Symbolic Computation with P-finite Sequences

Marko Petkovšek
University of Ljubljana, Slovenia

September 16, 1992

[summary by Bruno Salvy]

The talk consists in two parts. The first part is devoted to computer algebra algorithms for the manipulation of P-finite (holonomic in one variable) sequences and the second part, based on M. Petkovšek's thesis, to the resolution of linear recurrences with polynomial coefficients.

1. Manipulations of P-finite sequences

Given two sequences a_n and b_n defined by their linear recurrences and initial values, it is possible to build algorithmically the recurrences and initial values satisfied by αa_n, $a_n + b_n$, $a_n \cdot b_n$, $\sum_{k=0}^{n} a_k b_{n-k}$. From this, identities can be checked quite easily: to check that $a_n = b_n$ for all n, build up the equation satisfied by $a_n - b_n$, and check that the initial conditions are all zero. Note that the algorithms used for this purpose in the case of holonomic sequences (see the summary of P. Flajolet's talk on holonomic functions in last year seminar) can be replaced by faster and simpler ones in the case of C-finite sequences (solutions of recurrences with constant coefficients). These simpler algorithms have been implemented by W. Koepf [1][1]. An example of a C-finite identity which can be checked that way is Cassini's identity for Fibonacci numbers:

$$F_{n+1}F_{n-1} - F_n^2 = (-1)^n, \quad n \geq 1.$$

2. Resolution of linear difference equations

2.1. Classes of solutions. By "resolution" of an equation, one means finding an expression of a solution in a well-defined class of expressions. In general, not all solutions of an equation can be expressed in a class (e.g., not all solutions of polynomials in $\mathbb{Q}[x]$ are expressible in terms of radicals), and the problem is to find those solutions that admit such an expression, or to prove that none exists.

In the case of linear recurrence equations, Figure 1 shows the relationships between important classes. Here is a definition of the less well-known of them.

- Exponential polynomials are terms of the form
$$\sum_k e^{n\theta_k} P_k(n),$$
with P_k polynomials and the sum being finite. These are well-known to be the solutions of all C-finite recurrences.
- Quasirational terms are defined similarly, with rational functions instead of polynomials.
- Hypergeometric terms are functions $f(n)$ such that $f(n+1)/f(n)$ is rational in n. The general form of such a term is thus
$$C\frac{(\alpha_1+n)!\cdots(\alpha_p+n)!}{(\beta_1+n)!\cdots(\beta_q+n)!}Z^n,$$
with C in the ground field, and Z, α_i and β_j in its algebraic closure.

[1]The case with polynomial coefficients has been implemented in Maple by P. Zimmermann and B. Salvy in the gfun package, and of course by D. Zeilberger in a somewhat exotic Maple.

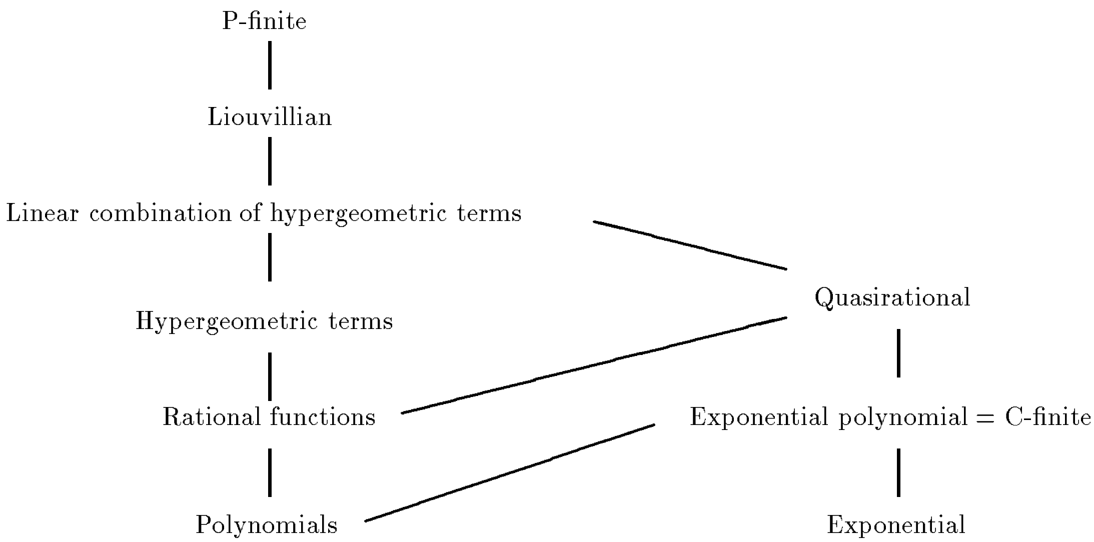

FIGURE 1. Classes of sequences

- Liouvillian terms in this context are terms built up by a finite number of products, sums, algebraic extensions and field operations.

From the algorithmic point of view, polynomial solutions can be found by indeterminate coefficients (the second coefficient yields the degree), rational and quasirational solutions can be found by S. Abramov's algorithm (see the summary of his talk in this seminar, and references there), solutions that are hypergeometric terms or linear combinations of them can be found thanks to M. Petkovšek's algorithm that we now describe.

2.2. Hypergeometric solutions. Let u_n be an hypergeometric solution of

$$p_d(n)u_{n+d} + p_{d-1}(n)u_{n+d-1} + \cdots + p_1(n)u_{n+1} + p_0(n)u_n = 0, \tag{1}$$

where the $p_i(n)$ are polynomials in n. Note that non-homogeneous equations can also be considered by increasing the order (a subtler solution will be presented later). Since u_n is hypergeometric, there exists a rational function $R(n)$ such that $u_{n+1} = R(n)u_n$. Substituting this into (1) and dividing out by u_n, we get a non-linear equation in R. The idea then is to consider poles of the $R(n+i)$, which have to be cancelled. A difficulty arises from the fact that a numerator of some other $R(n+j)$ might interfere in this cancellation. To make things clearer, Gosper introduced a slightly weaker form of the following decomposition lemma due to M. Petkovšek in this form.

LEMMA 1. *Let $\mathbb{F}$ be a field of characteristic 0 and $R \in \mathbb{F}(n)\backslash\{0\}$. Then there exists a unique decomposition*

$$R(n) = Z\frac{A(n)}{B(n)}\frac{C(n+1)}{C(n)},$$

where $Z \in \mathbb{F}$, A, B and C are monic polynomials with coefficients in $\mathbb{F}$ and

- $\gcd(A(n), B(n+k)) = 1$, *for all non-negative integer k,*
- $\gcd(A(n), C(n)) = \gcd(B(n), C(n+1)) = 1$.

Using this lemma and plugging this decomposition for u_{n+1}/u_n into (1), we get

$$\begin{aligned} &p_d(n)Z^dA(n+d)\cdots A(n)C(n+d) + p_{d-1}(n)Z^{d-1}A(n+d-1)\cdots A(n)B(n+d)C(n+d-1) \\ &\quad + \cdots + p_1(n)ZA(n)B(n+d)\cdots B(n+1)C(n+1) + p_0(n)B(n+d)\cdots B(n)C(n) = 0. \end{aligned} \tag{2}$$

Simple divisibility considerations then lead to

$$A(n) \mid p_0(n), \quad B(n+d) \mid p_d(n).$$

From this we deduce M. Petkovšek's algorithm HYPER:

(1) Compute the list of factors of $p_0(n)$ and $p_d(n-d)$ (over their splitting fields),
(2) for each pair of factors, compute equation (2),
(3) check the existence of a polynomial solution.

Note that the factorization in step 1 makes this algorithm expensive, and the loop in step 2 makes it exponential in the degrees of the leading and trailing coefficients of the recurrence.

2.3. Extensions.

A larger class. This algorithm can also be used to find non-hypergeometric solutions: once a solution $f_1(n)$ has been found, one can reduce the order of the recurrence, and call the algorithm recursively. If a solution $f_2(n)$ is then found, it corresponds to a solution $f_1(n)\sum f_2(n)$ of the initial equation. In general, this will not be hypergeometric. To check if it is, one can use either Gosper's or Abramov's algorithm to reduce the sum. Of course, the process can be repeated inductively.

Linear combinations. It is important to note that if $L(u_n)$ is a linear recurrence and h_n is an hypergeometric term (not necessarily a solution of L), then $L(h_n)/h_n$ is a rational function in n. Thus if $h_1,\dots,h_n$ are hypergeometric terms linearly independent over the field of rational functions, and if a linear combination of them is cancelled by L, then each h_i is a solution of L. Since these can be found by HYPER, we get that this algorithm cannot "miss" a linear combination of hypergeometric terms.

Non homogeneous equations. We now consider an equation of the form

$$Ly = f.$$

If f is a rational function, then an obvious solution is to make the equation homogeneous at the expense of increasing its order by one, and then apply the same algorithm. However, it follows from the considerations in the previous paragraph that there exists a particular polynomial solution in this case, and thus we just have to look for a polynomial.

If f is an hypergeometric term, then from the previous paragraph again, one has to look for a rational multiple of f. One then builds the equation this rational function has to satisfy, and appeal to Abramov's algorithm in order to get it.

When f is not an hypergeometric term and if a basis of solutions of $Ly = 0$ has been found either by HYPER or by its extension to multiple sums, then the following idea of S. Abramov's provides a nice alternative to the variation of constants method. Taking the solutions of the homogeneous equation in turn, one reduces the order of the equation and performs the same modification to the remaining solutions and to the right-hand side. At the end, we are left with an equation of order 0, hence the solution.

Definite hypergeometric summation. Consider

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

where $F(n+1,k)/F(n,k)$ and $F(n,k+1)/F(n,k)$ both are rational functions in n and k. Then Zeilberger has given an algorithm [6] that computes a linear recurrence for S_n. From this equation, HYPER will find if there is an hypergeometric sum.

Factorization of recurrence operators. The algorithm HYPER can be viewed as finding the right factors of order 1 or a recurrence operator. M. Petkovšek found another algorithm for the more general case of a right factor of any order. This algorithm is similar to F. Schwarz's algorithm in the differential case [5].

k-hypergeometric sequences. This is another natural extension of Hyper to sequences such that a_{n+k}/a_n is rational. Given k one can find all such sequences solutions of a linear recurrence equation. It is as yet unknown if there is a computable upper bound on the possible values of k for a given recurrence.

Bibliography

[1] Koepf (Wolfram). – Power series in computer algebra. *Journal of Symbolic Computation*, vol. 13, 1992, pp. 581–603.
[2] Petkovšek (Marko). – *Finding Closed-Form Solutions of Difference Equations by Symbolic Methods.* – PhD thesis, CMU, September 1990. Available as Technical Report CMU-CS-91-103.
[3] Petkovšek (Marko). – Factorization of linear difference operators. – Preprint, 1992.
[4] Petkovšek (Marko). – Hypergeometric solutions of linear recurrences with polynomial coefficients. *Journal of Symbolic Computation*, vol. 14, 1992, pp. 243–264.
[5] Schwarz (F.). – A factorization algorithm for linear ordinary differential equations. In *Proceedings of ISSAC'89*. pp. 17–25. – ACM Press, 1989.
[6] Zeilberger (Doron). – The method of creative telescoping. *Journal of Symbolic Computation*, vol. 11, 1991, pp. 195–204.