Computer Algebra Algorithms for Orthogonal Polynomials and Special Functions

Prof. Dr. Wolfram Koepf
Department of Mathematics
University of Kassel

koepf@mathematik.uni-kassel.de

http://www.mathematik.uni-kassel.de/~koepf

Yaounde, Cameroon
March 23, 2005

UNIKASSEL VERSITÄT

Overview

In this talk I will deal with the following algorithms:

- the computation of power series representations of hypergeometric type functions (like $\frac{\arcsin(x)}{x}$)
- the computation of holonomic differential equations
- the computation of holonomic differential and recurrence equations for hypergeometric series, given the series summand (like $P_n(x) = \sum\limits_{k=0}^{n} \binom{n}{k} \binom{-n-1}{k} \left(\frac{1-x}{2}\right)^k$) (Zeilberger's algorithm)
- the verification of identities for orthogonal polynomials and special functions

Automatic Computation of Power Series

- Given an expression $f(x)$ in the variable x, one would like to find the Taylor series

$$f(x) = \sum_{k=0}^{\infty} A_k\, x^k \ ,$$

i.e., a formula for the coefficient A_k.

- For example, if $f(x) = e^x$, then

$$f(x) = \sum_{k=0}^{\infty} \frac{1}{k!}\, x^k \ ,$$

hence $A_k = \frac{1}{k!}$.

FPS Algorithm

The main idea behind the FPS algorithm is

- to compute a holonomic differential equation for $f(x)$, i.e., a homogeneous linear differential equation with polynomial coefficients,
- to convert the differential equation to a holonomic recurrence equation for A_k,
- and to solve the recurrence equation for A_k.

The above procedure is successful at least is $f(x)$ is a hypergeometric power series. Maple demonstration

Computation of Holonomic Differential Equations

- Input: expression $f(x)$.
- Compute $c_0 f(x) + c_1 f'(x) + \cdots + c_J f^{(J)}(x)$ with still undetermined coefficients c_j.
- Collect w. r. t. linearly independent functions $\in \mathbb{Q}(x)$ and determine their coefficients.
- Set these zero, and solve the corresponding linear system for the unknowns $c_0, c_1, \ldots, c_J$.
- Output: DE $:= c_0 f(x) + c_1 f'(x) + \cdots + c_J f^{(J)}(x) = 0$.

Algebra of Holonomic Functions

- We call a function that satisfies a holonomic differential equation a holonomic function.
- Sum and product of holonomic functions turn out to be holonomic.
- We call a sequence that satisfies a holonomic recurrence equation a holonomic sequence.
- Sum and product of holonomic sequences are holonomic.
- A function is holonomic iff it is the generating function of a holonomic sequence.

Hypergeometric Functions

- The power series

$$
{}_pF_q\left(\begin{matrix} a_1, \ldots, a_p \\ b_1, \ldots, b_q \end{matrix} \middle| \, x \right) = \sum_{k=0}^{\infty} A_k \, x^k = \sum_{k=0}^{\infty} a_k \ ,
$$

whose coefficients A_k have a rational term ratio

$$
\frac{a_{k+1}}{a_k} = \frac{A_{k+1} \, x^{k+1}}{A_k \, x^k} = \frac{(k+a_1)\cdots(k+a_p)}{(k+b_1)\cdots(k+b_q)} \cdot \frac{x}{k+1} \ ,
$$

is called the generalized hypergeometric function.

Coefficients of the Generalized Hypergeometric Function

- For the coefficients of the hypergeometric function that are called hypergeometric terms, one gets the formula

$$ {}_pF_q\left(\left.\begin{array}{c} a_1, \ldots, a_p \\ b_1, \ldots, b_q \end{array}\right| z\right) = \sum_{k=0}^{\infty} \frac{(a_1)_k \cdots (a_p)_k}{(b_1)_k \cdots (b_q)_k} \frac{z^k}{k!} , $$

where $(a)_k = a(a+1)\cdots(a+k-1)$ is called the Pochhammer symbol or shifted factorial.

Identification of Hypergeometric Functions

- Assume we have
$$s = \sum_{k=0}^{\infty} a_k \; .$$
- How do we find out which ${}_pF_q(x)$ this is?
- Example: $\sin x = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} x^{2k+1} \; .$
- The coefficient term ratio yields

$$\frac{a_{k+1}}{a_k} = \frac{(-1)^{k+1}}{(2k+3)!} \frac{(2k+1)!}{(-1)^k} \frac{x^{2k+3}}{x^{2k+1}} = \frac{-1}{(2k+2)(2k+3)} x^2$$

Identification Algorithm

- Input: a_k .
- Compute the term ratio

$$r_k := \frac{a_{k+1}}{a_k} ,$$

  and check whether $r_k \in \mathbb{C}(k)$ is a rational function.
- Factorize r_k.
- Output: read off the upper and lower parameters and compute an initial value, e. g. a_0.

Recurrence Equations for Hypergeometric Functions

- Given a sequence s_n, as hypergeometric sum

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

- How do we find a recurrence equation for the sum s_n?

Celine Fasenmyer's Algorithm

- Input: summand $F(n,k)$.
- Compute for suitable $I, J \in \mathbb{N}$
$$\sum_{j=0}^{J} \sum_{i=0}^{I} a_{ij} \frac{F(n+j,k+i)}{F(n,k)} \in \mathbb{Q}(n,k) \; .$$
- Bring this into rational normal form, and set the numerator coefficient list w.r.t. k zero.
- If successful, linear algebra yields $a_{ij} \in \mathbb{Q}(n)$, and therefore a k-free recurrence equation for $F(n,k)$.
- Output: Sum the resulting recurrence equation for $F(n,k)$ w.r.t. k.

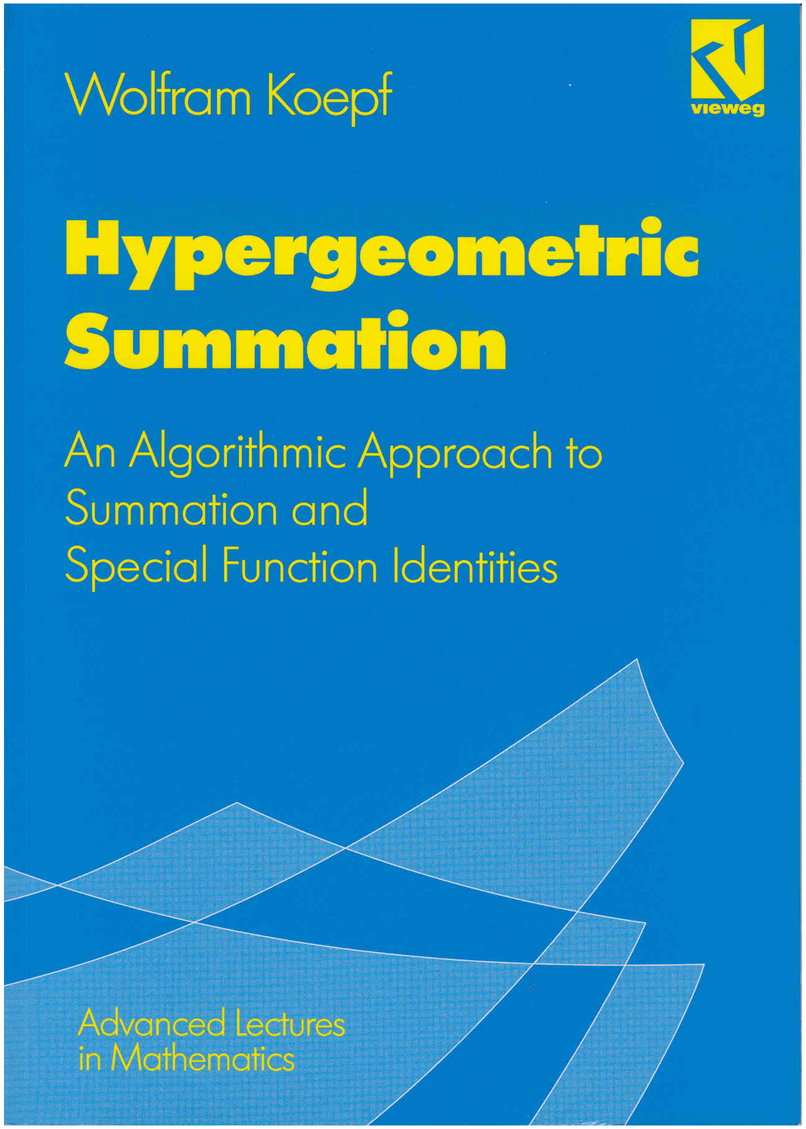
Drawbacks of Fasenmyer's Algorithm

In easy cases this algorithm succeeds, but:

- In many cases the algorithm generates a recurrence equation of too high order.
- From such a recurrence equation a lower order recurrence equation cannot be easily recovered.
- The algorithm is slow. If, e.g., $I = 2$ and $J = 2$, then already 9 linear equations have to be solved.
- Therefore the algorithm fails in many interesting cases.

The software used was developed in connection with my book

Hypergeometric Summation, Vieweg, 1998, Braunschweig/Wiesbaden



and can be downloaded from my home page:

http://www.mathematik.uni-kassel.de/~koepf

Different Representations of Legendre Polynomials

All the following hypergeometric functions represent the *Legendre Polynomials*:

$$P_n(x) = \sum_{k=0}^{n} \binom{n}{k} \binom{-n-1}{k} \left(\frac{1-x}{2}\right)^k = {}_2F_1\left(\left.\begin{matrix}-n, n+1\\ 1\end{matrix}\right| \frac{1-x}{2}\right)$$

$$= \frac{1}{2^n} \sum_{k=0}^{n} \binom{n}{k}^2 (x-1)^{n-k}(x+1)^k = \left(\frac{1-x}{2}\right)^n {}_2F_1\left(\left.\begin{matrix}-n, -n\\ 1\end{matrix}\right| \frac{1+x}{1-x}\right)$$

$$= \frac{1}{2^n} \sum_{k=0}^{\lfloor n/2 \rfloor} (-1)^k \binom{n}{k}\binom{2n-2k}{n} x^{n-2k} = \binom{2n}{n}\left(\frac{x}{2}\right)^n {}_2F_1\left(\left.\begin{matrix}-\frac{n}{2}, -\frac{n}{2}+\frac{1}{2}\\ -n+1/2\end{matrix}\right| \frac{1}{x^2}\right)$$

Recurrence Equation of the Legendre Polynomials

- This shows that special functions typically come in rather different disguises.
- However, the common recurrence equation of the different representations shows (after checking enough initial values) that they represent the same functions.
- This method is generally applicable to identify holonomic transcendental functions.
- In terms of computer algebra the recurrence equation forms a normal form for holonomic functions.

Differential Equations for Hypergeometric Series

- Zeilberger's algorithm can be adapted to generate holonomic differential equations for series
$$s(x) := \sum_{k=-\infty}^{\infty} F(x,k) \ .$$
- For this purpose, the summand $F(x,k)$ must be a hyperexponential term w.r.t. x, i.e.
$$\frac{F'(x,k)}{F(x,k)} \in \mathbb{Q}(x,k) \ .$$
- Similarly as recurrence equations holonomic differential equations form a normal form for holonomic functions.

Clausen's Formula

- Clausen's formula gives the cases when a Clausen ${}_3F_2$ function is the square of a Gauss ${}_2F_1$ function:

$$ {}_2F_1\left(\left.\begin{array}{c} a, b \\ a+b+1/2 \end{array}\right| x\right)^2 = {}_3F_2\left(\left.\begin{array}{c} 2a, 2b, a+b \\ a+b+1/2, 2a+2b \end{array}\right| x\right) . $$

- Clausen's formula can be proved (using a Cauchy product) by a recurrence equation from left to right
- or "classically" with the aid of differential equations.

Combining the Algorithms

- The following example combines some of the algorithms considered so far.

- We consider
$$F(x) = \sum_{k=0}^{\infty} \frac{x^{3k}}{(3k)!} \,.$$

- Zeilberger's algorithm finds a holonomic differential equation which can be explicitly solved.

- The FPS algorithm redetects the above representation.

Extensions

- To find recurrence and differential equations for hypergeometric and hyperexponential integrals, Almkvist and Zeilberger gave a continuous version of Gosper's algorithm. It finds hyperexponential antiderivatives if those exist.

- The resulting adaptations of the discrete versions of Zeilberger's algorithm find holonomic recurrence and differential equations for hypergeometric and hyperexponential integrals.

Extensions

- Using Cauchy's integral formula

$$h^{(n)}(x) = \frac{n!}{2\pi i} \oint \frac{h(t)}{(t-x)^{n+1}}\, dt$$

for the nth derivative makes the integration algorithm accessible for Rodrigues type expressions

$$f_n(x) = g_n(x)\, \frac{d^n}{dx^n} h_n(x)\ .$$

Orthogonal Polynomials

- Hence one can easily show that the functions

$$P_n(x) = \frac{(-1)^n}{2^n\, n!} \frac{d^n}{dx^n}(1 - x^2)^n$$

are the Legendre polynomials, and

$$L_n^{(\alpha)}(x) = \frac{e^x}{n!\, x^\alpha} \frac{d^n}{dx^n} e^{-x}\, x^{\alpha+n}$$

are the generalized Laguerre polynomials.

Extensions

- If $F(z)$ is the generating function of the sequence $a_n\, f_n(x)$, i. e.

$$F(z) = \sum_{n=0}^{\infty} a_n\, f_n(x)\, z^n \;,$$

then by Cauchy's formula and Taylor's theorem

$$f_n(x) = \frac{1}{a_n} \frac{F^{(n)}(0)}{n!} = \frac{1}{a_n} \frac{1}{2\pi i} \int\limits_{\Gamma} \frac{F(t)}{t^{n+1}}\, dt \;.$$

Laguerre Polynomials

- Hence we can easily prove the following generating function identity

$$(1-z)^{-\alpha-1}\exp\left(\frac{xz}{z-1}\right)=\sum_{n=0}^{\infty}L_n^{(\alpha)}(x)\,z^n$$

for the generalized Laguerre polynomials.

Extensions

- A further extension concerns the computation of basic hypergeometric series.

- Instead of considering series whose coefficients A_k have rational term ratio $A_{k+1}/A_k \in \mathbb{Q}(k)$, basic hypergeometric series are series whose coefficients A_k have term ratio $A_{k+1}/A_k \in \mathbb{Q}(q^k)$.

- The algorithms considered can be extended to the basic case.

Epilogue

- I hope I could give you an idea about the great algorithmic opportunities for orthogonal polynomials and special functions.
- Some of the algorithms considered are also implemented in *Macsyma*, *Mathematica*, *MuPAD* or in *Reduce*.
- I wish you much success in using them!
- If you have questions concerning this topic don't hesitate to ask me!