A COMPUTER PROOF OF A POLYNOMIAL IDENTITY IMPLYING A PARTITION THEOREM OF GÖLLNITZ

ALEXANDER BERKOVICH AND AXEL RIESE

Abstract. In this paper we give a computer proof of a new polynomial identity, which extends a recent result of Alladi and the first author. In addition, we provide computer proofs for new finite analogs of Jacobi and Euler formulas. All computer proofs are done with the aid of the new computer algebra package `qMultiSum` developed by the second author. `qMultiSum` implements an algorithmic refinement of Wilf and Zeilberger's multi-q-extension of Sister Celine's technique utilizing additional ideas of Verbaeten and Wegschaider.

1. Göllnitz's Partition Theorem and Related q-Hypergeometric Identities

In 1967, Göllnitz [6] proved the following deep partition theorem:

Theorem 1. *Let $A(N)$ denote the number of partitions of N in the form $N = n_1 + n_2 + n_3 + \cdots$, such that no part is equal to 1 or 3, and $n_i - n_{i+1} \geq 6$ with strict inequality if $n_i \equiv 0, 1, 3 \pmod 6$.*
Let $B(N)$ denote the number of partitions of N into distinct parts $\equiv 2, 4, 5 \pmod 6$. Then

$$A(N) = B(N).$$

In [2], Alladi, Andrews, and Gordon reformulated and refined Theorem 1 using the language of colored partitions. To state their theorem, we will need a few definitions.

Let $P(N, i, j, k)$ denote the number of partitions of N into parts occurring in three (primary) colors ordered as

$$\mathbf{A} < \mathbf{B} < \mathbf{C}, \tag{1.1}$$

such that parts in the same color are distinct and the number of parts in colors $\mathbf{A}, \mathbf{B}, \mathbf{C}$ is equal to i, j, k, respectively.

Next, consider partitions π, such that parts equal to 1 may occur in three primary colors (1.1), but parts ≥ 2 may occur in six colors ordered as

$$\mathbf{AB} < \mathbf{AC} < \mathbf{A} < \mathbf{BC} < \mathbf{B} < \mathbf{C}. \tag{1.2}$$

In addition, the gap between adjacent parts is required to be ≥ 1, where the gap may equal 1 only if both parts are either of the same primary color or the larger part is in a color of higher order according to (1.2). These partitions π were called Type–1 partitions in [2].

We can now state the following result [2].

2000 *Mathematics Subject Classification.* Primary 05A19, 05A30, 11P82, 33F10.

The first author was partially supported by SFB-grant F1305 of the Austrian FWF.
The second author was supported by SFB-grant F1305 of the Austrian FWF.

Theorem 2. *Let $G(N,a,b,c,ab,ac,bc)$ denote the number of Type–1 partitions of N with a parts in color* **A***, …, bc parts in color* **BC***. (Note that bc is not $b\cdot c$!) Then*

$$\sum_{\substack{i,j,k\\ constraints}} G(N,a,b,c,ab,ac,bc) = P(N,i,j,k),$$

where the sum on the left is over the variables a,b,c,ab,ac,bc subject to the i,j,k-constraints, which here (and throughout) are

$$\begin{cases} i = a+ab+ac, \\ j = b+ab+bc, \\ k = c+ac+bc. \end{cases}$$

To see the connection between Theorem 1 and Theorem 2, we denote part n in color **A** as $\mathbf{A}_n$, …, part n in color **BC** as $\mathbf{BC}_n$. Next, we replace the colored integers $\mathbf{A}_n,\ldots,\mathbf{BC}_n$ by regular integers according to the following rules:

$$\begin{cases} \mathbf{A}_n \to 6n-4, \\ \mathbf{B}_n \to 6n-2, \\ \mathbf{C}_n \to 6n-1, \\ \mathbf{AB}_n \to 6n-6, \quad n>1, \\ \mathbf{AC}_n \to 6n-5, \quad n>1, \\ \mathbf{BC}_n \to 6n-3, \quad n>1. \end{cases} \tag{1.3}$$

Note that this replacement converts the ordering

$$\mathbf{AB}_n < \mathbf{AC}_n < \mathbf{A}_n < \mathbf{BC}_n < \mathbf{B}_n < \mathbf{C}_n, \quad n>1$$

into the conventional ordering

$$6n-6 < 6n-5 < 6n-4 < 6n-3 < 6n-2 < 6n-1, \quad n>1.$$

Recalling that part 1 can occur only as $\mathbf{A}_1,\mathbf{B}_1,\mathbf{C}_1$, we infer that no conventional part equals 1 or 3. Also, one can easily check that under the transformations (1.3) the color-gap conditions become identical with the gap conditions in Theorem 1. So, summing over i,j,k one immediately obtains the Göllnitz partition theorem.

To prove Theorem 2, the authors of [2] stated it in an analytic form, which they called *Key Identity*, as follows:

$$\sum_{\substack{i,j,k\\ \text{constraints}}} \frac{q^{T_t+T_{ab}+T_{ac}+T_{bc-1}}(1-q^a+q^{a+bc})}{(q)_a(q)_b(q)_c(q)_{ab}(q)_{ac}(q)_{bc}} = \frac{q^{T_i+T_j+T_k}}{(q)_i(q)_j(q)_k}, \tag{1.4}$$

where

$$t = a+b+c+ab+ac+bc,$$

$$T_m = \frac{m(m+1)}{2},$$

and for $n\in\mathbb{Z}$, the q-shifted factorial of a is defined as

$$(a)_n = (a;q)_n = \begin{cases} \prod_{j=0}^{n-1}(1-aq^j), & \text{if } n>0, \\ 1, & \text{if } n=0, \\ \prod_{j=1}^{-n}(1-aq^{-j})^{-1}, & \text{if } n<0. \end{cases}$$

Several proofs of the Key Identity (1.4) have appeared in the literature [1, 2, 3, 7]. In [1], Alladi and Andrews provided a straightforward q-hypergeometric proof of (1.4). Their proof made essential use of Jackson's q-analog of Dougall's summation formula. (See, for instance, Gasper and Rahman [5, (II.21), p. 238].) In [7], Riese used his computer algebra package `qMultiSum` to find a very simple recursive proof of (1.4). In [3], Alladi and Berkovich proposed and proved a double bounded polynomial generalization of (1.4):

Theorem 3. *If i, j, k, L, M are integers, then*

$$\sum_{\substack{i,j,k\\ constraints}} q^{T_t+T_{ab}+T_{ac}+T_{bc-1}} \times \left\{ q^{bc} \begin{bmatrix} L-t+a \\ a \end{bmatrix}_q \begin{bmatrix} L-t+b \\ b \end{bmatrix}_q \begin{bmatrix} M-t+c \\ c \end{bmatrix}_q \begin{bmatrix} L-t \\ ab \end{bmatrix}_q \begin{bmatrix} M-t \\ ac \end{bmatrix}_q \begin{bmatrix} M-t \\ bc \end{bmatrix}_q \right.$$
$$\left. + \begin{bmatrix} L-t+a-1 \\ a-1 \end{bmatrix}_q \begin{bmatrix} L-t+b \\ b \end{bmatrix}_q \begin{bmatrix} M-t+c \\ c \end{bmatrix}_q \begin{bmatrix} L-t \\ ab \end{bmatrix}_q \begin{bmatrix} M-t \\ ac \end{bmatrix}_q \begin{bmatrix} M-t \\ bc-1 \end{bmatrix}_q \right\}$$
$$= \sum_{s\geq 0} q^{s(M+2)-T_s+T_{i-s}+T_{j-s}+T_{k-s}} \begin{bmatrix} L-s \\ s, i-s, j-s \end{bmatrix}_q \begin{bmatrix} M-i-j \\ k-s \end{bmatrix}_q . \tag{1.5}$$

In (1.5) and in what follows, the q-binomial and q-multinomial coefficients are defined by

$$\begin{bmatrix} n+m \\ n \end{bmatrix}_q = \begin{cases} \frac{(q^{m+1})_n}{(q)_n}, & \text{if } n \geq 0, \\ 0, & \text{otherwise,} \end{cases}$$

and

$$\begin{bmatrix} L \\ a_1, a_2, \ldots, a_n \end{bmatrix}_q = \begin{bmatrix} L \\ a_1 \end{bmatrix}_q \begin{bmatrix} L-a_1 \\ a_2 \end{bmatrix}_q \cdots \begin{bmatrix} L-a_1-a_2-\cdots-a_{n-1} \\ a_n \end{bmatrix}_q .$$

The proof of Theorem 3 given by Alladi and Berkovich [3] used recurrences along with Jackson's q-Dougall formula. It is easy to check that in the limit $L, M \to \infty$, (1.5) reduces to (1.4). Furthermore, if $L = M$, then the sum on the right in (1.5) can be evaluated with the aid of the q-Pfaff-Saalschütz formula. (See, for instance, Gasper and Rahman [5, (II.12), p. 237].) As a result, (1.5) becomes

$$\sum_{\substack{i,j,k\\ \text{constraints}}} q^{T_t+T_{ab}+T_{ac}+T_{bc-1}} \times \left\{ q^{bc} \begin{bmatrix} L-t+a \\ a \end{bmatrix}_q \begin{bmatrix} L-t+b \\ b \end{bmatrix}_q \begin{bmatrix} L-t+c \\ c \end{bmatrix}_q \begin{bmatrix} L-t \\ ab \end{bmatrix}_q \begin{bmatrix} L-t \\ ac \end{bmatrix}_q \begin{bmatrix} L-t \\ bc \end{bmatrix}_q \right.$$
$$\left. + \begin{bmatrix} L-t+a-1 \\ a-1 \end{bmatrix}_q \begin{bmatrix} L-t+b \\ b \end{bmatrix}_q \begin{bmatrix} L-t+c \\ c \end{bmatrix}_q \begin{bmatrix} L-t \\ ab \end{bmatrix}_q \begin{bmatrix} L-t \\ ac \end{bmatrix}_q \begin{bmatrix} L-t \\ bc-1 \end{bmatrix}_q \right\}$$
$$= q^{T_i+T_j+T_k} \begin{bmatrix} L-k \\ i \end{bmatrix}_q \begin{bmatrix} L-i \\ j \end{bmatrix}_q \begin{bmatrix} L-j \\ k \end{bmatrix}_q . \tag{1.6}$$

A partition theoretical interpretation of (1.6), given in [3], can be stated as follows:

Theorem 4. *Let $G_L(N,a,b,c,ab,ac,bc)$ denote $G(N,a,b,c,ab,ac,bc)$ with the additional constraint that no part exceeds $\mathbf{C}_L$. Let $P_L(N,i,j,k)$ denote $P(N,i,j,k)$ with the additional constraints*

$$\lambda(\mathbf{A}) \leq \mathbf{A}_{L-k}, \quad \lambda(\mathbf{B}) \leq \mathbf{B}_{L-i}, \quad \lambda(\mathbf{C}) \leq \mathbf{C}_{L-j},$$

where $\lambda(\mathbf{A})$ is the largest part in color $\mathbf{A}$, and $\lambda(\mathbf{B})$, $\lambda(\mathbf{C})$ have the analogous interpretation. Then, for $L \geq \max(i+j, j+k, k+i)$,

$$\sum_{\substack{i,j,k \\ constraints}} G_L(N,a,b,c,ab,ac,bc) = P_L(N,i,j,k).$$

It was pointed out in [3] that (1.6) and Theorem 4 can be employed to derive new finite versions of many classical q-hypergeometric identities, including those of Gauss, Jacobi and Lebesgue.

Since Theorem 2 deals with partitions into parts occurring in three primary colors, one may suspect that a polynomial analog of (1.4) should depend on three finitization parameters L_1, L_2, M and that (1.5) is just the special case $L_1 = L_2 = L$ of this more general result. And, indeed, further investigations led us to the following triple bounded polynomial generalization of (1.4):

Theorem 5. *Let*

$$g_{i,j,k}(L_1,L_2,M) := \sum_{\substack{i,j,k \\ constraints}} q^{T_t+T_{ab}+T_{ac}+T_{bc-1}}$$
$$\times \left\{ q^{bc} \begin{bmatrix} L_1-t+a \\ a \end{bmatrix}_q \begin{bmatrix} L_2-t+b \\ b \end{bmatrix}_q \begin{bmatrix} L_2-t \\ ab \end{bmatrix}_q \begin{bmatrix} M-t+c \\ c \end{bmatrix}_q \begin{bmatrix} M-t \\ ac \end{bmatrix}_q \begin{bmatrix} M-t \\ bc \end{bmatrix}_q \right.$$
$$\left. + \begin{bmatrix} L_1-t+a-1 \\ a-1 \end{bmatrix}_q \begin{bmatrix} L_2-t+b \\ b \end{bmatrix}_q \begin{bmatrix} L_2-t \\ ab \end{bmatrix}_q \begin{bmatrix} M-t+c \\ c \end{bmatrix}_q \begin{bmatrix} M-t \\ ac \end{bmatrix}_q \begin{bmatrix} M-t \\ bc-1 \end{bmatrix}_q \right\} \tag{1.7}$$

and

$$p_{i,j,k}(L_1,L_2,M) := \sum_{s\geq 0} q^{s(M+2)-T_s+T_{i-s}+T_{j-s}+T_{k-s}}$$
$$\times \begin{bmatrix} L_1-s \\ i-s \end{bmatrix}_q \begin{bmatrix} L_2-i \\ j-s \end{bmatrix}_q \begin{bmatrix} L_2-i-j+s \\ s \end{bmatrix}_q \begin{bmatrix} M-i-j \\ k-s \end{bmatrix}_q. \tag{1.8}$$

Then

$$g_{i,j,k}(L_1,L_2,M) = p_{i,j,k}(L_1,L_2,M). \tag{1.9}$$

We wish to comment that if any one of the parameters i, j, k is set to zero, then identity (1.9) reduces to the double bounded key identity for Schur's partition theorem [4].

While it is straightforward to extend the analysis of [3] to prove (1.9), our goal here is different. We would like to use (1.9) as the testing ground for the package `qMultiSum`, which has been recently developed by Riese [7]. This package is described in some detail in Section 2. In Section 3, we will use `qMultiSum` to obtain a nested recursive proof of Theorem 5. In Section 4, we will give a computer proof of the new finite version of Jacobi's formula [3] and then, propose and prove (automatically) a new finite analog of Euler's formula. In Section 5 we will conclude with a brief discussion of the "human insight", which went into the computer proof of (1.9), and with a short description of the prospects for future work.

2. The Package qMultiSum

The object of this section is to give a short account on the Mathematica package qMultiSum[1] which computes recurrences for q-hypergeometric multi-sums. The package has been written by the second author and will be described only briefly here. For more details the reader is referred to the forthcoming article [7].

The implementation is based on the method of $\mathbf{k}$-free recurrences, also known as Sister Celine's technique (developed by Wilf and Zeilberger [10]). For reasons of efficiency we also incorporated ideas from Wegschaider's [9] package MultiSum[2] for ordinary hypergeometric summation.

Let $\mathbf{n} = (n_1, \ldots, n_s)$ and $\mathbf{k} = (k_1, \ldots, k_r)$ be vectors of variables ranging over the integers. The central concept of (the q-version of) Sister Celine's technique is the computation of recurrences for multiple sums $\sum_{\mathbf{k}} F(\mathbf{n}, \mathbf{k})$, where $F(\mathbf{n}, \mathbf{k})$ is q-hypergeometric in all of its arguments. For this we proceed by computing a so-called $\mathbf{k}$-free recurrence for the summand first.

Definition 1. A q-hypergeometric function $F(\mathbf{n}, \mathbf{k})$ satisfies a $\mathbf{k}$-*free recurrence*, if there exist a finite set S of integer tuples of length $s + r$ and polynomials $\sigma_{\mathbf{i},\mathbf{j}}(\mathbf{n})$ not all zero, such that

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

holds at every point $(\mathbf{n}, \mathbf{k})$ where all values of F occurring in (2.1) are well-defined. The set S is called a *structure set*.

The computation of a $\mathbf{k}$-free recurrence is done by making an Ansatz of the form (2.1) for some structure set S and undetermined $\sigma_{\mathbf{i},\mathbf{j}}$. Dividing equation (2.1) by $F(\mathbf{n}, \mathbf{k})$ we get the rational equation

$$\sum_{(\mathbf{i},\mathbf{j}) \in S} \sigma_{\mathbf{i},\mathbf{j}}(\mathbf{n})\, R_{F,\mathbf{i},\mathbf{j}}(\mathbf{n}, \mathbf{k}) = 0, \tag{2.2}$$

which after clearing denominators turns into the polynomial equation

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

Next we compare the coefficients of all power products $q^{k_1 l_1} \cdots q^{k_r l_r}$ in (2.3) with zero to get a homogeneous system of linear equations for the $\sigma_{\mathbf{i},\mathbf{j}}(\mathbf{n})$. Note that in this system the coefficients are rational functions in several variables and not simply numbers. Every non-trivial solution of the system gives rise to a $\mathbf{k}$-free recurrence.

It has been shown by Wilf and Zeilberger [10] that every so-called q-proper hypergeometric function $F(\mathbf{n}, \mathbf{k})$ satisfies a $\mathbf{k}$-free recurrence over some "rectangular" structure set $S_{\mathbf{I},\mathbf{J}} := \{(\mathbf{i}, \mathbf{j}) \mid 0 \le i_l \le I_l, 0 \le j_m \le J_m\}$, since for large enough $\mathbf{I}$ and $\mathbf{J}$ the number of unknowns exceeds the number of equations.

---

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

[2] available at *http://www.risc.uni-linz.ac.at/research/combinat/risc/software/MultiSum*

However, this result is important only from theoretical point of view, because in practice the run-time and memory demand grow very fast with the size of the structure set S. In particular, it turns out that rectangular structure sets are in general not usable, since most recurrences live over a different domain, i.e. many $(\mathbf{i},\mathbf{j}) \in S_{\mathbf{I},\mathbf{J}}$ are superfluous points with $\sigma_{\mathbf{i},\mathbf{j}} = 0$.

Hence we also generalized the concept of *P-maximal* structure sets to the q-case, leading to more satisfactory results. The underlying existence theory was originally introduced by Verbaeten [8] for single-sums in the $q = 1$ case. Since it is based on arguments from plane geometry, there is no direct generalization to multi-sums. Nevertheless, as Wegschaider [9] pointed out, P-maximal structure sets can be computed also in this situation: the idea is to start with a small rectangular structure set and then to add all those points that do not increase the degree of the polynomial on the left-hand side of (2.3). This way the number of equations in the corresponding linear system remains the same, whereas we maximize the number of unknowns.

However, there are still some cases where also this method, called *Verbaeten completion*, misses the minimal structure set. In particular, this happens with most of the identities from Section 3. Hence we omit the details (see Riese [7]) and only remark that in these specific instances we could overcome the problem by first computing the structure set for $q = 1$ and using the same for the q-case. Right now we do not have an explanation why this actually works.

We want to emphasize that the correctness of a $\mathbf{k}$-free recurrence computed by our program can be checked independently. For that one simply divides equation (2.1) by $F(\mathbf{n},\mathbf{k})$ and verifies the resulting rational function identity (2.2) formally. As Wegschaider showed this implies the correctness of the $\mathbf{k}$-free recurrence even at those points $(\mathbf{n},\mathbf{k})$ where $F(\mathbf{n},\mathbf{k}) = 0$.

Once we have computed a $\mathbf{k}$-free recurrence for the summand, the recurrence for the whole sum can be obtained by summing over it. Doing so the left-hand side of the recurrence could collapse to 0. Since we do not know of any example where this actually occurs, we do not go further into the details here (see Wegschaider [9] or Riese [7]).

Moreover, one should keep in mind that for some specific $\mathbf{n}$ certain coefficients in the recurrence for the sum might vanish. In this case one possibly has to consider extra boundaries.

3. A Computer Proof of the Triple Bounded Identity

In this section we prove algorithmically identity (1.9) which we restate for convenience as

$$g_{i,j,k}(L_1, L_2, M) = p_{i,j,k}(L_1, L_2, M). \tag{3.1}$$

For this we proceed in two steps. First we compute a recurrence for $g_{i,j,k}(L_1, L_2, M)$ and show that $p_{i,j,k}(L_1, L_2, M)$ satisfies the same recurrence. Then we prove that (3.1) holds at a certain boundary. The latter is achieved by repeating both steps for the boundary identity.

3.1. **The Recurrence.** First of all we load the package:

```
In[1]:= <<qMultiSum.m
Out[1]= Axel Riese's qMultiSum implementation version 2.1 loaded
```

Then we enter the constraints and the summands.

```
In[2]:= a = i-ab-ac; b = j-ab-bc; c = k-ac-bc;
        t = a+b+c+ab+ac+bc;
        T[m_] := m(m+1)/2;
In[3]:= gsum = q^(T[t]+T[ab]+T[ac]+T[bc-1]) qBinomial[L1-t+a,a,q] *
               qBinomial[L2-t+b,b,q] qBinomial[M-t+c,c,q] *
               qBinomial[L2-t,ab,q] qBinomial[M-t,ac,q] qBinomial[M-t,bc,q] *
               (q^bc + (1-q^a)/(1-q^(L1-t+a)) (1-q^bc)/(1-q^(M-t-bc+1)));
In[4]:= psum = q^(s(M+2)-T[s]+T[i-s]+T[j-s]+T[k-s]) *
               qBinomial[L1-s,i-s,q] qBinomial[L2-i,j-s,q] *
               qBinomial[L2-i-j+s,s,q] qBinomial[M-i-j,k-s,q];
```

The function for computing **k**-free recurrences is called `qFindRecurrence` (or `qFR` for short). It takes as arguments the summand, the recurrence variables, the summation variables, the dimensions of the initial rectangular structure set over which Verbaeten completion is performed, and some optional parameters. In our case we specify a structure set explicitly, which — as mentioned above — we obtained by investigating the $q = 1$ case.

```
In[5]:= qFindRecurrence[gsum, {L1,L2,M,i,j}, {ab,ac,bc},
                        {0,0,0,0,0}, {0,0,0},
                        StructSet -> {{0,0,0,0,0,0,0,0}, {0,1,0,0,0,0,0,0},
                                      {1,1,1,0,1,0,0,0}, {1,2,1,1,1,1,0,0}}]
```

Out[5]=

$$
\begin{aligned}
&q^{k+L_2}\, F(-1+L_1,-2+L_2,-1+M,-1+i,-1+j,-1+ab,ac,bc)+\\
&q^{k+L_2}\, F(-1+L_1,-1+L_2,-1+M,i,-1+j,ab,ac,bc)+\\
&q^{k}\, F(L_1,-1+L_2,M,i,j,ab,ac,bc)-q^{k}\, F(L_1,L_2,M,i,j,ab,ac,bc)=0
\end{aligned}
$$

Then we sum up this recurrence by applying the function `qSumRecurrence` (or `qSR` for short) to the previous result. Of course we have to provide the information that the first 5 variables are recurrence variables. Since we prefer backward shifts, we also call the function `BackwardShifts`.

```
In[6]:= qSumRecurrence[%, 5] // BackwardShifts
```

Out[6]=

$$
\begin{aligned}
&q^{L_2}\ \mathrm{SUM}(-1+L_1,-2+L_2,-1+M,-1+i,-1+j)+\\
&q^{L_2}\ \mathrm{SUM}(-1+L_1,-1+L_2,-1+M,i,-1+j)+\\
&\mathrm{SUM}(L_1,-1+L_2,M,i,j)-\mathrm{SUM}(L_1,L_2,M,i,j)=0
\end{aligned}
\tag{3.2}
$$

Next we want to show that also $p_{i,j,k}(L_1, L_2, M)$ satisfies the same recurrence. Surprisingly, with our package we are not able to find (3.2) directly. On the other hand verification is a trivial task: we simply show that already the summand of p fulfills (3.2) and therefore also p itself. For this we call the function `qCheckRecurrence` (or `qCR` for short).

```
In[7]:= qCheckRecurrence[%, psum]
Out[7]= True
```

3.2. **The Boundary.** First we note that identity (3.1) is true if any of the parameters i, j, k is negative, since both sides vanish in this case. Thus, to complete the proof it suffices to show that

$$
\tilde{g}_{i,j,k}(L_1, M) = \tilde{p}_{i,j,k}(L_1, M), \tag{3.3}
$$

where

$$\tilde{g}_{i,j,k}(L_1, M) := g_{i,j,k}(L_1, i+j-1, M)$$

and

$$\tilde{p}_{i,j,k}(L_1, M) := p_{i,j,k}(L_1, i+j-1, M).$$

To see this we have to distinguish two cases. If $L_2 > i+j-1$ we rewrite recurrence (3.2) as

$$\begin{aligned}&\mathrm{SUM}(L_1, L_2, M, i, j)\\ &\quad = q^{L_2}\ \mathrm{SUM}(-1+L_1, -2+L_2, -1+M, -1+i, -1+j) + \cdots\end{aligned}$$

and find that on the new right-hand side either j or L_2 is shifted backwards by 1. Analogously, if $L_2 \le i+j-1$ we rewrite (3.2) as

$$\begin{aligned}&\mathrm{SUM}(L_1, -1+L_2, M, i, j)\\ &\quad = -q^{L_2}\ \mathrm{SUM}(-1+L_1, -2+L_2, -1+M, -1+i, -1+j) - \cdots\end{aligned}$$

and find that on the new right-hand side either j is shifted backwards or L_2 is shifted forwards by 1. Hence, if we show (3.3) for all integer valued L_1, then identity (3.1) holds for all integer valued L_1 and L_2.

To prove (3.3) we proceed as before. This means that we compute a recurrence for $\tilde{g}_{i,j,k}(L_1, M)$, check that $\tilde{p}_{i,j,k}(L_1, M)$ satisfies the same recurrence and prove (3.3) at a certain boundary.

3.2.1. *The Recurrence for the Boundary.* This time we succeed without spying out the structure set from the $q = 1$ case:

```
In[8]:= qFR[gsum /. L₂ -> i+j-1, {L₁,M,i}, {ab,ac,bc}, {0,0,0}, {0,0,1}] //
        qSR[#, 3]& // BackwardShifts
Out[8]=
```

$$q^{L_1}\ \mathrm{SUM}(-1+L_1, -1+M, -1+i) + \mathrm{SUM}(-1+L_1, M, i) - \mathrm{SUM}(L_1, M, i) = 0 \quad (3.4)$$

```
In[9]:= qCR[%, psum /. L₂ -> i+j-1]
Out[9]= True
```

3.2.2. *The Boundary of the Boundary.* Finally, to complete the proof of (3.3) and thus of (3.1), we prove the identity

$$\tilde{g}_{i,j,k}(i+j-1, M) = \tilde{p}_{i,j,k}(i+j-1, M). \tag{3.5}$$

Again we see that this is sufficient by viewing recurrence (3.4) as

$$\mathrm{SUM}(L_1, M, i) = q^{L_1}\ \mathrm{SUM}(-1+L_1, -1+M, -1+i) + \mathrm{SUM}(-1+L_1, M, i)$$

in case of $L_1 > i+j-1$, and as

$$\mathrm{SUM}(-1+L_1, M, i) = -q^{L_1}\ \mathrm{SUM}(-1+L_1, -1+M, -1+i) + \mathrm{SUM}(L_1, M, i)$$

if $L_1 \le i+j-1$. Hence, if we prove (3.5), then identity (3.3) holds for all integer valued L_1.

From the single-sum we immediately read off the relation

$$\tilde{p}_{i,j,k}(i+j-1, M) = \delta_{i,0}\, \delta_{j,0}\, q^{T_k} \begin{bmatrix} \Delta \\ k \end{bmatrix}_q ,$$

where $\Delta = M - i - j$. Hence our boundary identity to verify becomes

$$\tilde{g}_{i,j,k}(i+j-1, M) = \delta_{i,0}\,\delta_{j,0}\,q^{T_k} \begin{bmatrix} \Delta \\ k \end{bmatrix}_q . \tag{3.6}$$

The reason for switching from M to Δ here is that proving identity (3.6) for $\Delta = 0$ is easy, whereas for $M = 0$ it is not at all.

```
In[10]:= qFR[gsum /. {L1 -> i+j-1, L2 -> i+j-1, M -> Δ+i+j},
            {Δ,i,j,k}, {ab,ac,bc}, {0,0,0,0}, {0,0,0},
            StructSet -> {{0,0,0,0,0,0,0}, {0,1,1,1,1,0,0}, {1,0,0,0,0,0,0},
                          {1,0,0,1,0,0,0}, {1,0,1,1,0,0,0}, {1,0,1,1,0,0,1},
                          {1,1,0,1,0,0,0}, {1,1,0,1,0,1,0}, {1,1,1,1,0,0,0},
                          {1,1,1,2,0,1,1}}] //
          qSR[#, 4]& // BackwardShifts
Out[10]=
```

$$\begin{aligned} &q^{-1+2\Delta+2i+2j}\ \mathrm{SUM}(-1+\Delta,-1+i,-1+j,-2+k)+\\ &q^{-3+\Delta+3i+3j}\ \mathrm{SUM}(-1+\Delta,-1+i,-1+j,-1+k)-\\ &q^{\Delta+i+j}\,(-1+q^{-1+i+j})\ \mathrm{SUM}(-1+\Delta,-1+i,j,-1+k)-\\ &q^{\Delta+i+j}\,(-1+q^{-1+i+j})\ \mathrm{SUM}(-1+\Delta,i,-1+j,-1+k)+\\ &q^{\Delta+i+j}\ \mathrm{SUM}(-1+\Delta,i,j,-1+k)+\mathrm{SUM}(-1+\Delta,i,j,k)-\\ &q^{-1+\Delta+2i+2j}\ \mathrm{SUM}(\Delta,-1+i,-1+j,-1+k)-\mathrm{SUM}(\Delta,i,j,k)=0 \end{aligned} \tag{3.7}$$

Obviously $\delta_{i,0}\,\delta_{j,0}\,q^{T_k}\left[\begin{smallmatrix}\Delta\\k\end{smallmatrix}\right]_q$ is a solution of this recurrence, we only have to consider the two non-degenerate cases when $i = j = 0$ or $i = j = 1$. Note that once we have proved the validity of (3.6) for $\Delta = 0$, which follows immediately from [3, (3.7)], our recurrence implies the validity both for $\Delta > 0$ and $\Delta \le 0$. Indeed, for the case $\Delta > 0$ we rewrite (3.7) as

$$\mathrm{SUM}(\Delta,i,j,k) = q^{-1+2\Delta+2i+2j}\ \mathrm{SUM}(-1+\Delta,-1+i,-1+j,-2+k)+\cdots$$

and observe that on the new right-hand side at least one of the parameters among i, j, k is shifted backwards or Δ is shifted backwards by 1. Analogously, for the case $\Delta \le 0$ we rewrite (3.7) as

$$\mathrm{SUM}(-1+\Delta,i,j,k) = -q^{-1+2\Delta+2i+2j}\ \mathrm{SUM}(-1+\Delta,-1+i,-1+j,-2+k)-\cdots$$

and observe that on the new right-hand side at least one of the parameters among i, j, k is shifted backwards or Δ is shifted forwards by 1.

Finally, let us briefly summarize the run-times. The computation of (3.7) took approximately 35 seconds on an SGI Octane using Mathematica 4.0.1. All other computations could be carried out within less than 5 seconds.

4. Computer Proofs of New Finite Analogs of Jacobi and Euler Formulas

In [3, (5.6)] Alladi and Berkovich pointed out that (1.6) can be used to derive

$$\sum_{l=0}^{L} a^{-l}\,\frac{1+a^{2l+1}}{1+a}\,q^{T_l} = \sum_{i,j,k} a^{i-j}\,(-1)^k\,q^{T_i+T_j+T_k} \begin{bmatrix} L-k \\ i \end{bmatrix}_q \begin{bmatrix} L-i \\ j \end{bmatrix}_q \begin{bmatrix} L-j \\ k \end{bmatrix}_q , \tag{4.1}$$

from which Jacobi's formula

$$\sum_{l\geq 0}(-1)^l\,(2l+1)\,q^{T_l} = (q;q)^3_\infty,$$

follows in the limit $L\to\infty, a\to -1$. It is instructive to prove (4.1) in an automated fashion. For the right-hand side of (4.1) we obtain within 20 seconds the following recurrence of order 4:

```
In[11]:= Clear[a]

In[12]:= qFR[a^(i-j) (-1)^k q^(T[i]+T[j]+T[k]) qBinomial[L-k,i,q] *
           qBinomial[L-i,j,q] qBinomial[L-j,k,q],
           L, {i,j,k}, 2, {0,0,0}] // qSR

Out[12]=
```

$$-\left(a\,q^{9+3L}\,\mathrm{SUM}(L)\right) + q^{7+2L}\,(1-a+a^2+a\,q^{2+L})\,\mathrm{SUM}(1+L) -$$
$$(1-a+a^2)\,q^{4+L}\,(-1+q^{3+L})\,\mathrm{SUM}(2+L) +$$
$$(-a-q^{4+L}+a\,q^{4+L}-a^2\,q^{4+L})\,\mathrm{SUM}(3+L)+a\,\mathrm{SUM}(4+L)=0$$

Now we plug in the left-hand side of (4.1):

```
In[13]:= Simplify[% /. SUM[L + m_] :> SUM[L] +
                 Sum[a^(-l) (1+a^(2l+1))/(1+a) q^T[l], {l,L+1,L+m}]]

Out[13]= True
```

Once we have verified (4.1) for $L\in\{0,1,2,3\}$, we are done.

Recently, we came up with the identity

$$\sum_{l=0}^{L} q^{2(T_L-T_l)} = \sum_{i,j,k}(-1)^j\,q^{2T_i+2T_j+2T_k-i-j}\begin{bmatrix}L-k\\ i\end{bmatrix}_{q^2}\begin{bmatrix}L-i\\ j\end{bmatrix}_{q^2}\begin{bmatrix}L-j\\ k\end{bmatrix}_{q^2}, \qquad (4.2)$$

which for $L\to\infty$ turns into Euler's formula

$$1 = (-q;q)_\infty\,(q;q^2)_\infty.$$

With our package the proof can be done again in 20 seconds:

```
In[14]:= qFR[(-1)^j q^(2T[i]+2T[j]+2T[k]-i-j) qBinomial[L-k,i,q^2] *
           qBinomial[L-i,j,q^2] qBinomial[L-j,k,q^2],
           L, {i,j,k}, 2, {0,0,0}] // qSR

Out[14]=
```

$$-\left(q^{14+6L}\,\mathrm{SUM}(L)\right) + q^{12+4L}\,(1+q^{4+2L})\,\mathrm{SUM}(1+L) -$$
$$q^{6+2L}\,(-1+q^{3+L})\,(1+q^{3+L})\,\mathrm{SUM}(2+L) +$$
$$(-1-q^{8+2L})\,\mathrm{SUM}(3+L)+\mathrm{SUM}(4+L)=0$$

Plugging in the left-hand side of (4.2) gives:

```
In[15]:= Simplify[% /. SUM[L + m_] :> q^(2(L m + T[m])) *
                 (SUM[L] + Sum[q^(2(T[L]-T[l])), {l,L+1,L+m}])]

Out[15]= True
```

Again the remaining task is to verify (4.2) for $L\in\{0,1,2,3\}$.

5. Concluding Remarks

While it is apparent that the package `qMultiSum` is a valuable tool for proving q-hypergeometric identities, we would like to point out that our proof of (1.9) is not fully automated. First of all, in general, the computer generates not just one, but many recurrences. Some of these recurrences are "dead ends" in practical terms.

For instance, if one starts with the simple recursion relation

$$g_{i,j,k}(L_1, L_2, M) = g_{i,j,k}(L_1 - 1, L_2, M) + q^{L_1} g_{i-1,j,k}(L_1 - 1, L_2 - 1, M - 1) \quad (5.1)$$

and then sets $L_1 = i - 1$, one gets

$$g_{i,j,k}(i-1, L_2, M) = q^{i(M+2)-T_i+T_{j-1}+T_{k-i}} \begin{bmatrix} L_2 - i \\ j - i \end{bmatrix}_q \begin{bmatrix} L_2 - j \\ i \end{bmatrix}_q \begin{bmatrix} M - i - j \\ k - i \end{bmatrix}_q . \quad (5.2)$$

Unfortunately, we did not succeed in proving the boundary identity (5.2) with `qMultiSum` and, as a result, we were not able to complete the proof of (1.9), taking (5.1) as the starting point.

Second, crucial "human insight" was used in selecting the not so obvious boundaries

$$L_1 = i + j - 1, \quad L_2 = i + j - 1, \quad M = i + j$$

in Section 3.2. These boundaries were determined by two requirements:

(1) The validity of the identity on the boundary, along with the recursion relations, should imply the validity of the identity everywhere.
(2) The identity should take a particularly simple form on the chosen boundary.

We have already mentioned at the end of Section 2 that the recursion relations generated by the computer should be examined carefully, because on some specific hyperplanes these recurrences may become a triviality $0 = 0$. At present, this examination has to be carried out manually.

We would like to finish this article by pointing out that currently only (1.6) finds a partition theoretical interpretation. Clearly, it is highly desirable to find a partition theorem which corresponds to identity (1.9). It is encouraging that for the case when one of the parameters i, j, k in (1.9) is set to zero, such a theorem was recently found in [4].

Acknowledgment. We would like to thank Krishnaswami Alladi, George E. Andrews and Doron Zeilberger for their interest and comments on the manuscript.

References

[1] K. Alladi and G.E. Andrews, *A Quartic Key Identity for a partition theorem of Göllnitz*, J. Number Theory, **75** (1999), 220–236.
[2] K. Alladi, G.E. Andrews, and B. Gordon, *Generalizations and refinements of a partition theorem of Göllnitz*, J. Reine Angew. Math., **460** (1995), 165–188.
[3] K. Alladi and A. Berkovich, *A double bounded key identity for Göllnitz's (BIG) partition theorem*, to appear in Proc. 1999 Gainesville Conf. on Symbolic Computation and Applications, Developments in Math., Kluwer (2001), CO/0007001.
[4] K. Alladi and A. Berkovich, *A double bounded version of Schur's partition theorem*, submitted for publication, CO/0006207.
[5] G. Gasper and M. Rahman, *Basic Hypergeometric Series*, Encyclopedia Math. Appl. (G.-C. Rota, ed.), Vol. 35, Cambridge University Press, London and New York, 1990.
[6] H. Göllnitz, *Partitionen mit Differenzenbedingungen*, J. Reine Angew. Math., **225** (1967), 154–190.

[7] A. Riese, *qMultiSum — A package for proving q-hypergeometric multiple summation identities*, SFB Report **01-7**, J. Kepler University, Linz, submitted to J. Symbolic Comput., 2001.

[8] P. Verbaeten, *Rekursiebetrekkingen voor lineaire hypergeometrische funkties*, Ph.D. thesis, Katholieke Universiteit te Leuven, Heverlee, Belgium, 1976.

[9] K. Wegschaider, *Computer generated proofs of binomial multi-sum identities*, Diploma thesis, RISC, J. Kepler Universität, Linz, Austria, 1997.

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

Department of Mathematics, The Pennsylvania State University, University Park, PA 16802, USA

E-mail address: alexb@math.psu.edu

Research Institute for Symbolic Computation, Johannes Kepler University, A–4040 Linz, Austria

E-mail address: Axel.Riese@risc.uni-linz.ac.at