeil	Turbo-qZeil	Koornwinder
1	${}_2\phi_1$	1	1,59	0,88	0,55
2	${}_4\phi_3$	1	4,23	1,76	1,23
3	${}_6\phi_5$	1	5,33	2,25	1,80
4	${}_8\phi_7$	1	14,17	7,80	12,03
5	${}_4\phi_3$	2	13,07	9,39	4,59
5	${}_8\phi_7$	2	17,74	9,72	8,95
6	${}_{10}\phi_9$	2	235	224	out of memory[1]
6	${}_{10}\phi_9$	2	214	194	out of memory[2]
7	${}_5\phi_4$	3	47	38	113
7	${}_{12}\phi_{11}$	3	out of memory	out of memory	out of memory

5 Applications

In this section we present nontrivial examples illustrating various features of our package as well as the wide range of its applicability. For each subsection we assume that the `qZeil.m` package has been loaded as explained in Section 4.2.1.

5.1 An Identity Conjectured by D. Stanton. Dennis Stanton [1995] conjectured the following identity,

$$\sum_k (-1)^k q^{4k^2} \begin{bmatrix} 2n \\ n-4k \end{bmatrix}_q = \sum_k q^{2k^2} \begin{bmatrix} n \\ 2k \end{bmatrix}_{q^2} (-q;q^2)_{n-2k}(-1;q^4)_k. \tag{14}$$

As demonstrated below, for the package `qZeil` the proof of (14) is no problem at all. This computer proof was the first proof that was given for Stanton's conjecture.

At this occasion we want to point out that we tried to make the package as user-friendly as possible. This means, the translation from q-series notation to Mathematica input for `qZeil` should cause only minimal effort from side of the user.

Note also that for summations over finite support, as in (14), it is convenient to take as summation range {`k,-Infinity,Infinity`} which saves computing time.

For the left hand side of (14) one has

[1]after 5600 seconds

[2]after 6200 seconds

```
In[16]:=  qZeil[(-1)^k q^(4 k^2) qBinomial[2n,n-4k,q],
                 {k,-Infinity,Infinity}, n, 3]
Out[16]=
             -3 + 2 n        -5 + 2 n         -4 + 2 n
  SUM[n] == q         (1 - q         ) (1 - q         ) SUM[-3 + n] +

       -7 + 2 n   5    2 n    1 + 2 n    2 + 2 n
      q         (q  - q    - q        - q        ) SUM[-2 + n] +

        3    2 n    1 + 2 n    2 + 2 n
      (q  + q    + q        + q        ) SUM[-1 + n]
      ---------------------------------------------
                            3
                           q
```

and the proof certificate $cert(n,k)$ comes in the nicely factored form:

```
In[17]:= Cert
           -12 k + n   1 + 4 k    n     4 k    n     2 + 4 k    n
Out[17]= (q          (q        - q ) (-q    + q ) (-q        + q )

          3 + 4 k    n
       (-q        + q )) /

           n         n         n         n         2 n    3    2 n
     ((q - q ) (-1 + q ) (1 + q ) (q + q ) (q - q   ) (q  - q   ))
```

For the right hand side with input

```
In[18]:= qZeil[q^(2 k^2) qBinomial[n,2k,q^2] qfac[-q,q^2,n-2k] *
                qfac[-1,q^4,k], {k,-Infinity,Infinity}, n, 3];
```

we get the same recurrence with the proof certificate, also nicely factored:

```
In[19]:= Cert
Out[19]=
   6 + 4 k + 2 n              4 k     2 k    n     2 k    n
 (q              (1 + q) (1 + q   ) (-q    + q ) (q    + q )

          1 + 2 k    n    1 + 2 k    n
       (-q        + q ) (q        + q )) /

           n         n         n         n    1 + 4 k    2 n
     ((q - q ) (-1 + q ) (1 + q ) (q + q ) (q        + q   )

         3 + 4 k    2 n    5 + 4 k    2 n
       (q        + q   ) (q        + q   ))
```

Finally, the proof of (14) is completed by checking the identity at the initial values $n = 0$, $n = 1$, and $n = 2$.

5.2 Creative Symmetrizing. In the hypergeometric case it is well-known that Zeilberger's algorithm not always delivers the minimal recurrence; see Andrews [1995], Paule and Schorn [1993], or Petkovšek, Wilf and Zeilberger [1996]. In the q-case the situation is similar, but one can observe that more standard identities like q-Dixon (cf. Gasper and Rahman [1990, (II.15)]) are infected by the nonminimality-virus. However, creative symmetrizing removes this infection in most of these cases. Creative symmetrizing also explains the remarkable observation that in almost all of these instances the $q = 1$ specialization behaves nice with respect to minimality.

Consider the following identity due to L.J. Rogers:

$$\sum_k \frac{(-1)^k q^{k(3k-1)/2}}{(q;q)_{n+k}(q;q)_{n-k}} = \frac{1}{(q;q)_n}. \tag{15}$$

This identity not only is a finite version of Euler's pentagonal number theorem (cf. Andrews [1976]), but also plays a fundamental role in connection with the Rogers-Ramanujan identities and, more general, in the context of Bailey chains; an excellent account is given in Andrews [1986].

Trying to prove (15) with `qZeil` results in a surprise: the program succeeds only with a recurrence of order 3 instead of expected order 1:

```
In[20]:= f[n_,k_] := (-1)^k q^(k(3k-1)/2) /
                     (qfac[q,q,n+k] qfac[q,q,n-k])

In[21]:= qZeil[f[n,k], {k,-n,n}, n, 1]
Out[21]= No solution: Increase order by 1

In[22]:= qZeil[f[n,k], {k,-n,n}, n, 2]
Out[22]= No solution: Increase order by 1

In[23]:= qZeil[f[n,k], {k,-n,n}, n, 3]
                          3
                         q  SUM[-3 + n]
Out[23]= SUM[n] == ------------------------- -
                         2 n       -1 + 2 n
                   (1 - q   ) (1 - q        )

        2    3    4    2 n
      (q  + q  + q  + q   ) SUM[-2 + n]
      ------------------------------- -
                2 n       -1 + 2 n
        q (1 - q   ) (1 - q        )

        2    3    4    3 n    2 + 2 n    1 + 3 n
      (-q  - q  - q  + q    + q        + q       ) SUM[-1 + n]
      -----------------------------------------------------
                   2      2 n       -1 + 2 n
                  q  (1 - q   ) (1 - q        )
```

Hence in situations like that we are faced with three problems:
(i) How to find the minimal recurrence?
(ii) The computing time and size of the proof certificate might increase drastically.
(iii) One cannot apply the qWZ-pair machinery in order to get dual and companion identities "for free".

A solution to problem (i) is provided by the q-analogue of Petkovšek's algorithm `Hyper`; see Abramov, Paule and Petkovšek [1995]. Nevertheless, *creative symmetrizing* in many instances solves all of these problems.

Creative symmetrizing has been introduced by Paule [1994] for certifying finite versions of Rogers-Ramanujan type identities, for which the same problem as for (15) was observed. As found by Petkovšek, the method also can be applied in analogous $q = 1$ situations; see the book by Petkovšek, Wilf, and Zeilberger [1996], in which the term "creative antisymmetrizing" was introduced for a special case of the method (i.e., if $R(n, k) = -1$ in the lemma below).

Creative symmetrizing is based on the following essentially trivial lemma.

Lemma 4 ("Creative Symmetrizing") *For fixed $n \in \mathbf{N}$ let I_n be an integer interval with bounds α and β, both being free of k. If for all $k \in I_n$,*

$$f_{n,\alpha+\beta-k} = R(n, k) \cdot f_{n,k},$$

then

$$\sum_{k=\alpha}^{\beta} f_{n,k} = \frac{1}{2} \sum_{k=\alpha}^{\beta} (1 + R(n, k)) \cdot f_{n,k}.$$

Proof. The proof is an immediate consequence of symmetrizing around $(\alpha + \beta)/2$, i.e.,

$$2 \sum_{k=\alpha}^{\beta} f_{n,k} = \sum_{k=\alpha}^{\beta} (f_{n,k} + f_{n,\alpha+\beta-k}).$$

□

In the applications below, $R(n, k)$ will be either a polynomial or a rational function in q^n and q^k. The point of creative symmetrizing lies in the observation that applying `qZeil` to the symmetrized summand $(1 + R(n, k)) \cdot f_{n,k}$ instead of $f_{n,k}$, for certain types of input sequences increases the chances to get a minimal output recurrence.

As mentioned above there are cases, like q-Dixon, with minimal order of the Zeilberger recurrence in the ordinary case, but with nonminimal order for their q-analogue. If in these instances creative symmetrizing succeeds in minimal order reduction, the explanation lies in the fact that for $q = 1$ the rational function $R(n, k)$ reduces to a constant. See, for instance, the q-Dixon input `In[15]` in Section 4.2.9.

With respect to (15) creative symmetrizing works as follows: Denote the summand by $f_{n,k}$, then evidently $f_{n,-k} = q^k \cdot f_{n,k}$, i.e., $R(n, k) = q^k$ and $\alpha = -n$, $\beta = n$. Applying `qZeil` to the symmetrized summand $(1 + q^k) \cdot f_{n,k}$ now indeed delivers the expected minimal recurrence:

```
In[24]:= qZeil[(1+q^k) f[n,k], {k,-n,n}, n, 1]
                    SUM[-1 + n]
Out[24]= SUM[n] == -----------
                          n
                      1 - q

In[25]:= Cert
          k + n     k    n
         q       (-q  + q )
Out[25]= ------------------
              k         n
         (1 + q ) (-1 + q )
```

(Note that the proof certificate is particularly nice.)

The next example — one Bailey chain step from (15) and its companion (see Andrews [1986]) — concerns finite versions of the Rogers-Ramanujan identities. For instance, a finite version of the Rogers-Ramanujan identity related to 2 and 3 (mod 5) is

$$\sum_k \frac{q^{k^2+k}}{(q;q)_k (q;q)_{n-k}} = \sum_k \frac{(-1)^k q^{k(5k+3)/2}}{(q;q)_{n+k+1}(q;q)_{n-k}}, \tag{16}$$

which was stated first in this form by Andrews [1974], see also Bressoud [1981].

With qZeil, one proves that the left hand side satisfies a recurrence of order 2, namely:

```
In[26]:= qZeil[q^(k^2+k) / (qfac[q,q,k] qfac[q,q,n-k]),
                {k,0,n}, n, 2]

Out[26]=                                           n    2 n
                q SUM[-2 + n]     (1 + q - q  + q   ) SUM[-1 + n]
     SUM[n] == -(-------------) + ------------------------------
                          n                             n
                     1 - q                         1 - q

In[27]:= Cert
          -k + 2 n    k     n
         q          (q  - q )
Out[27]= -------------------
                         n
                 -1 + q
```

Now the surprise comes with respect to the right hand side. Namely, the program only finds a recurrence of order 5 (!) with a sufficiently lengthy certificate. Creative symmetrizing also resolves this problem: Denote by $f_{n,k}$ the corresponding summand, and observe that with $\alpha = -n-1$ and $\beta = n$ we have $f_{n,-1-k} = (-q^{2k+1}) \cdot f_{n,k}$, i.e., $R(n,k) = -q^{2k+1}$. Applying qZeil to the symmetrized summand $(1 - q^{2k+1}) \cdot f_{n,k}$ produces the expected minimal recurrence of the same form as the one for the left hand side:

```
In[28]:= f[n_,k_] := (-1)^k q^(k(5k+3)/2) /
                      (qfac[q,q,n+k+1] qfac[q,q,n-k])

In[29]:= qZeil[(1-q^(2k+1)) f[n,k], {k,0,n}, n, 2]
Out[29]=                                           n    2 n
                q SUM[-2 + n]     (1 + q - q  + q   ) SUM[-1 + n]
     SUM[n] == -(-------------) + ------------------------------
                          n                             n
                     1 - q                         1 - q

In[30]:= Cert
          2 + 2 k + 2 n        1 + k    k    n
         q              (-1 + q     ) (q  - q )
Out[30]= --------------------------------------
                        1 + 2 k        n
               (-1 + q        ) (1 - q )
```

Checking equality at the initial values at $n = 0$ and $n = 1$ completes the proof of (16). (Note that the proof certificates for both sides again are sufficiently nice, so that the verification of (12) can be easily done by a human.) The `qZeil` proof of the companion identity to (16) can be found in Paule [1994].

We want to remark that identity (16) and its companion are the specializations $a = 1$ and $a = q$ of

$$\sum_k \frac{a^k q^{k^2}}{(q;q)_k (q;q)_{n-k}} = \sum_k \frac{(-1)^k a^{2k} q^{k(5k-1)/2} (aq;q)_{k-1}}{(q;q)_{n-k} (aq;q)_{n+k} (q;q)_k} \cdot (1 - aq^{2k}), \qquad (17)$$

a terminating version of Watson's q-analogue of Whipple's theorem; see Gasper and Rahman [1990, (III.18)].

Indeed, for $a = q$ we find that the left hand side of (17) equals the *unilateral* sum

$$\sum_{k \geq 0} (1 - q^{2k+1}) f_{n,k}.$$

Hence, in the classical unilateral version the symmetrizing factor, which we have introduced into the bilateral summation, is already there! This can be taken as a classical explanation, why creative symmetrizing can be successfully applied in so many cases.

Since Watson's identity is a transformation formula for a terminating very-well poised ${}_8\phi_7$ series, this gives rise to the conjecture that creative symmetrizing could be applied successfully in very-well poised context. This is strongly confirmed, also for well- or nearly-poised series, by inspection of numerous examples from various sources; see the file `qZeilExamples.txt`.

We conclude this section by having a look at a summation that arose in work of Andrews and Jackson [1990], namely

$${}_3\phi_2 \left[\begin{matrix} q^{-2n}, b, c \\ q^{-2n}/b, q^{-2n}/c, \end{matrix} ; q, \frac{q^{-n}}{bc} \right] = \frac{(q^{-2n};q)_n (q^{-2n}/bc;q)_n}{(q^{-2n}/b;q)_n (q^{-2n}/c;q)_n}. \qquad (18)$$

Independently this *almost poised* series identity was proved by Bressoud [1987], who introduced this new notion together with numerous interesting examples.

Applying `qZeil` directly to the summand $f_{n,k}$ of the series in (18), the output recurrence order is 3.

With respect to Lemma 4 we have $\alpha = 0$, $\beta = 2n$, and $f_{n,2n-k} = R(n,k) \cdot f_{n,k}$ with

$$R(n,k) = q^{n-k} \frac{(1 - bq^k)(1 - cq^k)}{(1 - bq^{2n-k})(1 - cq^{2n-k})}.$$

Applying `qZeil` to the symmetrized summand $(1 + R(n,k)) \cdot f_{n,k}$ produces the expected minimal recurrence:

```
In[31]:= f[n_,k_] := qfac[q^(-2n),q,k] qfac[b,q,k] qfac[c,q,k] /
         (qfac[q^(-2n)/b,q,k] qfac[q^(-2n)/c,q,k] qfac[q,q,k]) *
         (q^(-n)/(b c))^k

In[32]:= R[n_,k_] := q^(n-k) (1-b q^k)(1-c q^k) /
                     ((1-b q^(2n-k))(1-c q^(2n-k)))
```

```
In[33]:= Factor[1+R[n,k]]
             k    n    k         3 n      k + n      k + n
Out[33]= ((q  + q ) (q  + b c q    - b q      - c q      +

         2 k + n        k + 2 n         k      2 n     k      2 n
    b c q        - b c q       )) / ((-q  + b q   ) (-q  + c q   ))

In[34]:= qZeil[Factor[1+R[n,k]] f[n,k], {k,0,2n}, n, 1]
                                  n          n          n           2 n
Out[34]= SUM[n] == ((1 + q ) (1 - b q ) (1 - c q ) (1 - b c q   )

              -1 + 2 n            -1 + 2 n
        (1 - q        ) (1 - b c q        ) SUM[-1 + n]) /

                n          2 n          2 n          -1 + 2 n
      ((1 - b c q ) (1 - b q   ) (1 - c q   ) (1 - b q        )

                -1 + 2 n
        (1 - c q        ))
```

(In this case the certificate is about the size of one page. In such instances the verification of (12) is left to the computer.)

5.3 An Identity of L.J. Rogers as a Dual Identity. As already mentioned, creative symmetrizing enables to apply the qWZ-pair machinery. As an example let us consider the finite form of Euler' pentagonal number theorem (15) due to L.J. Rogers.

We start out with the symmetrized version as input:

```
In[35]:= Dual = True;
         qZeil[(1+q^k) (-1)^k q^(k(3k-1)/2) qfac[q,q,n] /
               (qfac[q,q,n+k] qfac[q,q,n-k]), {k,-n,n}, n, 1]
```

and obtain as expected:

```
Out[36]= SUM[n] == 2
```

The dual pair reads as:

```
In[37]:= DualPair
                                  2
           -k + n - k n + n
          q                  qfac[q, q, k + n]
Out[37]= {-----------------------------------,
            qfac[q, q, k] qfac[q, q, -k + n]

                       2
        -n - k n + n         n
       q              (1 + q ) qfac[q, q, k + n]
     -(---------------------------------------)}
          qfac[q, q, k] qfac[q, q, -1 - k + n]
```

Summing the first entry we obtain:

```
In[38]:= qZeil[DualPair[[1]], {k,0,n}, n, 1]
Out[38]= SUM[n] == 1
```

which in usual notation is

$$\sum_{k=0}^{n} q^{(n-k)(n+1)} \frac{(q;q)_{n+k}}{(q;q)_k (q;q)_{n-k}} = 1. \tag{19}$$

Dualization of (19) brings us back to the symmetrized version of (15):

```
In[39]:= DualPair
                                2
               k   -k/2 + (3 k )/2       k
           (-1)   q                 (1 + q ) qfac[q, q, n]
Out[39]= {--------------------------------------------,
               qfac[q, q, -k + n] qfac[q, q, k + n]

                              2
          k   (3 k)/2 + (3 k )/2 + n
      (-1)   q                          qfac[q, q, -1 + n]
      ------------------------------------------------}
          qfac[q, q, -1 - k + n] qfac[q, q, k + n]
```

Summarizing, because identity (19) is the limiting case $c \to \infty$ of eq. (II.7) in Gasper and Rahman [1990] with $a = q^{n+1}$, we have obtained Rogers' finite version (15) of Euler's pentagonal number theorem as the dual of a limiting case of Chu-Vandermonde (reversed).

We want to conclude this section by computing the WZ-companion identity with respect to the symmetrized version of (15):

```
In[40]:= Companion = True;

In[41]:= qZeil[(1+q^k) (-1)^k q^(k(3k-1)/2) qfac[q,q,n] /
               (qfac[q,q,n+k] qfac[q,q,n-k]), {k,-n,n}, n, 1]
Out[41]= SUM[n] == 2

In[42]:= CompId
                                            2
                  k   1 + (3 k)/2 + (3 k )/2 + n
              (-1)   q                                qfac[q, q, n]
Out[42]= Sum[-----------------------------------------------,
                 qfac[q, q, -k + n] qfac[q, q, 1 + k + n]

      {n, 0, Infinity}] == -2 (k >= 0) +

                                    2
               jj   -jj/2 + (3 jj )/2       jj
           (-1)    q                   (1 + q   )
      Sum[----------------------------------, {jj, -Infinity, k}]
                 qfac[q, q, Infinity]
```

5.4 An Identity Discovered by R.J. McIntosh. Comparing asymptotic expansions associated with different q-series, R.J. McIntosh [1995] conjectured that

$$\sum_{k=0}^{\infty} \frac{q^{(n+2k)(n+2k+1)/2}}{(q^2;q^2)_k} = (-q;q)_\infty \sum_{k=0}^{\infty} \frac{(-1)^k q^{k(k+1)/2-nk}}{(q^2;q^2)_k}, \tag{20}$$

where n is any integer. This conjecture was communicated to G.E. Andrews who later provided a proof. Using the package `qZeil` the verification of (20) is mere routine.

First let us assume n to be a *nonnegative* integer. Define for nonnegative integer m:

$$L_n(m) := \sum_{k=0}^{m} \frac{q^{(n+2k)(n+2k+1)/2}}{(q^2;q^2)_k} \quad \text{and} \quad R_n(m) := \sum_{k=0}^{m} \frac{(-1)^k q^{k(k+1)/2-nk}}{(q^2;q^2)_k}.$$

Now make use of the "general bounds" feature of `qZeil`, explained in Section 4.1,

```
In[43]:= qZeil[q^((n+2k)(n+2k+1)/2) / qfac[q^2,q^2,k],
              {k,0,m}, n, 2]
```

and obtain

```
                                  2                   2
                        m + 2 m  + n/2 + 2 m n + n /2
                       q
Out[43]= SUM[n] == ----------------------------- + SUM[-2 + n] -
                                      2   2
                             qfac[q , q , m]

      1 - n
     q      SUM[-1 + n]
```

with

```
In[44]:= Cert
Out[44]= 1
```

This means, $cert(n,k) = 1$ is the proof certificate of the inhomogeneous recurrence ($n \geq 2$)

$$L_n(m) = \frac{q^{\binom{n+1}{2}+m+2mn+2m^2}}{(q^2;q^2)_m} - q^{-n+1} L_{n-1}(m) + L_{n-2}(m). \tag{21}$$

Analogously for R_n,

```
In[45]:= qZeil[(-1)^k q^(k(k+1)/2 - n k) / qfac[q^2,q^2,k],
              {k,0,m}, n, 2]
                                              2
                          m  1 + (3 m)/2 + m /2 - n - m n
                      (-1)  q
Out[45]= SUM[n] == ---------------------------------- +
                                        2   2
                               qfac[q , q , m]

                          1 - n
     SUM[-2 + n] - q      SUM[-1 + n]
```

```
In[46]:= Cert
            1 + k - n
Out[46]= q
```

which means, $cert(n,k) = q^{1+k-n}$ is the proof certificate of the inhomogeneous recurrence ($n \geq 2$)

$$R_n(m) = \frac{q^{1-n(m+1)+(m+3)m/2}}{(q^2;q^2)_m} - q^{-n+1} R_{n-1}(m) + R_{n-2}(m). \qquad (22)$$

Since for fixed n and $m \to \infty$ the inhomogeneous parts of (21) and (22) vanish, the sequences $L_n(\infty)$, $R_n(\infty)$, and also $(-q;q)_\infty \cdot R_n(\infty)$ satisfy the same *homogeneous* recurrence of order 2. Therefore, all what is left to prove (20) for $n \geq 0$ is to check (20) at the initial values $n = 0$ and $n = 1$.

For *negative* integer n the proof is entirely analogous; one simply replaces n by $-n$ in the defining sums for $L_n(m)$ and $R_n(m)$, and the initial value check at $n = 0$ and $n = -1$ completes the proof.

The initial value check is an easy consequence of a classical identity which is due to L.J. Rogers:

$$\sum_{k=0}^{\infty} \frac{q^{k(2k+1)} t^{2k}}{(q^2;q^2)_k} = (tq;q)_\infty \sum_{k=0}^{\infty} \frac{q^{\binom{k+1}{2}} t^k}{(q;q)_k (tq;q)_k}. \qquad (23)$$

This is eq. (3.22) in Andrews and Askey [1977], where one finds an elegant proof together with some background information.

Eq. (23) with $t = -1$ gives $L_0(\infty) = R_0(\infty)$. Eq. (23) with $t = -1/q$ results in

$$L_{-1}(\infty) = R_1(\infty) + R_0(\infty). \qquad (24)$$

Hence, by equality at $n = 0$ we have

$$R_1(\infty) = L_{-1}(\infty) - L_0(\infty) = \sum_{k \geq 0} \frac{q^{k(2k-1)}}{(q^2;q^2)_k} (1 - q^{2k}) = L_1(\infty).$$

Finally,

$$R_1(\infty) - R_{-1}(\infty) = (-q;q)_\infty \sum_{k \geq 0} (-1)^k \frac{q^{k(k-1)/2}}{(q^2;q^2)_k} (1 - q^{2k}) = -R_0(\infty),$$

which implies $R_{-1}(\infty) = R_1(\infty) + R_0(\infty) = L_{-1}(\infty)$, where the last equality follows from (24).

Acknowledgement. We wish to thank Doron Zeilberger for inspiring comments.

References

Abramov, S.A., Paule, P. and Petkovšek, M. [1995] *q-Hypergeometric Solutions of q-Difference Equations*, preprint.

Andrews, G.E. [1974] *Problem 74-12*, SIAM Rev., **16**, 390.

Andrews, G.E. [1976] *The Theory of Partitions*, Encyclopedia of Mathematics and its Applications, **2** (G.-C. Rota, ed.), Addison-Wesley, Reading (Reissued: Cambridge University Press, London and New York, 1985).

Andrews, G.E. [1986] *q-Series: Their Development and Application in Analysis, Number Theory, Combinatorics, Physics, and Computer Algebra*, CBMS Regional Conference Series, No. 66, Amer. Math. Soc., Providence.

Andrews, G.E. [1995] *Pfaff's Method (III): Comparison with the WZ Method*, preprint, to appear in: Elec. J. Combinatorics.

Andrews, G.E. and Askey, R. [1977] *Enumeration of partitions: The role of Eulerian series and q-orthogonal polynomials*, Higher Combinatorics (M. Aigner, ed.), Reidel Publ. Co., Dordrecht, Holland, pp. 3–26.

Andrews, G.E. and Jackson, D.M. [1990] *An algebraically derived q-analog of a character sum associated with a class of semiregular permutations*, Pac. J. Math., **14**, 207–218.

Bressoud, D.M. [1981] *Solution of Problem 74-12*, SIAM Rev., **23**, 101–104.

Bressoud, D.M. [1987] *Almost Poised Basic Hypergeometric Series*, Proc. Indian Acad. Sci. (Math. Sci.), **96**, 61–66.

Gasper, G. and Rahman, M. [1990] *Basic Hypergeometric Series*, Encyclopedia of Mathematics and its Applications, **35** (G.-C. Rota, ed.), Cambridge University Press, London and New York.

Gessel, I. [1995] *Finding identities with the WZ method*, preprint, to appear in: J. Symbolic Computation.

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

Karr, M. [1981] *Summation in finite terms*, Journal of the ACM, **28**, 305–350.

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

McIntosh, R.J. [1995] *Some Asymptotic Formulae for q-Hypergeometric Series*, preprint, to appear in: J. London Math. Soc.

Paule, P. [1994] *Short and Easy Computer Proofs of the Rogers-Ramanujan Identities and of Identities of Similar Type*, Elec. J. Combinatorics, **1**, R10.

Paule, P. [1995] *Greatest Factorial Factorization and Symbolic Summation*, preprint, to appear in: J. Symbolic Computation.

Paule, P. and Schorn, M. [1993] *A Mathematica version of Zeilberger's algorithm for proving binomial coefficient identities*, RISC-Linz Report Series, **93-36**, J. Kepler University, Linz, to appear in: J. Symbolic Computation.

Paule, P. and Strehl, V. [1995] *Symbolic summation — some recent developments*, Computeralgebra in science and engineering — algorithms, systems, applications (J. Fleischer, J. Grabmeier, F. Hehl, and W. Kuechlin, eds.), World Scientific, Singapore, pp. 138–162.

Petkovšek, M. [1992] *Hypergeometric solutions of linear recurrences with polynomial coefficients*, J. Symbolic Computation, **14**, 243–264.

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

Riese, A. [1995] *A Mathematica q-analogue of Zeilberger's algorithm for proving q-hypergeometric identities*, diploma thesis, J. Kepler University, Linz.

Stanton, D. [1995] *Talk* at the Workshop "Special Functions, q-Series and Related Topics", organized by the Fields Institute for Research in Mathematical Sciences at University College, 12–23 June 1995, Toronto, Ontario.

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`.

Wilf, H.S. and Zeilberger, D. [1992a] *Rational function certification of multisum/integral/ 'q'identities*, Bull. Amer. Math. Soc. (N.S.), **27**, 148–153.

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

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