Zeilberger's Algorithm – Creative Telescoping

Introduction & Examples

Noam Szyfer

April 30, 2020

University of Zurich

Overview

1. Introduction

definite sums

- Gospers algorithm: *indefinite* summation
- Zeilberger's algorithm: *definite* summation

$\rightarrow$ analogous to *definite* vs. *indefinite* integration

Indefinite

Integration

$$\int_0^a e^{-x^2}\,dx = \,?$$

Summation

$$\sum_{k=0}^{a} \binom{n}{k} = \,?$$

Definite

Integration

$$\int_{-\infty}^{\infty} e^{-x^2}\,dx = \sqrt{\pi}$$

Summation

$$\sum_{k=0}^{\infty} \binom{n}{k} = 2^n$$

what are we looking for?

We are given a sum of the form

$$\sum_k F(n,k), \text{ where } F(n,k) \text{ is hypergeometric.}$$

Question: What kind of recurrence relation can we expect?

$$F(n,k) = G(n,k+1) - G(n,k)$$ not always possible!

$$F(n+1,k) - F(n,k) = G(n,k+1) - G(n,k)$$ not always possible!

the guaranteed recurrence

We can show that we'll always find a recurrence of the form

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

polynomial in n.

This is the output of the algorithm!

forward shift operator

Denote by N the forward shift operator. Then we can write

$$a_0 f(n) + a_1 f(n+1) + \ldots + a_j f(n+j) = 0$$

as

$$(a_0 + a_1 N + \ldots + a_j N^j) f(n) = 0$$

forward shift operator: example

Example of using the forward shift operator:

$$(n^2 + 2) \cdot f(n) + 2n \cdot f(n+1) + n^{17} \cdot f(n+2) = 0$$

can be written as

$$(n^2 + 2 + 2nN + n^{17} N^2) f(n) = 0$$

↳ Maple prints the recursion like this

2. Application of the Algorithm

telescoping!

Say we have found a recurrence of the aforementioned form:

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

We can then sum over all k.

$\rightarrow$ Left side: coefficients independent of k

$\rightarrow$ Right side: telescopes to 0 (*)

three different scenarios

$$\sum_{j=0}^{J} a_j(n) f(n+j) = 0$$

There are three different scenarios:

1. $J = 1$, $a_j(n)$ (possibly) non-constant polynomials in n.
2. $J > 1$, $a_j(n)$ *constant* coefficients.
3. $J > 1$, $a_j(n)$ non-constant coefficients.

2. Application of the Algorithm

2.1 Recurrence of order 1

$J = 1$

$$a_0(n) f(n) + a_1(n) f(n+1) = 0$$

$$\Leftrightarrow \quad f(n+1) = f(n) \cdot -\frac{a_0(n)}{a_1(n)}$$

By iteration, we obtain:

$$f(n) = f(0) \cdot \prod_{j=0}^{n-1} -\frac{a_0(j)}{a_1(j)}$$

2. Application of the Algorithm

2.1 Recurrence of order 1

– Example A_1

example A_1

Task: Find a closed form of

$$f(n) = \sum_{k=0}^{\infty} \binom{n}{k}$$

We give the summand $F(n,k) := \binom{n}{k}$ to Maple, to find the desired recurrence relation.

example A_1 – maple

$$F := (n, k) \rightarrow \text{binomial}(n, k)$$

$$F := (n, k) \mapsto \binom{n}{k}$$

$$ct(F(n, k), 1, k, n, N)$$

$$N - 2, \frac{k}{-n - 1 + k}$$

example A_1 – recurrence

$$N-2, \frac{k}{-n-1+k}$$

Which translates to

$$F(n+1,k) - 2F(n,k) = G(n,k+1) - G(n,k)$$

and

$$G(n,k) = F(n,k)\cdot R(n,k) = -\binom{n}{k-1}$$

example A_1 – conclusion (1)

Summing over all k we obtain:

$$f(n+1) - 2f(n) = 0$$

$$a_0(n) = -2 \quad , \; a_1(n) = 1 \quad , \; f(0) = \sum_k \binom{0}{k} = 1 \, ,$$

$$f(n) = f(0) \prod_{j=0}^{n-1} -\frac{a_0(j)}{a_1(j)} = 1 \cdot \prod_{j=0}^{n-1} 2 = 2^n$$

example A_1 – conclusion (2)

final result

$$f(n) = \sum_{k=0}^{\infty} \binom{n}{k} = 2^n$$

2. Application of the Algorithm

2.1 Recurrence of order 1
– Example A_4

example A_4

Task: Find a closed form of

$$f(n) = \sum_k (-1)^k \binom{n}{k} \binom{2n-2k}{n+a}, \qquad a \in \mathbb{N}$$

Remark: $f(n) = 0$ for $n < a$

We give the summand $F(n,k) := (-1)^k \binom{n}{k} \binom{2n-2k}{n+a}$ to Maple, to find the desired recurrence relation.

example A_4 – maple

$$F := (n, k) \rightarrow (-1)^k \cdot \mathrm{binomial}(n, k) \cdot \mathrm{binomial}(2\,n - 2\,k, n + a)$$

$$F := (n, k) \mapsto (-1)^k \binom{n}{k} \binom{2\,n - 2\,k}{n + a}$$

$$ct(F(n, k), 1, k, n, N)$$

$$2\,n + 2 + (a - n - 1)\,N, \frac{8\left(-n + k - \frac{1}{2}\right)(n + 1)\,k}{(-n - 1 + 2\,k + a)\,(n + 1 + a)}$$

example A_4 – recurrence

$$2\,n+2+(a-n-1)\,N,\ \frac{8\left(-n+k-\frac{1}{2}\right)(n+1)\,k}{(-n-1+2\,k+a)\,(n+1+a)}$$

Which translates to

$$(2n+2)F(n,k)+(a-n-1)F(n+1,k)=G(n,k+1)-G(n,k)$$

$$G(n,k)=R(n,k)F(n,k)=(-1)^k\binom{n}{k}\frac{4k}{n+1+a}\binom{2n-2k+1}{n+a}$$

example A_4 – conclusion (1)

Summing over all k we obtain:

$$(2n+2)f(n) + (a-n-1)f(n+1) = 0$$

$a_0(n) =$ 2n+2 , $a_1(n) =$ a-n-1 , $f(a) =$ 1 ,

$$f(n) = f(a)\prod_{j=a}^{n-1} -\frac{a_0(j)}{a_1(j)} = \ldots = 2^{n-a}\binom{n}{k}$$

computation can be found in the report

example A_4 – conclusion (2)

final result

$$f(n) = \sum_k (-1)^k \binom{n}{k} \binom{2n-2k}{n+a} = 2^{n-a} \binom{n}{a}$$

2. Application of the Algorithm

2.2 Linear recurrence of order > 1

$J > 0$

$J > 1$ and $a_i(n)$ are *constant* polynomials for all i
$\rightarrow$ We have tools to solve linear recurrence relations!

2. Application of the Algorithm

2.2 Linear recurrence of order > 1 – Example B_1

example B_1

Task: Find a closed form of

$$f(n) = \sum_{0 \le k \le n/2} \frac{n}{n-k} \binom{n-k}{k}, \qquad n \ge 1$$

We give the summand $F(n,k) := \frac{n}{n-k}\binom{n-k}{k}$ to Maple, to find the desired recurrence relation.

example B_1 – maple

$$F := (n,k) \rightarrow \frac{n}{n-k} \cdot \mathrm{binomial}(n-k,k)$$

$$F := (n,k) \mapsto \frac{n \binom{n-k}{k}}{n-k}$$

$$ct(F(n,k), 1, k, n, N)$$

$$0$$

$$ct(F(n,k), 2, k, n, N)$$

$$N^2 - N - 1, \frac{k(-n+k)}{(-n-1+2k)(-n-2+2k)}$$

example B_1 – recurrence

$$N^2 - N - 1, \; \frac{k\,(-n+k)}{(-n-1+2\,k)\,(-n-2+2\,k)}$$

Which translates to

$$F(n+2,k) - F(n+1,k) - F(n,k) = G(n,k+1) - G(n,k)$$

and

$$G(n,k) = R(n,k)F(n,k) = -\frac{n}{n-2(k-1)}\binom{n-k}{k-1}$$

example B_1 – conclusion (1)

Summing over all k? ⚠ → careful, don't divide by 0

$$f(n+2) - f(n+1) - f(n) = 0$$

characteristic polynomial: $x^2 - x - 1$
roots $\alpha_1 = \frac{1-\sqrt{5}}{2}$ and $\alpha_2 = \frac{1+\sqrt{5}}{2}$

general solution: $f(n) = c_1\alpha_1^n + c_2\alpha_2^n$

example B_1 – conclusion (2)

initial conditions

$$f(1) = \sum_{0 \le k \le 1/2} F(1,k) = \frac{1}{1-0}\binom{1-0}{0} = 1$$

$$f(2) = \sum_{0 \le k \le 1} F(2,k) = \frac{2}{2-0}\binom{2-0}{0} + \frac{2}{2-1}\binom{2-1}{1} = 3$$

solve this system

$$\begin{cases} c_1\alpha_1 + c_2\alpha_2 = 1 \\ c_1\alpha_1^2 + c_2\alpha_2^2 = 3 \end{cases}$$

example B_1 – conclusion (3)

final result

$$f(n) = \sum_{0 \le k \le n/2} \frac{n}{n-k} \binom{n-k}{k} = \left(\frac{1-\sqrt{5}}{2} \right)^n + \left(\frac{1+\sqrt{5}}{2} \right)^n$$

the *Lucas Numbers*

2. Application of the Algorithm

2.3 Non-linear recurrence of order > 1

$J > 0$

$J > 1$ and not all $a_i(n)$ are *constant* polynomials
$\rightarrow$ We need Petkovšek's algorithm (chapter 8)

2. Application of the Algorithm

2.3 Non-linear recurrence of order > 1 – Example C_1

example C_1

Task: Find a recurrence formula for the Legendre polynomials, given by

$$P_n(x) = 2^{-n} \sum_k (-1)^k \binom{2n-2k}{n-k} \binom{n-k}{k} x^{n-2k}$$

We give the summand $F(n,k) := (-1)^k \binom{2n-2k}{n-k} \binom{n-k}{k} x^{n-2k}$ to Maple, to find the desired recurrence relation.

example C_1 – maple

$$F := (n, k) \to 2^{-n}(-1)^k \mathrm{binomial}(2\,n - 2\,k, n - k) \cdot \mathrm{binomial}(n - k, k) \cdot x^{n-2\,k}$$

$$F := (n, k) \mapsto 2^{-n} \cdot (-1)^k \cdot \binom{2 \cdot n - 2 \cdot k}{n - k} \cdot \binom{n - k}{k} \cdot x^{n - 2 \cdot k}$$

$$ct(F(n, k), 1, k, n, N)$$

$$0$$

$$ct(F(n, k), 2, k, n, N)$$

$$(n+2)\,N^2 - (2\,n+3)\,N\,x + n + 1, \frac{4\,k\,(n+1)\,x^2\left(-n + k - \frac{1}{2}\right)}{(-n - 1 + 2\,k)\,(-n - 2 + 2\,k)}$$

example C_1 – recurrence

$$(n+2)\,N^2 - (2\,n+3)\,N\,x + n + 1,\ \frac{4\,k\,(n+1)\,x^2\left(-n+k-\frac{1}{2}\right)}{(-n-1+2\,k)\,(-n-2+2\,k)}$$

Which translates to

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

and

$$G(n,k) = 2^{-n+1}(-1)^k(n+1)\binom{2n-2k+1}{n-k+1}\binom{n-k+1}{n-2k+2}x^{n-2k+2}$$

example C_1 – conclusion (1)

Summing over all k we obtain:

$$(n+2)P_{n+2}(x) - x(2n+3)P_{n+1}(x) + (n+1)P_n(x) = 0$$

Rearrange, shift the value of n:

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x)$$

→ *Bonnet's Recursion Formula*

3. Outlook

Next Week:

- Proof that recurrence always exists
- How exactly does the algorithm work?

Thank you!