HYP and HYPQ

Mathematica packages for the manipulation of binomial sums and hypergeometric series respectively q-binomial sums and basic hypergeometric series

C. Krattenthaler
Institut für Mathematik der Universität Wien,
Strudlhofgasse 4, A-1090 Wien, Austria.
e-mail: KRATT@PAP.UNIVIE.AC.AT

Abstract

We give a brief description of the main features of the *Mathematica* packages HYP and HYPQ. HYP allows a convenient handling of binomial sums and hypergeometric series, while its "q-analogue", the package HYPQ, allows a convenient handling of q-binomial sums and basic hypergeometric series.

1 Introduction

In some branches of mathematics, in particular in combinatorics, research workers frequently have to deal with binomial sums and q-binomial sums. Hence it would be desirable to have something like a standardized list of binomial and q-binomial identities. An attempt for binomial identities was made by H. W. Gould [4]. However, since there are many different ways of writing the same binomial (respectively q-binomial) sum, such a list could be never satisfying. But "almost all" binomial respectively q-binomial sums admit a normal form, namely when written as a hypergeometric respectively basic hypergeometric series. And even better, there are not too many basic identities for hypergeometric respectively basic hypergeometric series (most of them being known for more than 50 years) from which "everything else" can be derived, and so are sufficient to work with. These facts were emphatically pointed out by G. E. Andrews [1] in 1974, however even now are not as accepted as they should be. A reason for this might be that until recently no attempt has been made to set up a comprehensive list of hypergeometric and basic hypergeometric identities, which are spread over the literature. Until 1990 the main sources to be recommended had to be W. N. Bailey's booklet [2] and its "enhanced version" [5] by L. J. Slater. At least Slater's book in the appendix contains a list (though narrow) of hypergeometric and basic hypergeometric summation formulas, however no transformation formulas. So, for the unexperienced it remained to be a difficult task to conveniently search for identities which might help in a particular problem. Finally, in 1990 G. Gasper and M. Rahman published a long-sought book [3] about basic hypergeometric series. The appendix of this book contains a list of the most important basic hypergeometric summations *and* transformations. This list can also be used by those who are only interested in hypergeometric identities since "almost all" hypergeometric identities have their q-analogues (i.e. basic hypergeometric analogues), and it is fairly easy to deduce from a basic hypergeometric identity its hypergeometric (i.e. $q \uparrow 1$) special case.

So, only one inconvenience remained. Everybody who ever dealt with (basic) hypergeometric series knows that doing manipulations with (basic) hypergeometric expressions amounts to working with cumbersome expressions which involve a lot of parameters. Therefore the probability of mistakes during computations done by hand is very high. Obviously, since manipulations of (basic) hypergeometric expressions usually are symbolic in nature, symbolic computer algebra must be the appropriate tool for a convenient treatment of (basic) hypergeometric expressions, in particular if a specialized package is available for this purpose. This is just what the *Mathematica* packages HYP and HYPQ are supposed to be: They provide researchers with a tool box for doing manipulations of such expressions fast and error-free. The package HYP provides tools for manipulating binomial and hypergeometric expressions, while its "q-analogue", the package HYPQ, provides tools for manipulating q-binomial and basic hypergeometric expressions.

2 The structure of HYP and HYPQ

The packages HYP and HYPQ provide tools for

(A) manipulating factorial respectively q-factorial expressions,

(B) transforming binomial respectively q-binomial sums into hypergeometric respectively basic hypergeometric notation,

(C) summing hypergeometric respectively basic hypergeometric series,

(D) transforming hypergeometric respectively basic hypergeometric series,

(E) doing formal limits of hypergeometric respectively basic hypergeometric expressions,

(F) transforming hypergeometric respectively basic hypergeometric *MATHEMATICA* expressions into TEX-code.

By now the collection of summations and transformations that is included basically consists of the Appendices II and III (respectively their hypergeometric, i.e. $q \uparrow 1$, special cases) of [3]. Of course it is planned to enlarge this collection in the future.

The philosophy behind both packages is:

Do it by yourself!

The idea is that every step in a series of manipulations should be controlled by the user. Consequently, these packages do not make any attempt to sum or transform a series automatically. (This is in contrary to the MAPLE program, which automatically tries to sum a series and, if successful, displays the result without any information how the system found the result, which is not what a serious research worker actually desires.) It is the user who has to tell the packages HYP and HYPQ which command has to be applied next. In order to help the user to decide which command *could* be applied next, for both packages a comprehensive manual (88 respectively 114 pages) giving a complete list of all the available objects with detailed descriptions, examples, and cross references is provided.

Both packages are mainly *rule*-based (a feature which is heavily supported by *Mathematica* but unfortunately not by MAPLE). This gives a lot more flexibility in doing manipulations compared to sticking to functions only.

3 Requirements and how to get the packages

In order to use these packages you need a computer with some version of *Mathematica* installed. The input files for the packages and the TEX-files for the manuals are available by anonymous `ftp` at node `pap.univie.ac.at` Contact the author for further information. In order to use the packages efficiently it suffices to have a basic knowledge of *Mathematica*. It is of great help to have an advanced knowledge of pattern matching, rules, and pure functions (cf. [7, sections 2.3, 2.4, 2.2.5]).

In the following I am describing the main features of the packages. The examples should give a flavour of how to work with these packages. The displayed output of the examples imitates the output of the packages under the DOS-version of MATHEMATICA. The output under other versions of MATHEMATICA is a little bit less attractive, but similar.

4 The package HYP

4.1 The basic objects

Of course, the basic objects of HYP are the binomial coefficient $\binom{n}{k}$, the Pochhammer symbol $(a)_n$, the Gamma function $\Gamma(x)$, and the (generalized) hypergeometric series

$$ {}_rF_s\left[\begin{matrix} a_1,\ldots,a_r \\ b_1,\ldots,b_s \end{matrix}; z\right] = \sum_{n=0}^{\infty} \frac{(a_1)_n \cdots (a_r)_n}{n!\,(b_1)_n \cdots (b_s)_n} z^n \ . $$

(All the notation and terminology is adopted from [3, pp. 1–6].) The example below shows how to enter these basic objects.

```
In[1]:= Binomial[n,k]*p[a,n]*GAMMA[x]*F[{a,b,c},{d,e},z]
```

```
         (   )  ┌                ┐
         ( n )  │  a, b, c       │
Out[1]=  (   )  F│           ; z │ Γ(x) (a)
         ( k ) 3 2│  d, e         │          n
         (   )  └                ┘
```

4.2 Converting binomial sums into hypergeometric notation

As mentioned in the Introduction, “almost all” binomial sums can be written in hypergeometric notation. For accomplishing this task quickly there are the rules `SumRegeln` and `SumF`. `SumRegeln` tries to transform a binomial sum into a form that finally can be written as a hypergeometric series. This last step (if possible) is done by the rule `SumF`. As an example we consider the Vandermonde sum.

```
In[2]:= Sum[Binomial[N,l]*Binomial[M,K-l],{l,0,Infinity}]
```

```
           ∞
          ┌───┐(        ) (   )
           \    (   M    ) ( N )
Out[2]=     〉  (        ) (   )
           /    ( K - l ) ( l )
          └───┘(        ) (   )
           l=0
```

and convert it into hypergeometric notation

```
In[3]:= %//.SumRegeln
```

```
            ∞
           ┌───┐    (-K)   (-N)
            \           l      l
          (  〉  -----------------) (1 - K + M)
            /    (1)   (1 - K + M)              K
           └───┘    l             l
            l=0
Out[3]= ------------------------------------
                         (1)
                            K
```

```
In[4]:= %/.SumF
```

```
              ┌                   ┐
                -K, -N
           F  |                 ; 1 | (1 - K + M)
          2 1 | 1 - K + M           |            K
              └                   ┘
Out[4]= --------------------------------
                       (1)
                          K
```

4.3 Summations for hypergeometric series

By now, the package HYP includes 17 summation formulas in form of rules. All the available summations (with references) are listed and displayed in the manual. Besides, there is the rule `SListe` which for a hypergeometric series gives a list of applicable summations.

```
In[5]:= %4/.SListe
Is N a nonnegative integer?
[y|n]:    y
```

```
Be sure to apply "FOrdne" before using the following information!

          S02 (1 - K + M)     S04 (1 - K + M)
                         K                   K
Out[5]= {{---------------}, {---------------}}
                (1)                 (1)
                  K                   K
```

Hence, let us apply S02 (the Vandermonde summation [5, (1.7.7)]; S04 is Gauss' ${}_2F_1[1]$-evaluation [5, (1.7.6)]).

```
In[6]:= %4/.S02
Is N a nonnegative integer?
[y|n]:    y

        (1 + M)  (1 - K + M)
               N            K
Out[6]= ---------------------
          (1)  (1 - K + M)
             K            N
```

4.4 Manipulations of hypergeometric expressions

The result in Out[6] is not completely convincing since everybody knows that the result for the Vandermonde sum Out[2] should read $\binom{M+N}{K}$. To do simplifications of hypergeometric expressions there are 15 rules which allow to do all the manipulations which are the contents of Appendix I in [5]. Besides, there are two functions for *controlled* application of rules: PosListe gives a list of all subexpressions of an expression, together with their respective positions. Ers allows the application of a rule to a specified subexpression.

Now starting with Out[6], we first want to replace $(1+M)_N\,(1-K+M)_K$ by $(1-K+M)_{K+N}$. This is done with the help of the rule erw1, which replaces $(a)_n$ by $(a)_{n+m}/(a+n)_m$ where m has to be entered on request.

```
In[7]:= PosListe[%]


          1
Out[7]= {{----, {{1}}}, {(1 + M) , {{2}}}, {(1 - K + M) , {{3}}},
          (1)                   N                      K
             K


          1
>     {-----------, {{4}}}}
       (1 - K + M)
                  N
```

```
In[8]:= Ers[%%,erw1,{3}]
top-extend by:  N

        (1 - K + M)
                   K + N
Out[8]= -----------------
        (1)  (1 - K + M)
           K            N
```

Next we want to replace $(1-K+M)_{K+N}/(1-K+M)_N$ by $(1-K+M+N)_K$. This is done with the help of `zerl` which splits $(a)_n$ into $(a)_m\,(a+m)_{n-m}$ where m has to be entered on request.

```
In[9]:= PosListe[%]

           1                  1
Out[9]= {{----, {{1}}}, {------------, {{2}}}, {(1 - K + M)      , {{3}}}}
          (1)             (1 - K + M)                     K + N
             K                       N

In[10]:= Ers[%%,zerl,{3}]
bottom-split by:  N

         (1 - K + M + N)
                        K
Out[10]= ----------------
               (1)
                  K
```

The last expression clearly is identical with $\binom{M+N}{K}$.

Additional tools are provided for reversing finite summations, for splitting summations, for shifting summation indices, for exchanging sums, etc.

4.5 Transformations of hypergeometric series

By now, the package HYP includes 35 transformation formulas in form of rules. All the available transformations (with references) are listed and displayed in the manual. Besides, there is the rule `TListe` which for a hypergeometric series gives a list of applicable transformations.

As an example, we find a proof for

```
In[11]:= Sum[Binomial[n,j]^3,{j,0,Infinity}]==
         Sum[Binomial[n,k]^2*Binomial[2*(n-k),n],{k,0,Infinity}]
```

$$\text{Out[11]= } \sum_{j=0}^{\infty} \binom{n}{j}^3 == \sum_{k=0}^{\infty} \binom{n}{k}^2 \binom{2\,(-k+n)}{n}$$

an identity that occured in the work of V. Strehl [6]. Of course, the first step is to transform `Out[11]` into hypergeometric notation.

```
In[12]:= %//.SumRegeln
```

$$\text{Out[12]= } \sum_{j=0}^{\infty} \frac{(-1)^j\,((-n)_j)^3}{((1)_j)^3} == \frac{\left(\displaystyle\sum_{k=0}^{\infty} \frac{\left(\frac{1}{2}-\frac{n}{2}\right)_k (-n)_k \left(\frac{-n}{2}\right)_k}{((1)_k)^2 \left(\frac{1}{2}-n\right)_k}\right) (1+n)_n}{(1)_n}$$

```
In[13]:= %/.SumF
```

$$\text{Out[13]= } {}_3F_2\left[\begin{matrix} -n,\ -n,\ -n \\ 1,\ 1 \end{matrix};\ -1\right] == \frac{{}_3F_2\left[\begin{matrix} -n,\ \frac{1}{2}-\frac{n}{2},\ \frac{-n}{2} \\ 1,\ \frac{1}{2}-n \end{matrix};\ 1\right] (1+n)_n}{(1)_n}$$

Now, let us continue with the left-hand side,

```
In[14]:= %[[1]]
```

```
Out[14]=  F  [ -n, -n, -n      ]
         3 2 [            ; -1 ]
             [    1, 1         ]
```

We apply TListe to find out which transformation might be applicable.

```
In[15]:= %/.TListe

Be sure to apply "FOrdne" before using the following information!

Out[15]= {{T36}}
```

There is only a single transformation provided by HYP (namely [2, Ex. 4.(iv), p. 97]) which can be applied. So, let us apply this transformation.

```
In[16]:= %14/.T36

                  -n  1 - n
          n       --, -----, 1 + n
Out[16]= 2   F    2     2           ; 1
            3 2

                        1, 1
```

Now there are several ${}_3F_2[1]$-transformations which can be applied.

```
In[17]:= %16/.TListe
Is -1 - n a nonnegative integer?
[y|n]:    n
Is n/2 a nonnegative integer?
[y|n]:    y
Is -1 - n a nonnegative integer?
[y|n]:    n
Is n/2 a nonnegative integer?
[y|n]:    y

Be sure to apply "FOrdne" before using the following information!

            n          n          n          n          n          n
Out[17]= {{2  T23}, {2  T24}, {2  T25}, {2  T26}, {2  T95}, {2  T96}}
```

(Respective references are [2, Ex. 7, p. 98], [5, ((2.3.3.7)], [2, Ex. 7, p. 98], [3, (3.1.1), $q \uparrow 1$], [3, (III.33), $q \uparrow 1$], [5, (4.3.4.2)].) The task is to apply T23–T26 (T95 and T96 are three-term relations) and parameter permutations (being performed by the rules FPerm and FTausche) in some order until we arrive at the right-hand side of Out[13]. After some trials we may find the following series of transformations.

```
In[18]:= %16/.FTausche[1,3,u]

                 1 - n         -n
               [                     ]
          n    [ -----, 1 + n, --    ]
Out[18]= 2   F [   2            2 ; 1]
            3 2[                     ]
               [       1, 1          ]

In[19]:= %/.T23

                 1 - n          n                    1 - n   n
               [ -----, -n, 1 + -    ]  [ 1, 1 - ----- - - ]
          n    [   2            2    ]  [          2     2 ]
Out[19]= 2   F [                ; 1  ] Γ[                  ]
            3 2[               n     ]  [    1 - n       n ]
               [     1, 1 - -        ]  [ 1 - -----, 1 - - ]
                            2                  2         2

In[20]:= %/.FPerm[3,1,2,u]

                     n  1 - n                1 - n   n
               [ 1 + -, -----, -n    ]  [ 1, 1 - ----- - - ]
          n    [     2    2          ]  [          2     2 ]
Out[20]= 2   F [                ; 1  ] Γ[                  ]
            3 2[               n     ]  [    1 - n       n ]
               [     1, 1 - -        ]  [ 1 - -----, 1 - - ]
                            2                  2         2

In[21]:= %/.T26
Is n a nonnegative integer?
[y|n]:    y

                    1 - n  -n                1 - n   n
               [ -n, -----, --    ]  [ 1, 1 - ----- - - ]
          n    [       2    2     ]  [          2     2 ]       1 - n   n
         2   F [              ; 1 ] Γ[                  ]   (1 - ----- - -)
            3 2[   1 - n   n      ]  [    1 - n       n ]         2     2 n
               [ 1, ----- - -     ]  [ 1 - -----, 1 - - ]
                      2    2                2         2
Out[21]= ------------------------------------------------------------------
                                              n
                                          (1 - -)
                                               2 n
```

```
In[22]:= ExpandAll[%]

                  1    n   -n                         1
             -n, - - - -, --              1, -
                  2    2   2                          2
   n                                                             1
  2    F                     ; 1   Γ                             (-)
     3 2          1                   1    n       n          2 n
          1, - - n                    - + -, 1 - -
                  2                   2    2       2
Out[22]= -----------------------------------------------------
                                      n
                               (1 - -)
                                      2 n
```

The ${}_3F_2$ agrees with the ${}_3F_2$ at the right-hand side of `Out[13]`. It is easy to show that the other terms simplify to $(1+n)_n/n!$. Hence `Out[11]` is proved.

4.6 Explicit evaluation

Very often one wants to compute special values of hypergeometric expressions, in particular when checking if some identity might be true or not. For this purpose one should use the *evaluation mode* of HYP. So far, all examples were done in the *symbolic mode* of HYP. In symbolic mode even expressions like $(a)_3$ (which equals $a(a+1)(a+2)$) are not evaluated explicitely. In evaluation mode every expressions (that can be evaluated) is evaluated explicitely. There is the switch `P` that toggles between the two modes of HYP. For example, let us evaluate the Vandermonde sum `Out[4]` for $K = 2$ (which must result into $\binom{M+N}{2}$).

```
In[23]:= %4/.K->2

            -2, -N
   F                ; 1   (-1 + M)
  2 1   -1 + M                     2
Out[23]= --------------------------
                    (1)
                       2

In[24]:= P

In[25]:= %23
Is N a nonnegative integer?
[y|n]:    n
```

```
                          2 N      (1 - N) N
         (-1 + M) M (1 + ------ - ----------)
                         -1 + M   (-1 + M) M
Out[25]= ------------------------------------
                          2

In[26]:= Factor[%]

         (-1 + M + N) (M + N)
Out[26]= --------------------
                  2

In[27]:= P
```

The last P was entered to switch back to symbolic mode.

4.7 Formal limits of hypergeometric expressions

The function Limes enables the user to do *formal* limits of hypergeometric expressions fast. However, it is left to the user to check in each particular situation if taking the limit in a formal way is actually allowed. As an example we derive Bailey's [5, Appendix, (III.7)] ${}_2F_1[\frac{1}{2}]$-sum from Whipple's [5, Appendix, (III.24)] ${}_3F_2$-sum.

```
In[28]:= Sgl33
Do you want to set values for the equation?  [y|n]:  n

              [                     ]
              [   a, 1 - a, c       ]
Out[28]=  F   [                ; 1  ]  ==
         3 2  [ d, 1 + 2 c - d      ]
              [                     ]

                                         d, 1 + 2 c - d
     1 - 2 c      [                                                                ]
〉    2         π Γ[ a   d  a   1 + 2 c - d  1   a   d  1   a   1 + 2 c - d ]
                  [ - + -, - + -----------, - - - + -, - - - + ----------- ]
                  [ 2   2  2        2       2   2   2  2   2        2      ]

In[29]:= Limes[%,c-〉Infinity]

              [                ]      1 - d
              [ a, 1 - a   1   ]     2      Sqrt[π] Γ(d)
Out[29]=  F   [          ; -   ]  == ---------------------
         2 1  [   d        2   ]       1   a   d     a   d
              [                ]     Γ(- - - + -) Γ(- + -)
                                       2   2   2     2   2
```

Incidentally, this example shows another feature of the packages: Every summation or transformation is not only available in form of a rule, but also directly in form of an equation. This saves a lot of typing in many situations.

4.8 Transforming hypergeometric and basic hypergeometric *Mathematica* expressions into TEX-code

Of course, the package HYP gives full support for writing binomial or hypergeometric expressions in TEX-code. Besides, the user may choose between PlainTEX-, LATEX-, or AMS-TEX-compatibility.

5 The package HYPQ

All the features and the organization of the package HYPQ are completely analogous to those of HYP.

Of course, the basic objects of HYPQ are the q-binomial coefficient $\begin{bmatrix} n \\ k \end{bmatrix}_q$, the upper q-factorial $(a;q)_n$, the infinite q-factorial $(a;q)_\infty$, and the basic hypergeometric series

$$ {}_r\phi_s\left[\begin{matrix} a_1,\ldots,a_r \\ b_1,\ldots,b_s \end{matrix};q,z\right] = \sum_{n=0}^{\infty} \frac{(a_1;q)_n\cdots(a_r;q)_n}{(q;q)_n(b_1;q)_n\cdots(b_s;q)_n} \left((-1)^n q^{\binom{n}{2}}\right)^{s-r+1} z^n \ . $$

(Cf. [3, pp. 1–6].) These objects are entered as follows,

```
In[1]:= Binomialq[n,k]*pq[a,n,q^2]*Gammaq[x,1/q]*ph[{a,b,c},{d,e},q,z]
```

```
         [ n ]      1        [ a, b, c          ]      2
Out[1]=  [   ]   (x;-)    φ  [                  ]  (a; q )
         [ k ]      q∞ 3 2   [             ; q, z ]        n
         [   ]               [  d, e            ]
              q
```

The conversion of q-binomal sums into basic hypergeometric notation is accomplished by the rules `SumRegeln` and `Sumph` (see the example below).

By now, the package HYPQ includes 18 rules for the simplification of q-factorial expressions, 22 summations in form of rules, 46 transformations in form of rules, and of course the same rules for reversing, splitting, exchanging, etc., sums, and the functions `PosListe` and `Ers` for the controlled application of rules. Also the writing of q-binomial or basic hypergeometric expressions in TEX-code is fully supported, again leaving the user the choice between PlainTEX-, LATEX-, or AMS-TEX-compatibility.

Once being introduced to HYP, a single example should suffice for a demonstration of the abilities of HYPQ. It concerns the q-Vandermonde sum ([3, (1.5.3)]; compare with subsections 4.2–4.4).

```
In[2]:= Sum[q^((N-l)*(R-l))*Binomialq[N,l]*Binomialq[M,R-l],{l,0,Infinity}]
```

$$\texttt{Out[2]=}\quad \sum_{l=0}^{\infty} q^{(N-l)\,(R-l)} \begin{bmatrix} M \\ R-l \end{bmatrix}_q \begin{bmatrix} N \\ l \end{bmatrix}_q$$

```
In[3]:= %//.SumRegeln
```

$$\texttt{Out[3]=}\quad \frac{q^{N\,R}\left(\displaystyle\sum_{l=0}^{\infty} \frac{q^{l}\,(q^{-N};\,q)_{l}\,(q^{-R};\,q)_{l}}{(q;\,q)_{l}\,(q^{1+M-R};\,q)_{l}}\right)(q^{1+M-R};\,q)_{R}}{(q;\,q)_{R}}$$

```
In[4]:= %/.Sumph
```

$$\texttt{Out[4]=}\quad \frac{q^{N\,R}\;{}_{2}\phi_{1}\!\left[\begin{matrix} q^{-R},\ q^{-N} \\ q^{1+M-R} \end{matrix};\ q,\ q\right](q^{1+M-R};\,q)_{R}}{(q;\,q)_{R}}$$

```
In[5]:= %/.SListe
Is R a nonnegative integer?
[y|n]:    y
Is R a nonnegative integer?
[y|n]:    y
Is N a nonnegative integer?
[y|n]:    y
```

Be sure to apply "phOrdne" before using the following information!

```
               N R   1 + M - R             N R   1 + M - R
          S02 q    (q          ; q)     S61 q    (q          ; q)
                                  R                             R
Out[5]= {{------------------------}, {------------------------}}
                   (q; q)                        (q; q)
                         R                             R

In[6]:= %4/.S02
Is N a nonnegative integer?
[y|n]:    y

          1 + M        1 + M - R
        (q     ; q)  (q          ; q)
                   N                R
Out[6]= -----------------------------
                    1 + M - R
          (q; q)  (q          ; q)
                R                 N

In[7]:= PosListe[%]

             1                1 + M                        1
Out[7]= {{-------, {{1}}}, {(q     ; q) , {{2}}}, {---------------, {{3}}},
          (q; q)                       N              1 + M - R
                R                                   (q          ; q)
                                                                   N

        1 + M - R
⟩    {(q          ; q) , {{4}}}}
                     R

In[8]:= Ers[%%,erw1,{4}]
top-extend by:  N

            1 + M - R
          (q          ; q)
                         N + R
Out[8]= ------------------------
                  1 + M - R
        (q; q)  (q          ; q)
              R                 N

In[9]:= PosListe[%]
```

```
                1                      1
Out[9]= {{-------, {{1}}}, {----------------, {{2}}},
          (q; q)               1 + M - R
                R           (q          ; q)
                                            N

         1 + M - R
>     {(q          ; q)      , {{3}}}}
                      N + R

In[10]:= Ers[%%,zerl,{3}]
bottom-split by:  N

            1 + M + N - R
          (q             ; q)
                            R
Out[10]= --------------------
               (q; q)
                     R
```

Clearly this is the q-binomial coefficient $\begin{bmatrix} M+N \\ R \end{bmatrix}_q$, as it should be.

References

[1] G. E. Andrews, *Applications of basic hypergeometric series,* SIAM Review, **16** (1974), 441–484.

[2] W. N. Bailey, *Generalized hypergeometric series,* Cambridge University Press, Cambridge, 1935.

[3] G. Gasper and M. Rahman, *Basic hypergeometric series,* Encyclopedia of Mathematics And Its Applications 35, Cambridge University Press, Cambridge, 1990.

[4] H. W. Gould, *Combinatorial identities: A standardized set of tables listing 500 binomial coefficient identities,* (rev. ed.) Morgantown, W. Va., 1972.

[5] L. J. Slater, *Generalized hypergeometric functions,* Cambridge University Press, Cambridge, 1966.

[6] V. Strehl, *Binomial identities — Combinatorial and algorithmic aspects,* preprint, 1993.

[7] S. Wolfram, *Mathematica — A system for doing mathematics by computer,* second edition, Addison–Wesley, New York, 1991.