Symbolic Summation

Victor Adamchik

Carnegie Mellon University

How to do math with your computer.

"All binomial identities are verifiable"

D. Zeilberger.

Methods

$$S_n = \sum_{k=-\infty}^{\infty} F(n,\ k)$$

Celine's algorithm

Gosper's algorithm

Wilf-Zeilberger's algorithm

Zeilberger's algorithm

Petkovsek's algorithm

"All truths are easy to understand once they are discovered;

the point is to discover them"

Galileo Galilei.

Celine's algorithm

Her work contains the original ideas on which the later algorithms have built.

In short, the algorithm finds a recurrence for a hypergeometric summand.

The fundamental theorem: every proper hypergeometric summand does satisfy a recurrence relation.

Gosper's algorithm

It completely solves the problem of indefinite hypergeometric summation.

The algorithm finds a hypergeometric term $G(k)$ such that

$$F(k) = G(k+1) - G(k)$$

if one exists, or prove that none exists.

Wilf-Zeilberger's algorithm

It is a creative modification of Gosper' s algorithm and it's also a special case of Zeilberger' s algorithm.

In order to prove an identity of the type

$$\sum_k F(n, k) = 1$$

the algorithm finds a recurrence of the form

$$F(n+1, k) - F(n, k) = G(k+1) - G(k)$$

Zeilberger's algorithm

The algorithm finds a recurrence for a hypergeometric summand, same as in Celine's algorithm. Though the form of the recurrence that it finds is different, namely

$$\sum_{j=0}^{N} d_j(n) F(n+j, k) = G(k+1) - G(k)$$

The fundamental theorem guarantees that such recurrences always exists if F is a proper hypergeometric summand.

Petkovsek's algorithm

Finds closed form (hypergeometric) solutions to

$$\sum_{j=0}^{N} d_j(n) S(n+j) = 0$$

when such solutions exist, or it proves that they do not exist, when they do not.

Problems and Solutions - *American Mathematical Monthly*

"The real work of us mathematicians, from now until, roughly, fifty years from now, when computers won't need us anymore, is to make the transition from human-centric math to machine-centric math as smooth and efficient as possible."

D. Zeilberger.

■ Problem E3258 (*American Mathematical Monthly, 1989)*

Prove

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

Proof.

The WZ method proves the identity with a proof certifcate

$$\frac{4k(k+1)}{(2k-n-1)(2n+3)}$$

QED.

How would you use the certificate? To verify the following identity

$$F(n+1,k) - F(n,k) = G(n,k+1) - G(n,k)$$

where

$$\frac{G(n,k)}{F(n,k)} = \frac{4k(k+1)}{(2k-n-1)(2n+3)}$$

The identity can be rewritten as

$$\frac{F(n+1,k)}{F(n,k)} - 1 = \frac{G(n,k+1)}{F(n,k)} - \frac{G(n,k)}{F(n,k)}$$

or

$$\frac{F(n+1,k)}{F(n,k)} - 1 = \frac{G(n,k+1)}{F(n,k+1)} \frac{F(n,k+1)}{F(n,k)} - \frac{G(n,k)}{F(n,k)}$$

The verification boils down to a polynomial manipulation.

```
R[n_, k_] := 4 k (k + 1) / ((2 k - n - 1) (2 n + 3))
```

```
F[n_, k_] := 2^(n - 2 k - 1) / (2 k + 1) Binomial[n, 2 k]
  Binomial[2 k + 1, k] / Binomial[2 n + 1, n] (n + 2)
```

$$\frac{F[n+1,k]}{F[n,k]} - 1 == R[n,k+1]\,\frac{F[n,k+1]}{F[n,k]} - R[n,k];$$

```
FullSimplify[%]
```

```
True
```

Steps of the WZ algorithm

```
s[k_] := F[n + 1, k] - F[n, k]
```

```
s[k + 1] / s[k] // FunctionExpand // Factor
```

$$\left((-1+2k-n)(2k-n)\left(6+6k+3n+4kn-n^2\right)\right) / \left(4(1+k)(2+k)\left(6k-n+4kn-n^2\right)\right)$$

```
p[k_] := 6 k - n + 4 k n - n^2
q[k + 1] = (-1 + 2 k - n) (2 k - n);
r[k_] := 4 k (1 + k)
```

```
q[k + 1] f[k] - r[k] f[k - 1] == p[k]
```

$$-4k(1+k)\,f[-1+k] + (-1+2k-n)(2k-n)\,f[k] == 6k-n+4kn-n^2$$

```
f[k_] := -1
```

Check an initial case

```
n = 0;
Sum[2^(n - 2 k - 1) / (2 k + 1) Binomial[n, 2 k]
  Binomial[2 k + 1, k] / Binomial[2 n + 1, n] (n + 2), {k, 0, n}]
```

```
1
```

The algorithm returns the following certificate

```
G[n_, k_] := r[k]/p[k] f[k - 1] s[k]
```

```
n =.; G[n, k]/F[n, k] // FunctionExpand // Factor
```

$$\frac{4\,k\,(1+k)}{(-1+2\,k-n)\,(3+2\,n)}$$

■ Problem 10424 (*American Mathematical Monthly, 1995)*

Compute

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

```
Sum[2^k n Binomial[n - k, 2 k]/(n - k), {k, 0, n / 3}]
```

```
F[n_, k_] := 2^k n/(n - k) Binomial[n - k, 2 k]
a[k_] := Sum[d[j] F[n + j, k], {j, 0, 3}]
```

```
a[k + 1]/a[k] // FunctionExpand // Factor
```

```
p[k_] := -6 n d[0] + 33 k n d[0] - 54 k^2 n d[0] + 27 k^3 n d[0] -
   11 n^2 d[0] + 36 k n^2 d[0] - 27 k^2 n^2 d[0] - 6 n^3 d[0] +
   9 k n^3 d[0] - n^4 d[0] + 6 k d[1] - 15 k^2 d[1] + 9 k^3 d[1] -
   6 n d[1] + 26 k n d[1] - 30 k^2 n d[1] + 9 k^3 n d[1] - 11 n^2 d[1] +
   27 k n^2 d[1] - 15 k^2 n^2 d[1] - 6 n^3 d[1] + 7 k n^3 d[1] - n^4 d[1] +
   6 k d[2] - 12 k^2 d[2] + 6 k^3 d[2] - 6 n d[2] + 23 k n d[2] -
   20 k^2 n d[2] + 3 k^3 n d[2] - 11 n^2 d[2] + 20 k n^2 d[2] -
   7 k^2 n^2 d[2] - 6 n^3 d[2] + 5 k n^3 d[2] - n^4 d[2] + 6 k d[3] -
   9 k^2 d[3] + 3 k^3 d[3] - 6 n d[3] + 20 k n d[3] - 12 k^2 n d[3] +
   k^3 n d[3] - 11 n^2 d[3] + 15 k n^2 d[3] - 3 k^2 n^2 d[3] - 6 n^3 d[3] +
   3 k n^3 d[3] - n^4 d[3]
q[k + 1] = (-3 + 3 k - n) (-2 + 3 k - n) (-1 + 3 k - n) ;
r[k_] := k (-1 + 2 k) (k - n)
```

```
Clear[f];
q[k + 1] f[k] - r[k] f[k - 1] - p[k];
```

```
Collect[% /. f[k_] → c , k, Factor]
```

```
# == 0 & /@ CoefficientList[%, k];
```

```
sol = First[ Solve[%, {d[0], d[1], d[2], d[3], c}] ]
```

```
Clear[F];
Sum[d[j] F[n + j, k], {j, 0, J}];
Factor[%]
```

Here is the equation for the summand

$$F(n+3,\, k) - 2\,F(n+2,\, k) + F(n+1,\, k) - 2\,F(n,\, k) = 0$$

from which we immediately get an equation for the sum

$$S(n+3) - 2\,S(n+2) + S(n+1) - 2\,S(n) = 0$$

The solution must satisfy the following initial conditions

$$S(1) = 1$$
$$S(2) = 1$$
$$S(3) = 4$$

We solve it by using a standard technique for solving recurrence equation with constant coefficients

```
RSolve[{S[n + 3] - 2 S[n + 2] + S[n + 1] - 2 S[n] == 0,
  S[1] == 1, S[2] == 1, S[3] == 4}, S[n], n]
```

$$\left\{\left\{S[n] \to \frac{1}{2}\left(2^n + 2\,\mathrm{Cos}\left[\frac{n\,\pi}{2}\right]\right)\right\}\right\}$$

$$\sum_{k=0}^{\frac{n}{3}} \frac{2^k\, n \binom{n-k}{2\,k}}{n-k} = 2^{n-1} + \cos\left(\frac{n\,\pi}{2}\right)$$

Harmonic Sums

In this section nwe will discussan an algorithmic approach (outlined in [2]) for computing finite sums involving harmonic numbers, which are defined by

$$H_k = 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{k}$$

Our emphasis is on finding a closed form for the definite sums (to show a few) of the following forms:

$$\sum_{k=1}^{n} H_k$$

$$\sum_{k=1}^{n} k^2 H_k$$

$$\sum_{k=1}^{n} (n - 2k) \binom{n}{k} H_k$$

Note, these sums are not of the hypergeometric type. The two building blocks of this approach are the following: Zeilberger's algorithm combined with an operator method for rewriting harmonic numbers in terms of binomial coefficients:

$$\lim_{x \to 0} \frac{\partial}{\partial x} \binom{x+n}{x} = H_n \tag{1}$$

This identity (due to Issak Newton) allows us in many cases to handle harmonic sums by reducing them to a hypergeometric problem. We illustrate the method by an example. Let us find a closed from for

$$\sum_{k=1}^{n} (n - 2k) \binom{n}{k} H_k$$

It's worth noting that the above sum is not that elementary and cannot be computed by *Mathematica*. Using (1), the given sum becomes

$$\sum_{k=1}^{n} (n - 2k) \binom{n}{k} H_k = \lim_{x \to 0} \frac{\partial}{\partial x} \sum_{k=1}^{n} (n - 2k) \binom{n}{k} \binom{x+k}{k}$$

Next we apply Zeilberger's algorithm to

$$S_n(x) = \sum_{k=1}^{n} (n - 2k) \binom{n}{k} \binom{x+k}{k} \tag{2}$$

```
F[n_, k_] := (n - 2 k) Binomial[n, k] Binomial[x + k, k]
s[k_] := Evaluate[Sum[d[j] F[n + j, k], {j, 0, 2}]]
s[k + 1] / s[k] // FunctionExpand // Factor
```

```
-((-2 + k - n) (1 + k + x) (-2 k d[0] + 2 k^3 d[0] + 2 n d[0] - k n d[0] -
      5 k^2 n d[0] + n^2 d[0] + 4 k n^2 d[0] - n^3 d[0] + d[1] + k d[1] - 2 k^2 d[1] +
      n d[1] + 4 k n d[1] - 2 k^2 n d[1] - n^2 d[1] + 3 k n^2 d[1] - n^3 d[1] +
      4 k d[2] - 2 n d[2] + 6 k n d[2] - 3 n^2 d[2] + 2 k n^2 d[2] - n^3 d[2])) /
  ((1 + k)^2 (4 k d[0] - 6 k^2 d[0] + 2 k^3 d[0] - 2 n d[0] + 9 k n d[0] - 5 k^2 n d[0] -
      3 n^2 d[0] + 4 k n^2 d[0] - n^3 d[0] - 2 d[1] + 5 k d[1] - 2 k^2 d[1] - 5 n d[1] +
      8 k n d[1] - 2 k^2 n d[1] - 4 n^2 d[1] + 3 k n^2 d[1] - n^3 d[1] - 4 d[2] +
      4 k d[2] - 8 n d[2] + 6 k n d[2] - 5 n^2 d[2] + 2 k n^2 d[2] - n^3 d[2]))
```

```
p[k_] :=
  (4 k d[0] - 6 k^2 d[0] + 2 k^3 d[0] - 2 n d[0] + 9 k n d[0] -
    5 k^2 n d[0] - 3 n^2 d[0] + 4 k n^2 d[0] - n^3 d[0] - 2 d[1] +
    5 k d[1] - 2 k^2 d[1] - 5 n d[1] + 8 k n d[1] - 2 k^2 n d[1] -
    4 n^2 d[1] + 3 k n^2 d[1] - n^3 d[1] - 4 d[2] + 4 k d[2] -
    8 n d[2] + 6 k n d[2] - 5 n^2 d[2] + 2 k n^2 d[2] - n^3 d[2])
q[k_] := -(k - 3 - n) (k + x)
r[k_] := k^2
```

```
p[k + 1] q[k + 1] / (p[k] r[k + 1]) - s[k + 1] / s[k] // FunctionExpand // Simplify
```

```
0
```

```
Collect[q[k + 1] f[k] - r[k] f[k - 1] - p[k] /. f[k_] → a k + b,
 k, Simplify]
```

```
-2 k^3 (a + d[0]) + k^2 (-2 b + a (2 + n - x) + 6 d[0] + 5 n d[0] + 2 d[1] + 2 n d[1]) +
 k (b (1 + n - x) + a (2 + n) (1 + x) - 4 d[0] - 9 n d[0] - 4 n^2 d[0] -
    5 d[1] - 8 n d[1] - 3 n^2 d[1] - 4 d[2] - 6 n d[2] - 2 n^2 d[2]) +
 (2 + n) (b (1 + x) + (1 + n) (d[1] + 2 d[2] + n (d[0] + d[1] + d[2])))
```

```
# == 0 & /@ CoefficientList[%, k]
```

```
{(2 + n) (b (1 + x) + (1 + n) (d[1] + 2 d[2] + n (d[0] + d[1] + d[2]))) == 0,
 b (1 + n - x) + a (2 + n) (1 + x) - 4 d[0] - 9 n d[0] - 4 n^2 d[0] -
   5 d[1] - 8 n d[1] - 3 n^2 d[1] - 4 d[2] - 6 n d[2] - 2 n^2 d[2] == 0,
 -2 b + a (2 + n - x) + 6 d[0] + 5 n d[0] + 2 d[1] + 2 n d[1] == 0, -2 (a + d[0]) == 0}
```

```
sol = Solve[%, {d[0], d[1], d[2], a, b}]
```

Solve::svars : Equations may not give solutions for all "solve" variables. ≫

```
{{d[1] -> -((3 + 3 n + x) d[0])/(2 (1 + n)), d[2] -> d[0]/2, a -> -d[0], b -> -1/2 (-1 - n) d[0]}}
```

```
Clear[F];
Sum[d[j] F[n + j, k], {j, 0, 2}] /. sol[[1]]
```

```
d[0] F[n, k] - ((3 + 3 n + x) d[0] F[1 + n, k])/(2 (1 + n)) + 1/2 d[0] F[2 + n, k]
```

Zeilberger's algorithm returns the following recurrence relation for (2)

$$2\,(n+1)\,S_n(x) - (3\,n+3+x)\,S_{n+1}(x) + (n+1)\,S_{n+2}(x) = 0$$

Differentiate it wrt x and then setting $x = 0$, yields

$$2\,(n+1)\,S_n'(0) - 3\,(n+1)\,S_{n+1}'(0) + S_{n+2}'(0) = S_{n+1}(0) \tag{3}$$

where

$$S_n'(0) = \lim_{x\to 0} \frac{\partial}{\partial x} S_n(x)$$

Next we compute $S_{n+1}(0)$ from (2)

$$S_n(0) = \sum_{k=1}^{n} (n-2\,k)\binom{n}{k}\binom{0+k}{k} = \sum_{k=1}^{n} (n-2\,k)\binom{n}{k}$$

It is elementary to verify (by executing Gosper's algorithm) that for $n \geq 0$

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

```
Sum[(n - 2 k) Binomial[n, k], {k, 0, n}]
```

```
0
```

Thus, equation (3) becomes

$$2\,S_n'(0) - 3\,S_{n+1}'(0) + S_{n+2}'(0) = 0 \tag{4}$$

with initial conditions

$$S_0'(0) = 0,\;\; S_1'(0) = -1$$

that follow from

```
n = 0; D[Sum[(n - 2 k) Binomial[n, k] Binomial[x + k, k], {k, 0, n}], x] /. x → 0
```

```
0
```

```
n = 1; D[Sum[(n - 2 k) Binomial[n, k] Binomial[x + k, k], {k, 0, n}], x] /. x → 0
```

```
-1
```

Finally we solve (4)

```
Clear[t, n];
RSolve[{2 t[n] - 3 t[n + 1] + t[n + 2] == 0, t[0] == 0,
  t[1] == -1}, t[n], n]
```

```
{{t[n] → 1 - 2^n}}
```

to get

$$\sum_{k=1}^{n} (n-2\,k) \binom{n}{k} H_k = 1 - 2^n$$

Exercise

Find a closed form for

$$\sum_{k=1}^{n} \binom{n}{k}^2 H_k$$

```
Sum[Binomial[n, k]^2 HarmonicNumber[k], {k, 1, n}]
```

```
DifferenceRoot[Function[{ẏ, ṇ},
    {-(-1 - ṇ + n)^2 (-ṇ + n)^2 ẏ[ṇ] + (-1 - ṇ + n)^2 (3 + 5 ṇ + 3 ṇ^2 - 2 ṇ n + n^2) ẏ[1 + ṇ] -
       (1 + ṇ) (11 + 20 ṇ + 13 ṇ^2 + 3 ṇ^3 - 6 n - 10 ṇ n - 4 ṇ^2 n + 3 n^2 + 2 ṇ n^2) ẏ[2 + ṇ] +
       (1 + ṇ) (2 + ṇ)^3 ẏ[3 + ṇ] == 0, ẏ[0] == 0, ẏ[1] == 0, ẏ[2] == n^2}]][1 + n]
```

References

[1] I.Nemes, M. Petkovsek, H. S.Wilf, D. Zeilberger, How to do *Monthly* problems with your computer, *Amer. Math. Monthly*, **104**(1997), 505--519.

[2] P. Paule, C. Schneider, Computer proofs of a new family of harmonic number identities, *Adv. Appl. Math.,* **31** (2003), 359--378.

[3] Wenchang Chu, Summation formulae Involving Harmonic Numbers, *Filomat,* **26** (2012), 143--152.