Henrici's Friendly Monster Identity Revisited

Peter Paule*
Research Institute for Symbolic Computation (RISC)
Johannes Kepler University, A-4040 Linz, Austria
ppaule@risc.uni-linz.ac.at

November 25, 2008

Dedicated to Professor Georgy Egorychev on the occasion of his 70th birthday

Abstract

We revisit Peter Henrici's friendly monster identity to present a case study on Egorychev's method. Connections to various computer algebra approaches are drawn.

1 Introduction

Let us consider the "bonus problem" 5.94 in [4]: Show that if $w = e^{2\pi i/3}$ we have

$$\sum_{k+l+m=3n} \left(\frac{(3n)!}{k!l!m!}\right)^2 w^{m-l} = \frac{(4n)!}{n!n!(2n)!} \qquad (n \geq 0). \tag{1}$$

In the "Answers to Exercises" [4, p.526] one finds that this is a consequence of Henrici's "friendly monster" identity [7, p.118]. Namely, set $c = 1$ and compare the coefficients of x^{3n} on both sides of

$$f(c,x)f(c,wx)f(c,w^2x) = \sum_{j=0}^{\infty} \frac{(\frac{1}{2}c-\frac{1}{4})_j(\frac{1}{2}c+\frac{1}{4})_j}{(\frac{1}{3}c)_j(\frac{1}{3}c+\frac{1}{3})_j(\frac{1}{3}c+\frac{2}{3})_j} \times \frac{(\frac{4x}{9})^{3j}}{(\frac{2}{3}c-\frac{1}{3})_j(\frac{2}{3}c)_j(\frac{2}{3}c+\frac{1}{3})_j(c)_j j!}, \tag{2}$$

*Partially supported by SFB grant F1305 of the Austrian Science Foundation FWF.

where

$$f(c,x) \doteqdot \sum_{j\geq 0} \frac{x^j}{(c)_j j!} \tag{3}$$

with $(c)_j = c(c+1)\cdots(c+j-1)$ if $j \geq 1$, and $(c)_0 = 1$. In addition, it is stated that "If we replace $3n$ by $3n+1$ or $3n+2$, the given sum is zero."

Remark. The right hand side of (2) is a hypergeometric ${}_2F_7$ series; $f(c,x)$ is a ${}_0F_1$ series which is a variant of a Bessel function of first kind.

In the second edition [5, p.546] one finds an alternative solution that has been provided by the author of this note. In this alternative solution the way of simplifying the original multiple sum has been strongly inspired by *Egorychev's method*, treated extensively in the monograph [3]. In this note, we present a more detailed account of this solution; in addition, we relate Egorychev's method to some recent developments in computer algebra.

In Section 2 we use Egorychev's method to reduce the problem to a single sum (Sect. 2.1) which then is simplified by computer algebra (Sect. 2.2.1) and, alternatively, by classical hypergeometric methods (Sect. 2.2.2). Using the software package `MultiSum`, in Section 3 we simplify the friendly monster identity by a direct computer algebra attack on the originally given form. In Section 4 we use the package `GeneratingFunctions` to present an alternative computer algebra solution. Finally, in Section 5 we conclude with a discussion of the methods presented.

2 Egorychev's Method in Action

To cover the cases $3n$, $3n+1$, and $3n+2$ in one stroke, we define for $N \geq 0$,

$$S(N) \doteqdot \sum_{k,l} \left(\frac{N!}{k!l!(N-k-l)!} \right)^2 w^{(N-k-l)-l}.$$

The multiple sum in (1) is the case $N = 3n$. To simplify $S(N)$ we do not need to invoke the full power of Egorychev's method exploiting residue calculus on integral representations. As it turns out the usage of the residue functional res_z on series expansions will be sufficient.

Definition. Let $f(z) = \sum_{k=-m}^{\infty} f_n z^n$ be a (formal) Laurent series for some integer m. Then the residue functional $\mathrm{res}_z\, f(z)$ takes the coefficient of z^{-1} in the series expansion of $f(z)$, i.e.,

$$\operatorname*{res}_z f(z) \doteqdot f_{-1}.$$

An addition, for later convenience we also define the constant term functional,

$$\langle z^0 \rangle f(z) \doteqdot f_0.$$

2.1 Reduction to a single sum

We begin with a slight rewriting:

$$S(N) = \sum_{k,l} \left(\frac{N!}{k!l!(N-k-l)!} \right)^2 w^{N-k+l} = \sum_{k,l} \binom{N}{k}^2 \binom{k}{l}^2 w^{l+k}. \tag{4}$$

The last equality is obtained by replacing k with $N-k$. After this preprocessing step we reduce with Egorychev's method. Rewriting the inner sum at the right hand side as

$$\sum_l \binom{k}{l}^2 w^l = \sum_l \binom{k}{l} \operatorname*{res}_z \frac{(1+z)^k}{z^{l+1}} w^l = \operatorname*{res}_z \frac{(1+z)^k}{z} \left(1 + \frac{w}{z}\right)^k,$$

we obtain,

$$\begin{aligned} S(N) &= \sum_k \binom{N}{k}^2 w^k \operatorname*{res}_z \frac{(1+z)^k (w+z)^k}{z^{k+1}} \\ &= \langle z^0 \rangle \sum_k \binom{N}{k}^2 \left(\frac{w(1+z)(w+z)}{z} \right)^k. \end{aligned}$$

The observation,

$$\frac{w(1+z)(w+z)}{z} - 1 = \frac{(1-wz)^2}{wz} \quad \text{if} \quad w = e^{2\pi i/3},$$

suggests to invoke binomial expansion together with the fact that $\langle z^0 \rangle g(cz) = \langle z^0 \rangle g(z)$:

$$\begin{aligned} S(N) &= \langle z^0 \rangle \sum_k \binom{N}{k}^2 \left(1 + \frac{(1-wz)^2}{wz}\right)^k \\ &= \langle z^0 \rangle \sum_{k,j} \binom{N}{k}^2 \binom{k}{j} \left(\frac{(1-wz)^2}{wz} \right)^j \\ &= \sum_{k,j} \binom{N}{k}^2 \binom{k}{j} \langle z^0 \rangle \left(\frac{(1-z)^2}{z} \right)^j \\ &= \sum_{k,j} \binom{N}{k} \binom{N-j}{N-k} \binom{N}{j} \binom{2j}{j} (-1)^j. \end{aligned}$$

For the last equality we applied

$$\binom{N}{k}\binom{k}{j} = \binom{N-j}{N-k}\binom{N}{j}.$$

Finally, invoking Vandermonde's formula on the sum over k reduces $S(N)$ to the single sum

$$S(N) = \sum_j \binom{2N-j}{N}\binom{N}{j}\binom{2j}{j}(-1)^j. \tag{5}$$

2.2 Simplifying the single sum

2.2.1 Using computer algebra

Taking the single sum in (5) as input, any implementation of Zeilberger's algorithm [14] reveals that

$$(N+3)^2 S(N+3) - 4(4N+3)(4N+9)S(N) = 0 \qquad (N \geq 0). \tag{6}$$

This recurrence together with the initial values $S(0) = 1$, $S(1) = 0$, and $S(2) = 0$ proves the evaluation stated above.

2.2.2 Using classical hypergeometric machinery

We find it instructive to present an alternative evaluation of the single sum in (5), namely with classical hypergeometric methods. See [5] for an introduction to basic notions and terminology or, for a more detailed account, the monograph [1].

First of all, we rewrite (5) in hypergeometric series notation:

$$S(N) = \binom{2N}{N} {}_3F_2\left(\begin{matrix} -N, -N, \frac{1}{2} \\ -2N, 1 \end{matrix}; 4\right). \tag{7}$$

Next, consider an important but less known cubic transformation of W. N. Bailey [2, (4.06)]:

$$\begin{aligned} {}_3F_2\left(\begin{matrix} 3a, b, 3a-b+\frac{1}{2} \\ 2b, 6a-2b+1 \end{matrix}; 4x\right) &= (1-x)^{-3a} \\ \times\, {}_3F_2\left(\begin{matrix} a, a+\frac{1}{3}, a+\frac{2}{3} \\ b+\frac{1}{2}, 3a-b+1 \end{matrix}; \frac{27x^2}{4(1-x)^3}\right). \end{aligned} \tag{8}$$

For complex numbers a, b, and x the identity holds if $|x| \leq 1/4$; in this case both ${}_3F_2$ series converge absolutely. If $a = -N/3$ and $b = -N$ for a positive integer N, then identity (8) holds for all x. Namely, in this case both ${}_3F_2$ series terminate, and both sides of (8) turn into polynomials in x of degree N. In particular, note that the right hand side of (8) can be viewed as a polynomial expanded in terms of polynomials of the form $x^{2k}(1-x)^{N-3k}$ where $0 \le k \le N/3$.

Finally observe that $S(N)$ as given in (7) is nothing but $\binom{2N}{N}$ times the left hand side of (8) with the choice $a = -N/3$, $b = -N$, and $x = 1$. According to (8) this equals 0 if N is not divisible by 3. If $N = 3n$ then $S(3n)$ equals $\binom{6n}{3n}$ times the coefficient of $x^{2k}(1-x)^{3n-3k}$ with $k = n$ of the polynomial on the right hand side of (8), which gives

$$\binom{6n}{3n} \text{ times } \frac{(-n)_n(-n+\frac{1}{3})_n(-n+\frac{2}{3})_n}{(-3n+\frac{1}{2})_n(1)_n}\frac{\left(\frac{27}{4}\right)^n}{n!}.$$

This can be rewritten as $(4n)!/(n!n!(2n)!)$, in accordance with (1).

3 MultiSum in Action

Instead of first reducing the original problem to a single sum, one could apply computer algebra *directly* to the given double sum to obtain the recurrence (6).

We illustrate this approach by using Wegschaider's Mathematica package `MultiSum` [12], which is based on WZ summation [13]. We initialize by loading the package

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

The first step is to compute a "certificate recurrence" for the summand:

```
In[2]:=FindRecurrence[
         Binomial[N,k]^2 Binomial[k,l]^2 w^(k + l),{N},{k,l}]
```

```
Out[2]= <certificate recurrence>
```

Without going into details it suffices to note that the output `<certificate recurrence>` is of the form

$$\sum_{(r,s,t)\in I} p_{r,s,t}(N,w)F(N-r,k-s,l-t) = \Delta_k Q_w^{(1)}(N,k,l) + \Delta_l Q_w^{(2)}(N,k,l). \quad (9)$$

Here $F(N,k,l)$ denotes the summand $\binom{N}{k}^2\binom{k}{l}^2 w^{l+k}$, I denotes some index set for the integer shifts, the $p_{r,s,t}(N,w)$ are polynomials in N and w (free of the summation variables k and l), and the $Q_w^{(i)}(N,k,l)$ are of the form as the left hand side of (9) with different coefficient polynomials and index sets. The symbol Δ_m denotes the difference operator, i.e., $\Delta_m g(m) = g(m+1) - g(m)$. Note that for fixed N the support of $F(N,k,l)$ with respect to integers k and l is finite.

Due to the finite support property we can sum both sides of (9) over all integers k and l. This produces a recurrence for $S(N) = \sum_{k,l} F(N,k,l)$ which is executed by the command:

```
In[3]:=ShiftRecurrence[SumCertificate[%],{N,1}]
Out[3]=<w-recurrence>
```

The output `<w-recurrence>` is of the form:

$$p_{-1}(N,w)\ \mathrm{SUM}[N-1] + p_0(N,w)\ \mathrm{SUM}[N] + \cdots + p_3(N,w)\mathrm{SUM}[N+3] = 0$$

with $\mathrm{SUM}[N] = S(n)$ and where the $p_i(N,w)$ are polynomials in N and w.

Finally the recurrence (6) is obtained by the substitution $w = e^{2\pi i/3}$:

```
In[4]:=FullSimplify[% /.w->Exp[2PiI/3]]
Out[4]:={(4N+1)(4N+5)
          (4(4N+3)(4N+9)SUM[N]-(N+3)^2 SUM[N+3])==0}
```

4 GeneratingFunctions in Action

A completely different approach to Henrici's friendly monster (2) is via the use of generating functions.

Considering the special case $c = 1$ of $f(c,x)$ in (3), we define

$$F(x) \doteqdot f(1,x) = \sum_{j=0}^{\infty} \frac{x^j}{(j!)^2}.$$

Recalling $S(N) = \sum_{k+l+m=N} \left(\frac{N!}{k!l!m!}\right)^2 w^{l-m}$, it is easy to verify that

$$F(x)\ F(wx)\ F(w^2x) = \sum_{N=0}^{\infty} S(N) \frac{x^N}{(N!)^2}. \tag{10}$$

Our strategy to derive the desired closed form for $S(N)$ will make use of D-finite (also called: holonomic) closure properties. More precisely: Starting with a differential equation for $F(ux)$, we first will derive a differential equation for the left hand side of (10), which in the next step will be converted to recurrences for $S(N)/(N!)^2$ and $S(N)$, respectively. All these steps will be carried out automatically by using Mallinger's Mathematica package `GeneratingFunctions` [9]. In the Maple system analogous procedures are available; see the pioneering work [10]. For more detailed theoretical background information consult [11, Sect. 6.4].

We initialize by loading the package

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

First we transform the recurrence for the coefficient sequence $a[j] = u^j/(j!)^2$ into a differential equation for $F(ux)$:

`In[2]:=DE[u_]:=RE2DE[`$(j+1)^2 a[j+1] - u\ a[j] == 0$`,`$a[j]$`,`$g[x]$`]`

`In[3]:=DE[u]`

`Out[3]:=` $-ug[x] + g'[x] + xg''[x] = 0$

Then we derive a differential equation for $g(x) \doteqdot F(x)F(wx)F(w^2x)$:

`In[4]:=DiffEq=FullSimplify[`
`DECauchy[DECauchy[DE[1],DE[`w`],`$g[x]$`],DE[`w^2`],`$g[x]$`]/.` $w \to e^{2\pi i/3}$`]`

`Out[4]=`$108\ g[x] + 256\ x\ g'[x] == 54\ g^{(3)}[x] + x(330\ g^{(4)}[x] + x$
$\times(-64\ g''[x] + 393\ g^{(5)}[x] + 153\ x\ g^{(6)}[x] + 22\ x^2\ g^{(7)}[x] + x^3\ g^{(8)}[x]))$

The differential equation `DiffEq` for $g(x) \doteqdot F(x)F(wx)F(w^2x)$ is transformed into a recurrence for the coefficients $c[N] = S(N)/(N!)^2$ in the Taylor expansion of $g(x) = \sum_{N\geq 0} c[N]x^N$:

`In[5]:=crec=DE2RE[DiffEq,` $g[x]$`,`$c[N]$`]`

`Out[5]=`$4(4N+3)(4N+9)c[N] - (N+1)^2(N+2)^2(N+4)^4 c[N+3] == 0$

Finally, to obtain the desired recurrence for $S(N)$ we multiply $c[N]$ with $(N!)^2$; again this operation is carried out on the recurrence representations:

`In[6]:=REHadamard[crec,` $c[N+1] == (N+1)^2 c[N]$ `,` $c[N]$`]`

`Out[6]=`$-4(4N+3)(4N+9)c[N] + (N+3)^3 c[N+3] == 0$

This way we again arrived at recurrence (6).

Remarks. (a) The full form (2) of the friendly monster can be proven automatically with executing exactly the same steps!
(b) From Mallinger's package we used the procedure calls: `RE2DE`, `DECauchy`, `DE2RE`, and `REHadamard`. Full descriptions of their functionality can be found in [9].
(c) It is instructive to compare the steps of the computer derivation in this Section to Henrici's original proof in [8] which also uses differential equations. In his concluding remark [8, p.1518], Henrici seems to anticipate future computer developments: "The method used in this work has the advantage of

not requiring to be known in advance. Although the algebraic manipulations, if done by hand, soon become unmanagable, the method can be used in principle whenever the irreducible terms in the derivatives of a product satisfy a recurrence relation of the general form [as specified in the paper]."

5 Conclusion

In Section 2.1 we have seen how Egorychev's method can be applied to reduce a multiple sum to a single one. The single sum then was reduced by Zeilberger's algorithm. We have shown in Section 3 that computer algebra already can be applied to the originally given multiple sum in the version of (4). Namely, using Wegschaider's package `MultiSum`, we derived recurrences which led to the desired simplification. Despite not discussed explicitly in the text, we want to note that both of these computer algebra applications are such that the programs also produce proofs, respectively proof certificates, that can be used for (human) verification of the computer output - independently from the particular steps of the algorithms.

Moreover, three aspects of Egorychev's method should be emphasized. First, in contrast to the `MultiSum` approach, it sheds additional light on the structure of the problem. Namely, in Section 2.2.2 we have seen that it reduced the original problem to a special instance of Bailey's cubic transformation (8). Second, it is well known that computer algebra algorithms for simplifying multiple sums like `MultiSum` in various applications still struggle with efficiency. In such situations, a method like Egorychev's can be applied as a preprocessing step to reduce complexity. Third, for these reasons we suggest future investigations whether certain aspects of Egorychev's approach could be supported with symbolic computation. In the context of MacMahon's partition analysis [16] some developments have been done in this direction; see [15] and in particular the algorithm by Han [6].

Finally the algorithmic approach used in Section 4 gives rise to the following remarks. It produced no proof certificates, but replaces Henrici's cumbersome hand manipulations by automatic procedures. In addition, it provides insight into the structure of the problem, and also allows to generalize; see the Remarks above. There we pointed out that in exactly the same fashion one can derive a proof of the full monster identity (2). We have not investigated whether this applies also to the derivation carried out in Section 2. More precisely: Can the Egorychev approach as used in Section 2 be modified to prove the full monster identity?

Acknowledgement. I want to thank Professor Johann Cigler who during

my student days at the University of Vienna has introduced me and my colleagues to Egorychev's book [3] in the frame of a highly inspiring seminar.

References

[1] G. E. Andrews, R. Askey, and R. Roy, *Special Functions*, Cambridge University Press, 1999.

[2] W. N. Bailey, *Products of generalized hypergeometric series*, Proc. London Math. Soc. **28** (1928), 242–254.

[3] G. P. Egorychev, *Integral Representation and the Computation of Combinatorial Sums*, Translations of Mathematical Monographs, Vol. 59, Amer. Math. Soc., Providence, RI, 1984. (Translation of: Integral'noe predstavlenie i vychislenie kombinatornykh summ.)

[4] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics*, 1st edition, Addison-Wesley, 1989.

[5] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics*, 2nd edition, Addison-Wesley, 1994.

[6] G. N. Han, *A general algorithm for the MacMahon Operator*, Ann. of Comb. **7** (2003), 467–480.

[7] P. Henrici, *De Branges' proof of the Bieberbach conjecture: a view from computational analysis*, Sitzungsber. der Berliner Math. Ges. (1987), 105–121.

[8] P. Henrici, *A triple product theorem for hypergeometric series*, SIAM J. Math. Anal. **18** (1987), 1513–1518.

[9] C. Mallinger, *Algorithmic Manipulations and Transformations of Univariate Holonomic Functions and Sequences*, Master's Thesis (Diplomarbeit), RISC, Johannes Kepler University Linz, 1996. Available at: http://www.risc.uni-linz.ac.at/research/combinat.

[10] B. Salvy and P. Zimmermann, *Gfun: a package for the manipulation of generating and holonomic functions in one variable*, ACM Trans. Math. Software **20** (1994), 163–177.

[11] R. P. Stanley, *Enumerative Combinatorics Vol. 2*, Cambridge University Press, 1999.

[12] K. Wegschaider, *Computer Generated Proofs of Binomial Multi-Sum Identities*, Master's Thesis (Diplomarbeit), RISC, Johannes Kepler University Linz, 1997. Available at: http://www.risc.uni-linz.ac.at/research/combinat.

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

[14] D. Zeilberger, *A fast algorithm for proving terminating hypergeometric identities*, Discrete Math. **80** (1982), 207–211.

[15] G. E. Andrews and P. Paule, *MacMahon's Partition Analysis IV: Hypergeometric multisums*, Séminaire Lotharingien de Combinatoire **42** (1998), Paper B42i. (Also in: *The Andrews Festschrift: Seventeen Papers on Classical Number Theory and Combinatorics*, D. Foata and G.-N. Han, eds., pp. 189–208. Berlin, Springer, 2001.)

[16] G. E. Andrews, P. Paule and A. Riese, *MacMahon's Partition Analysis VI: A new reduction algorithm*, Annals Combin. **5** (2002), 251–270.