Finding all hypergeometric solutions of linear differential equations

M. Petkovsek, Bruno Salvy

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET AUTOMATIQUE

Finding All Hypergeometric Solutions of Linear Differential Equations

Marko PETKOVŠEK, Bruno SALVY

N° 1907

Avril 1993

PROGRAMME 2

Rapport de recherche

1994

ISSN 0249-6399

Finding All Hypergeometric Solutions of Linear Differential Equations

Marko Petkovšek and Bruno Salvy

Abstract

Hypergeometric sequences are such that the quotient of two successive terms is a fixed rational function of the index. We give a generalization of M. Petkovšek's algorithm to find all hypergeometric sequence solutions of linear *recurrences*, and we describe a program to find all hypergeometric functions that solve a linear *differential equation*.

Solutions hypergéométriques des équations différentielles linéaires

Résumé

Les suites hypergéométriques sont telles que le quotient de deux termes consécutifs est une fonction rationnelle fixe de l'indice. Nous donnons une généralisation de l'algorithme de M. Petkovšek qui détermine toutes les solutions hypergéométriques de *récurrences linéaires*, et nous décrivons un programme qui donne toutes les fonctions hypergéométriques solutions *d'équations différentielles linéaires.*

To appear in *Proceedings ISSAC'93.* M. Bronstein ed. ACM Press.

Finding All Hypergeometric Solutions of Linear Differential Equations

Marko Petkovšek
Department of Mathematics
University of Ljubljana
Slovenia.
marko-petkovsek@uni-lj.si , *Bruno Salvy*
INRIA Rocquencourt,
78153 Le Chesnay,
France.
Bruno.Salvy@inria.fr

Abstract

Hypergeometric sequences are such that the quotient of two successive terms is a fixed rational function of the index. We give a generalization of M. Petkovšek's algorithm to find all hypergeometric sequence solutions of linear *recurrences*, and we describe a program to find all hypergeometric functions that solve a linear *differential equation*.

Introduction

Most of the effort on finding closed-form solutions to linear differential equations has been focused on finding *liouvillian* solutions, *i.e.,* functions built over rational functions by application of exp, log, $\int$, algebraic closure, and field operations (see [9] for a bibliography on this).

While liouvillian functions correspond more or less to the intuitive notion of "elementary" functions and their integrals, it is natural to try to extend the existing algorithms to special functions. One reason for doing this is that special functions do arise in practice; another reason is that finding any solution permits to reduce the order of the equation under study. This may bring the equation within reach of existing algorithms since their complexity increases dramatically with the order of the equation. (Factorisation of linear differential operators is currently hopeless on equations of large order, see [4].)

The special functions we consider in this article are *generalized hypergeometric* functions. A hypergeometric series is a power series in z with $p+q$ parameters:

$$ {}_pF_q\left(\begin{array}{c} a_1,\dots,a_p \\ b_1,\dots,b_q \end{array} ; z\right) := \sum_{k=0}^{\infty} \frac{(a_1)_k\cdots(a_p)_k}{(b_1)_k\cdots(b_q)_k}\frac{z^k}{k!}, \quad (1) $$

where $(a)_k = a(a+1)\cdots(a+k-1) = \Gamma(a+k)/\Gamma(a)$ is Pochhammer's symbol. It is easy to see that a series $F = \sum_{k\geq 0} f_k$ is hypergeometric if and only if $f_0 = 1$ and the ratio f_{k+1}/f_k is a rational function of k. The a_i's (resp. b_i's) are the negatives of the zeros (resp. poles) of this function, counted with their multiplicities, and z is the leading coefficient. If -1 is not a pole we append it to the list of zeros in order to cancel out the factor $k!$.

In general (see [2, Ch. IV]), this series converges for all finite z when $p \leq q$, and for $|z| < 1$ when $p = q+1$ so that one can talk of the hypergeometric *function* in these cases. The series diverges for all $z \neq 0$ when $p > q+1$. In all cases, (1) satisfies the following differential equation

$$ [\theta \prod_{i=1}^{q} (\theta + b_i - 1) - z \prod_{j=1}^{p} (\theta + a_j)]y(z) = 0, \quad (2) $$

where $\theta = z(d/dz)$. This equation makes it possible to give a meaning to (1) even when $p > q+1$ through the introduction of Meijer's G-functions (see [2, Ch. V]).

Special cases of hypergeometric functions include $\exp(z)$, $(1-z)^a$, $-\frac{\log(1-z)}{z}$ (see [7, ch VII] for an extensive list of such special cases). While all the examples we have given are also liouvillian, there is no strict inclusion between the set of hypergeometric functions and the set of liouvillian functions. Thus the Bessel function $J_0(2i\sqrt{z})$ is hypergeometric but not liouvillian, and $\exp(e^z)$ is liouvillian but not hypergeometric (because of the type of its singularity at infinity).

The purpose of this paper is to describe an algorithm and a Maple version of it that finds all hypergeometric solutions of linear differential equations with rational coefficients. In other words, given a linear differential operator $L \in \mathbb{Q}[z][d/dz]$, we find whether there exists a factorisation $L = L_1 L_2$ with L_2 as in (2), and the values of the parameters involved. In Section 1, we describe a generalization of algorithm Hyper [6] to find m-hypergeometric solutions to linear recurrences. In Section 2, we use this algorithm to find hypergeometric functions satisfying a linear differential equation. Section 3 describes several natural extensions of our algorithm that widen the class of solutions it finds. In Section 4, we conclude with a detailed example.

1 m-hypergeometric sequences

Definition 1 *Let $\mathbb{F}$ be a field of characteristic zero. A sequence u_n is* m-hypergeometric *over $\mathbb{F}$ if there exists a rational function $r(n)$ over $\mathbb{F}$ and an integer $n_0 \geq 0$ such that, for all $n \geq n_0$, $u_{n+m}/u_n = r(n)$.*

An m-hypergeometric sequence u_n is primitive *if it satisfies no linear homogeneous recurrence with polynomial coefficients of order less than m.*

When $m = 1$, this corresponds to the usual definition of hypergeometric sequences. In this section, we describe an algorithm to find m-hypergeometric solutions to a linear recurrence equation. It reduces to algorithm Hyper [6] when $m = 1$. The reader who is not familiar with algorithm Hyper is encouraged to read this section focusing on the case $m = 1$, in order to get a feeling of how this algorithm works.

Let E denote the shift operator, $Ea(n) = a(n+1)$, and let $p_0(n)$, $p_1(n)$, …, $p_d(n)$ be rational functions of n such that $p_0, p_d \not\equiv 0$. Then

$$L := \sum_{k=0}^{d} p_k(n) E^k \tag{3}$$

is a *linear recurrence operator* of order d. With composition defined in the usual way, such operators form a ring, and they can be divided as polynomials in the indeterminate E except that factors do not commute. An operator L is *reducible* if it is a product of two operators L_1 and L_2, both of positive degree. Note that m-hypergeometric sequences are those which are annihilated by operators of the form $L_{m,r} := E^m - r(n)$ where $r(n)$ is a rational function of n. Clearly, if a sequence u satisfies $L_{m,r}u = 0$ and $L = L_1 L_{m,r}$ then u is an m-hypergeometric solution of recurrence $Ly = 0$. Therefore it is desirable to have an algorithm which will find all right factors of the form $L_{m,r}$ of a given recurrence operator L.

Fix $m \geq 1$. Let L be as in (3), normalized in such a way that its coefficients are polynomials, and assume that $L = L_1 L_{m,r}$. Denote $d_s = \lfloor (d-s)/m \rfloor$. Computing directly the division of L by $L_{m,r}$, we obtain $L = L_1 L_{m,r} + S$, where

$$S = \sum_{s=0}^{m-1} \left(\sum_{j=0}^{d_s} p_{mj+s}(n) \prod_{k=0}^{j-1} r(n+mk+s) \right) E^s$$

is the remainder. Hence for $s = 0, 1, \ldots, m-1$, r satisfies

$$\sum_{j=0}^{d_s} p_{mj+s}(n) \prod_{k=0}^{j-1} r(n+mk+s) = 0\,. \tag{4}$$

Fix s and write $t_j(n) = p_{mj+s}(n)$, $R(n) = r(n+s)$. Then (4) can be rewritten as

$$\sum_{j=0}^{d_s} t_j(n) \prod_{k=0}^{j-1} R(n+mk) = 0\,. \tag{5}$$

For $m = 1$, this nonlinear equation for the unknown rational sequence R is solved in [6]. The algorithm described there generalises to arbitrary m, as we now show.

Lemma 1 *Let $\mathbb{F}$ be a field of characteristic zero and m a positive integer. Every non-zero rational function $R(x)$ over $\mathbb{F}$ has a factorisation of the form*

$$R(x) = Z \frac{A(x)}{B(x)} \frac{C(x+m)}{C(x)} \tag{6}$$

where

1. *$Z \in \mathbb{F}$, $Z \neq 0$,*
2. *$A(x), B(x), C(x)$ are monic polynomials over $\mathbb{F}$,*
3. *$A(x), C(x)$ are relatively prime,*
4. *$B(x), C(x+m)$ are relatively prime,*
5. *$A(x), B(x+km)$ are relatively prime for every non-negative integer k.*

We omit the proof which is analogous to the one given in [6] for the special case $m = 1$ (see also [3]).

Let
$$R(n) = Z \frac{A(n)}{B(n)} \frac{C(n+m)}{C(n)}$$

where R is from (5) and Z, A, B, C are as in Lemma 1. Inserting this into (5) and clearing denominators gives

$$\sum_{j=0}^{d_s} Z^j P_j(n) C(n+mj) = 0 \tag{7}$$

where

$$P_j(n) := t_j(n) \prod_{k=0}^{j-1} A(n+mk) \prod_{k=j}^{d_s-1} B(n+mk),$$

for $j = 0, 1, \ldots, d_s$. From this it follows by the properties of A, B, C that

$$A(n) \mid t_0(n)$$

and

$$B(n) \mid t_{d_s}(n - m(d_s - 1)).$$

This leaves us with a finite choice for $A(n)$ and $B(n)$ – they are monic factors of polynomials $t_0(n)$ and $t_{d_s}(n - m(d_s - 1))$, respectively. Given the choice of A and B, a glance at the leading coefficient of (7) shows that Z satisfies an algebraic equation over $\mathbb{F}$ of degree at most d_s, so the set of possible values for Z is finite as well. For a fixed choice of A, B, and Z, we can use an algorithm for finding polynomial solutions of recurrences with polynomial coefficients (see [1] or [6]) to determine if (7) has any non-zero polynomial solution C.

After all rational solutions of (4) have been found for one value of s, those which satisfy (4) for all s, $0 \le s \le m-1$, have to be selected. This requires solving linear algebraic equations for the unknown constants.

Running this algorithm for $m = 1, 2, \ldots, d$ will give us all m-hypergeometric right factors of operator L from (3), and hence all primitive m-hypergeometric solutions of recurrence $Ly = 0$.

2 Basic Algorithm

In this section we apply algorithm Hyper — the algorithm of the previous section with $m = 1$ — to find hypergeometric solutions of linear differential equations. Our algorithm consists of three steps which we now describe.

Step 1. Recurrence for formal power series solutions. We start from a homogeneous linear differential equation of the form

$$\sum_{i=0}^{r} \left(\sum_{j=0}^{d_i} p_{ij} z^j \right) y^{(i)}(z) = 0, \tag{8}$$

where we assume that the leading coefficients p_{id_i} are different from 0. Suppose the formal power series $f(z) = \sum_{n \ge 0} f_n z^n$ satisfies (8). Then by substituting f into (8) and equating coefficients of z^n, it is well known (see [10]) that one gets a linear recurrence with polynomial coefficients:

$$\sum_{i=0}^{r} \sum_{j=0}^{d_i} p_{ij}(n+i-j)\cdots(n-j+1) f_{n+i-j} = 0, \tag{9}$$

valid for all n, with the convention that $f_k = 0$ when $k < 0$. We shall denote $M = \max(\{d_i - i; i = 1, \ldots, r\})$, then the order of the recurrence (9) is $r + M$. What makes our algorithm useful is that it is not restricted to the case $r + M = 1$.

Step 2. Solving the recurrence. Algorithm Hyper [6] finds hypergeometric-sequence solutions of equations of the form (9). Given a linear recurrence like (9) with coefficients in $\mathbb{Q}[n]$, it outputs an algebraic number Z and three monic polynomials $A(n)$, $B(n)$, and $C(n)$ in $\overline{\mathbb{Q}}[n]$ such that there exists a solution of (9) verifying

$$B(n)C(n)f_{n+1} = ZA(n)C(n+1)f_n, \tag{10}$$

with (Z, A, B, C) as in Lemma 1. It is shown in [6] that algorithm Hyper can actually produce *a basis* of such solutions. Our step 2 thus consists first in computing this basis for (9).

For each element of this basis $\{(Z_p, A_p, B_p, C_p)\}$, we have to compute the first values of the sequence. Let n_0 be the smallest non-negative integer such that $A(n_0 + k)$, $B(n_0 + k)$, $C(n_0 + k)$ are different from 0 for every integer $k \ge 0$. Then for any constant K, the sequence

$$f_n = KZ^{n-n_0} C(n) \prod_{i=n_0}^{n-1} \frac{A(i)}{B(i)}, \quad n \ge n_0,$$

satisfies (9) for all $n \ge n_0 + M$.

If $n_0 > 0$, we still have to determine $f_0, \ldots, f_{n_0-1}$. To get these values, we write down the linear system of n_0 equations obtained by setting $n = M, \ldots, M+n_0-1$ in (9), and solve this system for $f_0, \ldots, f_{n_0-1}$ and K. If the dimension of the solution is greater than 1, we can then isolate solutions with finite support. Note that the dimension of the solution may also be 0, when $A(n_0 - 1) = 0$. In this case the trailing coefficient of (9) vanishes at $n = n_0 + M - 1$ (this can be seen from the way Hyper works) and the recurrence (9) then gives rise to a linear combination of $\{f_{n_0+r-1}, \ldots, f_{n_0}\}$ which is an extra constraint on the f_n's that may be 0 or not.

Once this has been done for each element of $\{(Z_p, A_p, B_p, C_p)\}$, we have a family of solutions of two types:

$$\{f_{n,p} = a_{n,p},\ 0 \le n < n_{0,p};\ f_{n,p} = 0,\ n_{0,p} \le n\}$$

and

$$\begin{aligned} \{f_{n,p} &= a_{n,p},\ 0 \le n < n_{0,p}; \\ f_{n,p} &= Z_p^{n-n_{0,p}} C_p(n) \prod_{i=n_{0,p}}^{n-1} \frac{A_p(i)}{B_p(i)},\ n_{0,p} \le n\}, \end{aligned}$$

for some constants $a_{n,p}$. Any linear combination of these sequences is also a solution of (9) for $n \geq M$.

We now have to match the condition that $f_k = 0$ when $k < 0$ in (9). In other words we are looking for a basis of the subspace of the vector space generated by our solutions where $f_k = 0$ when $k < 0$. This is done by solving a linear system of M equations obtained by substituting a generic linear combination of our solutions into (9) for $n = 0, \ldots, M-1$. If $M \leq 0$, then all linear combinations are solutions of (9) for all n.

Step 3. Definite summation of hypergeometric sequences. We enter this step with a sequence u_n which is a linear combination (where the coefficients may be symbolic) of hypergeometric terms of the types above. We now compute the definite sum from 0 to infinity of $u_n z^n$. Note that definite summation of the sequence corresponds to $z = 1$.

Obviously sequences of the first type above correspond to polynomial solutions of (8), of which we get a basis. To deal with solutions of the second type, we first rewrite $A(n) = \prod(n+\alpha_i)$, $B(n) = \prod(n+\beta_j)$ and $C(n+n_0) = \sum_{i=0}^{\deg_n(C)} c_i n(n-1)\cdots(n-i+1)$, and then it is not difficult to see that the corresponding series is

$$Kz^{n_0}\left[\sum_{i=0}^{d_C} c_i\theta^i\right] {}_{d_A+1}F_{d_B}\left(\begin{matrix} n_0+\alpha_1,\ldots,n_0+\alpha_{d_A},1 \\ n_0+\beta_1,\ldots,n_0+\beta_{d_B}\end{matrix}\;;Zz\right),$$

where $\theta = z\frac{d}{dz}$, $d_A = \deg_n(A)$, $d_B = \deg_n(B)$, $d_C = \deg_n(C)$. This expression can be further reduced to a linear combination of hypergeometric series with polynomial coefficients by the usual formula for the derivative of an hypergeometric series.

Theorem 1 *Let $\mathcal{H}(z)$ be the vector space of hypergeometric series in z over $\overline{\mathbb{Q}}$, then Steps 1–3 find a basis of solutions of (8) in the vector space $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$.*

Proof. Let $S = \{F_1(z), \ldots, F_k(z)\}$ be the set of solutions found by the algorithm. It is clear from the description of the algorithm that $S \subset \overline{\mathbb{Q}}[z]\mathcal{H}(z)$. That S is linearly independent follows from translating a linear combination of the F_i's into a linear combination of the coefficient sequences. The last thing to prove is that any solution belonging to $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$ can be written as a linear combination of the F_i's. Let F be a solution in $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$. Then there exists a positive integer N such that for $n \geq N$, the sequence of Taylor coefficients of F is a linear combination of hypergeometric sequences which satisfy (9). We can group together sequences whose ratios are rational functions of the index. Then by [6, Corollary 5.1], each of these hypergeometric sequences is also a solution of (9) and thus will be found by HYPER. This concludes the proof. □

Note that fast algorithms exist to find polynomial solutions of linear differential equations [1], and these should be used instead of looking for solutions of the recurrence with finite support.

3 Extensions

In this section we examine simple modifications of the algorithm which make it find solutions in larger classes of expressions.

3.1 m-hypergeometric series

Many special functions can be expressed in terms of generalized hypergeometric functions evaluated at the m-th power z^m of the argument. The sequence of Taylor coefficients of these functions are often of m-hypergeometric type, and it is therefore desirable to extend our algorithm to find these.

For instance,

$$\sin z = z \cdot {}_0F_1\left(\begin{matrix} \\ 3/2 \end{matrix}\;;-\frac{z^2}{4}\right),$$

and

$$e^{\arcsin z} = {}_2F_1\left(\begin{matrix} \frac{i}{2},\frac{-i}{2} \\ 1/2 \end{matrix}\;;z^2\right) + z\cdot {}_2F_1\left(\begin{matrix} \frac{1+i}{2},\frac{1-i}{2} \\ 3/2 \end{matrix}\;;z^2\right).$$

If $\sin z = \sum_{k=0}^{\infty} u_k z^k$, and $e^{\arcsin z} = \sum_{k=0}^{\infty} v_k z^k$, the corresponding recurrences are

$$u_{n+2} = -\frac{1}{(n+1)(n+2)}u_n\,, \quad u_0 = 0,\, u_1 = 1,$$

and

$$v_{n+2} = \frac{n^2+1}{(n+1)(n+2)}v_n\,, \quad v_0 = v_1 = 1$$

(see [5, (9.3)]), and both of them are primitive 2-hypergeometric.

Using the algorithm described in Section 1 instead of HYPER in Step 2 of our algorithm gives us all primitive m-hypergeometric solutions of recurrence (9). A m-hypergeometric solution (Z, A, B, C) actually contributes for m solutions in the basis, corresponding to the sequences $f_{km}, \ldots, f_{km+m-1}$. We then proceed as before to compute the initial values of the sequence, and this may add extra contraints on the coefficients of the linear combination of these sequences. Once this has been done for all the m-hypergeometric solutions (Z, A, B, C), we compute as previously the basis of those solutions that also satisfy $f_k = 0$ when $k < 0$. The definite summation is exactly as before, except that when translating the sequence f_{km+i}, z is replaced by z^m and an extra factor of z^i has to be taken into account. One can check that the result must satisfy the following generalization of (2)

$$[\prod_{i=1}^{q}(\theta+b_i-m)-z^m\prod_{j=1}^{p}(\theta+a_j)]y(z)=0.$$

3.2 Translating the origin.

Once it is reordered, Equation (2) has the form

$$\sum_{i=0}^{r}(\gamma_i z+\epsilon_i)z^i y^{(i)}(z)=0,$$

with γ_i and ϵ_i rational constants. This means that, unless the equation has the special form

$$(az+b)y^{(r)}(z)+y^{(r-1)}(z)=0\,, \tag{11}$$

the point 0 is always a singular point of the operator. In particular, it is useless to look for solutions of hypergeometric type when 0 is not a root of the leading coefficient of the differential equation. (Solutions of (11) can be checked independently.) An algorithm for finding all hypergeometric solutions of a linear differential equation consists in first changing the variable z into $z-\alpha$, where α is a root of the leading coefficient of the equation, and then applying our algorithm. Iterating this over all the roots of the leading coefficient, as well as the change of variable $z\mapsto 1/z$ to deal with the singularity at infinity, one will get a generating set of the hypergeometric solutions. However, since the equation is not supposed to be irreducible, it is difficult in general to isolate a basis from this generating set.

When $m>1$, (11) becomes

$$y^{(r)}(z)+\sum_{j=r+1}^{r+m}(a_j z^m+b_j)z^{r+m-j}y^{(j)}(z)=0,$$

but apart from this special case, 0 is still always a singular point.

Another operation which it is natural to perform after having translated the origin is to compute the indicial equation. Then for each of the roots ν of the indicial equation we change the unknown function $y(z)$ into $z^\nu u(z)$, and look for an hypergeometric solution of this new equation. This should allow us to find solutions of the type $z^\alpha {}_qF_p(\cdot;z)$ with α an algebraic number, and a convergent ${}_qF_p$ when the singularity is regular.

3.3 Initial conditions.

There are several issues in the use of initial conditions. If the initial conditions are of the form $y^{(k)}(0)=c_k\in\overline{\mathbb{Q}}$, then obviously solving a linear system of equations will give the proper linear combination of the basis (if any). If the initial conditions are given at points different from the origin, then there is still a linear system to solve, but one has to decide when a coefficient is 0 in a class of hypergeometric constants. This seems to be a very difficult problem at the time.

3.4 Non-homogeneous equations.

If the right-hand side of (8) is a polynomial, there must be a polynomial solution of the equation [1]. One should first look for it and then apply our algorithm to the homogeneous part. If the right-hand side is not in $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$, then there are no solutions in $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$. This follows from noticing that applying a linear differential operator with polynomial coefficients to an element of $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$ yields another element of $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$. The last case to consider is when the right-hand side belongs to $\overline{\mathbb{Q}}[z]\mathcal{H}(z)$. Then a simple solution is to translate the equation at the level of coefficients and then use the extension of Hyper to non-homogeneous equations. Step 3 remains unchanged.

3.5 Reduction of order

As usual, once a solution of a linear differential equation has been found, it is possible to reduce the order by changing the unknown function. The new equation has coefficients that are polynomials in the algebraic closure of the coefficient field of the original equation, and any algorithm can be applied to it. Recursive application of our algorithm should yield solutions that are products of hypergeometric functions.

4 Detailed examples

We start with a simple example of order 3:

$$(\frac{81}{4}x^3-3x^2)y^{(3)}(x)+(\frac{567}{4}x^2-\frac{39}{2})y''(x)$$
$$+(207x-\frac{45}{2})y'(x)+45y(x)=0.$$

The associated recurrence is then of order 1:

$$(27k^2+81k+60)u_k=(4k^2+22k+30)u_{k+1}.$$

It is then trivial to find an hypergeometric sequence solution, and there are no initial conditions to satisfy. We thus get with our program:

```
> dsolvehyper(eqn,y(x));
  _C1 hypergeom([4/3, 5/3, 1], [5/2, 3], 27/4 x)
```

The following equation of order 3 is more complicated:

$$(z^3-2z^2+z)y^{(3)}(z)+(7z^2-15z+8)y''(z) \\ +(10z-13)y'(z)-2y(z)=0.$$

When translated into a recurrence, this equation gives a recurrence of order 2:

$$(n^2+10n+16)u_{n+2}-(2n^2+13n+13)u_{n+1} \\ +(n^2-3n-2)u_n=0.$$

This is where HYPER becomes necessary. It finds a basis of hypergeometric solutions which consists of only one vector:

$$Z=1, \quad A(n)=n+2, \quad B(n)=n+7, \quad C(n)=1.$$

Since in this case $M=0$ and $n_0<0$, this solution yields an hypergeometric series and this is what the program gets:

```
> dsolvehyper(eqn2,y(z));

            _C1 hypergeom([1, 2], [7], z)
```

Conclusion

The algorithm we describe in this paper is a first step towards a better use of hypergeometric functions in computer algebra. In many cases, a linear differential equation contains all the information which is needed to work with a function. However, when one is interested in using the function globally, then it becomes useful to have any kind of "closed-form". Hypergeometric functions are one class of such closed-forms.

To complete this work, it will be useful to delimit precisely the class of solutions that the extensions of Section 3 can find. Another point is that although it is possible to extend the algorithm in many directions, all these computations are rather expensive. It would be interesting to determine heuristically which of these extensions are worth the computation, and in which order. All of this will hopefully be the object of a subsequent paper.

The program implementing both the algorithm of Section 1 and the algorithm of Section 2 will soon be part of the Maple share library. Combinatorialists might use it fruitfully in conjunction with the GFUN package which provides tools for manipulating linear recurrent sequences and linear differential equations [8].

Another useful application of this program is in conjunction with Zeilberger's technique of "creative telescoping" [11]. Thus one could get a closed-form solution to some definite hypergeometric summations.

As a final note, our algorithms extend without any modification to fields of coefficients containing $\mathbb{Q}$. The only difficulty is that the computation of integer solution of polynomials over the field has to be effective, as well as the decomposition of polynomials into linear factors.

Acknowledgement This work was supported in part by the ESPRIT III Basic Research Action Programme of the E.C. under contract ALCOM II (#7141).

References

[1] ABRAMOV, S. A. Problems in computer algebra that are connected with a search for polynomial solutions of linear differential and difference equations. *Moscow University Comput. Math. Cybernet.*, 3 (1989), 63–68. Transl. from Vestn. Moskov. univ. Ser. XV Vychisl. mat. kibernet. 3, 56–60.

[2] ERDÉLYI, A. *Higher Transcendental Functions*, second ed., vol. 1-2-3. R. E. Krieger publishing Company, Inc., Malabar, Florida, 1981.

[3] GOSPER, R. W. Decision procedure for indefinite hypergeometric summation. *Proceedings of the National Academy of Sciences USA 75*, 1 (Jan. 1978), 40–42.

[4] GRIGOR'EV, D. Y. Complexity of factoring and calculating the GCD of linear ordinary differential operators. *Journal of Symbolic Computation 10* (1990), 7–37.

[5] KOEPF, W. Power series in computer algebra. *Journal of Symbolic Computation 13* (1992), 581–603.

[6] PETKOVŠEK, M. Hypergeometric solutions of linear recurrences with polynomial coefficients. *Journal of Symbolic Computation 14* (1992), 243–264.

[7] PRUDNIKOV, A. P., BRYCHKOV, Y. A., AND MARICHEV, O. I. *Integrals and Series. Volume 3: More special functions.* Gordon and Breach, 1989. 800 pages. First edition in Moscow, Nauka, 1986.

[8] SALVY, B., AND ZIMMERMANN, P. Gfun: a Maple package for the manipulation of generating and holonomic functions in one variable. Technical Report 143, Institut National de Recherche en Informatique et en Automatique, 1992. To appear in *ACM Transactions on Mathematical Software.*

[9] SINGER, M. F. Formal solutions of differential equations. *Journal of Symbolic Computation 10* (1990), 59–94.

[10] Stanley, R. P. Generating functions. In *Studies in Combinatorics, M.A.A. Studies in Mathematics, Vol. 17.* (1978), G.-C. Rota, Ed., The Mathematical Association of America, pp. 100–141.

[11] Zeilberger, D. A Maple program for proving hypergeometric identities. *SIGSAM Bulletin 25*, 3 (July 1991), 4–13.

[12] Zeilberger, D. The method of creative telescoping. *Journal of Symbolic Computation 11* (1991), 195–204.