Polynomial Solutions of Linear Operator Equations

Marko Petkovšek
University of Ljubljana (Slovenia)

June 7, 1994

[summary by Frédéric Chyzak]

Abstract

The algorithm described here extends the algorithm to find all polynomial solutions of differential and difference equations that was given in [1, 2] to more general operators. It also takes a more efficient approach that avoids using undetermined coefficients. This summary is based on [4].

Let K be a field of characteristic 0 and $L : K[x] \to K[x]$ a K-linear endomorphism of $K[x]$. A new algorithm is presented in [4] that finds all polynomial solutions of homogeneous equations of the form $Ly = 0$, of nonhomogeneous equations of the form $Ly = f$ and of parametric nonhomogeneous equations of the form $Ly = \sum_{i=1}^{m} \lambda_i f_i$. The endomorphisms L under consideration in the following are polynomials in one of the following operators, and with coefficients in $K[x]$:

- the differential operator D defined by $Df(x) = df/dx$;
- the difference operator Δ defined by $\Delta f(x) = f(x+1) - f(x)$;
- the q-dilation operator Q used for q-difference equations and defined by $Qf(x) = f(qx)$. (In this case, $q \in K$, is not zero and not a root of unity.)

The interest of the new algorithm is twofold. First, numerous algorithms need to solve homogeneous, nonhomogeneous or parametric nonhomogeneous equations in $K[x]$ as subproblems. Examples are algorithms to find all rational, hyperexponential, geometric or Liouvillian solutions, to perform indefinite or definite hypergeometric summation, to factorize linear operators, etc. (See for instance [5, 3, 7, 6].) Second, the algorithm that is described here has lower complexity than the usual algorithms, that are often based on undetermined coefficients. The approach here is to find a degree bound on the solutions to be computed, and next find recurrences to compute the coefficients of the solutions efficiently. The problem with undetermined coefficients arises with very concise equations having high degree solutions. Although the number of coefficients to be determined is high, the recurrences that are found by the new algorithm in [4] are of small order.

The idea is to view the space $K[x]$ as a subspace of a unusual space of formal power series, and to embed the space of polynomial solutions into a space of formal power series solutions.

1. Algebraic setup

Let $(P_n(x))_{n\in\mathbb{N}}$ be a sequence of polynomials satisfying the following conditions:

(H1) $\deg P_n = n$, which makes $(P_n(x))_{n\in\mathbb{N}}$ a basis of $K[x]$;
(H2) $P_n \mid P_m$ as soon as $n < m$;

(H3) there exists $(A, B) \in \mathbb{Z}^2$, $A \leq 0$, $A \leq B$, and polynomials $\alpha_i \in K[n]$ such that for all n

$$LP_n = \sum_{i=A}^{B} \alpha_i(n) P_{n+i},$$

with α_A and α_B two non-zero polynomials, and $P_n = 0$ when $n < 0$.

Let $(l_n)_{n\in\mathbb{N}}$ be the dual basis of the $K[x]$-basis $(P_n(x))_{n\in\mathbb{N}}$. By definition, $l_n(P_m) = \delta_{n,m}$ and

$$P_n P_m = \sum_{k\in\mathbb{N}} l_k(P_n P_m) P_k.$$

As a consequence of (H2), computing modulo P_n and considering degrees yields $l_k(P_n P_m) = 0$ when $k < n$. Similarly, $l_k(P_n P_m) = 0$ when $k < m$. It follows that

$$l_k(P_n P_m) = 0 \qquad \text{when} \qquad k < \max(n, m) \text{ or } n + m < k.$$

The next step is to consider formal power series: let $S(x) = \sum_{n=0}^{\infty} c_n P_n(x)$ denote a formal series, and $K[[(P_n(x))_{n\in\mathbb{N}}]]$ the vector space of all such series. Let λ_n be the linear forms over $(P_n(x))_{n\in\mathbb{N}}$ such that $\lambda_n(S) = c_n$. Then $K[[(P_n(x))_{n\in\mathbb{N}}]]$ is a K-algebra for termwise sum and outer product, and for the following inner product

$$ST = \sum_{k=0}^{\infty} \left(\sum_{n,m \leq k \leq n+m} \lambda_n(S)\, \lambda_m(T)\, l_k(P_n P_m) \right) P_k.$$

Thanks to (H3), the operator L is extended to Λ on $K[[(P_n(x))_{n\in\mathbb{N}}]]$ by the following rule:

$$\Lambda S = \sum_{n=0}^{\infty} \left(\sum_{i=-B}^{-A} \alpha_{-i}(n+i)\, \lambda_{n+i}(S) \right) P_n.$$

Now, for any given $f \in K[[(P_n(x))_{n\in\mathbb{N}}]]$, $Ly = f$ is equivalent to

$$\sum_{i=-B}^{-A} \alpha_{-i}(n+i)\, \lambda_{n+i}(y) = l_n(f) \tag{1}$$

for all $n \in \mathbb{N}$ (with $\lambda_k = 0$ when $k < 0$).

The degree bound and the algorithm follow from the following theorems (see [4]).

THEOREM 1. *Let $Ly = f$ where y and f are polynomials. Then*

$$\deg y \leq N = \max\{-B - 1, \deg f - B, n \in \mathbb{Z} \text{ such that } \alpha_B(n) = 0\}.$$

THEOREM 2. *Let N be defined as in the previous theorem. Assume $N \geq 0$. Then, for any formal power series $y \in K[[(P_n(x))_{n\in\mathbb{N}}]]$, the following are equivalent:*

(1) *the formal power series y is a polynomial solution of $Ly = f$;*
(2) *the $\lambda_n(y)$'s satisfy equation (1) for $n \leq N + B$ and $\lambda_n(y) = 0$ for $n < N$.*

2. The algorithm

The goal is to find a basis for the affine space of vectors $g \in K^{N-A+B+1}$ that satisfy

$$\sum_{i=-B}^{-A} \alpha_{-i}(n+i)\, g_{n+i}(y) = l_n(f). \tag{2}$$

Of course, the direction of the affine solution space is given by vectors v that cancel the left hand-side. The vectors g and v represent polynomials in $K[x]$ denoted by

$$h(g,f) = \sum_{n=0}^{N-A+B} g_n P_n \qquad \text{and} \qquad s(v) = \sum_{n=0}^{N-A+B} v_n P_n \qquad \text{respectively.}$$

The set of singularities $\mathcal{S} = \{n \in \mathbb{N} \mid \alpha_A(n) = 0\}$ and the set $\mathcal{N} = \{0, \ldots, A-1\} \cup \mathcal{S}$ play an important rôle. The algorithm proceeds by iteratively computing the coefficients of the solutions. Each time a coefficient is not fully determined by equation (2) or its left hand-side, i.e. for each integer in $\mathcal{N}$, a new parameter is added, along with an equation to guarantee consistency between the elements of the basis under construction.

The algorithm maintains a list of vectors $\mathcal{V}$, a list of indeterminates $\mathcal{I}$, a list of equations $\mathcal{E}$ and an additional vector g. The vectors in $\mathcal{V}$ almost form a basis of the direction space of the affine solution set for the homogeneous equation. Indeed, the solutions are linear combinations of them ruled by the equations in $\mathcal{E}$. The vector g takes the nonhomogeneous part of the equation into account. The algorithm is the following:

(1) set $\mathcal{V}$, $\mathcal{I}$, $\mathcal{E}$ and g to empty lists or vectors;
(2) for each n from 0 to $N-A+B$, perform Step 3 when $n \notin \mathcal{N}$, or Step 4 when $n \in \mathcal{N}$, then go to Step 5;
(3) (*extension step*) extend all vectors $v \in \mathcal{V}$ (resp. g) by using the appropriate instance of equation (2) or its left hand-side;
(4) (*singularity step*) extend all vector $v \in \mathcal{V}$ and g by 0, then add the vector $[0, \ldots, 0, 1]$ (of length $n+1$) to $\mathcal{V}$, add the indeterminate c_n to the list $\mathcal{I}$, and finally add equation (2) for the index $n-A$ (when non-negative), where each g_t has been replaced by the sum cv_t over all pairs $(c, v) \in (\mathcal{I}, \mathcal{V})$ that have been added in a previous singularity step;
(5) let (c_k, v_k) be the pairs added into $(\mathcal{I}, \mathcal{V})$ during singularity steps and $\mathcal{E}(f)$ denote the final list of equations computed by the previous process; perform the following action, according to the type of equation being solved:
   - *homogeneous*: solve the system in the c_k's composed of the equations $\mathcal{E}(f)$ and the equations $\sum_k c_k l_n(s(v_k)) = 0$, for $N < n \leq N-A+B$, and return the general polynomial solution $y = \sum_k c_k s(v_k)$;
   - *nonhomogeneous*: solve the system in the c_k's composed of the equations $\mathcal{E}(f)$ and the equations $\sum_k c_k l_n(s(v_k)) = -l_n(h(g,f))$, for $N < n \leq N-A+B$, and return the general polynomial solution $y = \sum_k c_k s(v_k) + h(g,f)$;
   - *parametric nonhomogeneous*: solve the system in the c_k's and λ_i's composed of the equations $\mathcal{E}(\sum_{i=1}^m \lambda_i f_i)$ and the equations $\sum_k c_k l_n(s(v_k)) = -l_n\left(h\left(g, \sum_{i=1}^m \lambda_i f_i\right)\right)$, for $N < n \leq N-A+B$, and return the general polynomial solution $y = \sum_k c_k s(v_k) + h\left(g, \sum_{i=1}^m \lambda_i f_i\right)$.

3. Choice of the basis $(P_n(x))_{n \in \mathbb{N}}$

The previous algorithm ends by solving a linear system that consists of at most $\sigma - A + B$ equations in at most $\sigma - A$ (resp. $\sigma - A + m$ in the parametric case) variables, where σ is the number of

singularities between $-A$ and $N - A + B$. Therefore, avoiding singularities lessens the complexity. Let $L = \sum_{k=0}^{r} p_k(x)\,\partial^r$, where ∂ is either of D, Δ or Q, and let $d = \max\{j \mid \exists k \quad l_j(p_k) \neq 0\}$. The following choices for the P_n's satisfy conditions (H1–H3).

- *differential case*: $P_n = (x-a)^n/n!$, for a such that $p_r(a) \neq 0$; with this basis, no singularity can occur, $A = r$, $B = d$, and
$$\alpha_i(n) = \sum_{j=0}^{d} \binom{n+i}{j} l_j(p_{j-i});$$
- *difference case*: $P_n = \binom{x-a}{n}$, for $a > \max\{n \in \mathbb{N} \mid p_r(n) = 0\}$; with this basis, no singularity can occur, $A = r$, $B = d$, and
$$\alpha_i(n) = \sum_{k=0}^{r} \sum_{j=0}^{d} \binom{n+i}{j} \binom{j}{i+k} l_j(p_k);$$
- *q-difference case*: $P_n - x^n$; with this basis, singularity can occur, but $A = 0$, $B = d$, and
$$\alpha_i(n) = \sum_{k=0}^{r} q^{nk} l_i(p_k).$$

4. Formal power series solutions

In conclusion, it should be noted that the algorithm that has been discussed can also be used to compute (a description of) formal power series solutions in $(P_n(x))_{n\in\mathbb{N}}$. Indeed, let $M = \max(\mathcal{S})$. Then running the loop of the algorithm for $n = 0, \ldots, M$ and computing the $s(v)$ for $v \in \mathcal{V}$ yields the set of all polynomials $p \in K[x]$ of degree less than or equal to M such that $Lp(x) = f$ modulo $P_{M+1}(x)$. Iterating infinitely many times the extension step (by using equation (2)) yields a formal power series that is a solution.

Bibliography

[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.*, n° 3, 1989, pp. 63–68. – Transl. from Vestn. Moskov. univ. Ser. XV Vychisl. mat. kibernet. 3, 56–60.

[2] Abramov (S. A.). – Rational solutions of linear differential and difference equations with polynomial coefficients. *USSR Computational Mathematics and Mathematical Physics*, vol. 29, n° 11, 1989, pp. 1611–1620. – Translation of the Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki.

[3] Abramov (S. A.) and Petkovšek (M.). – Finding all q-hypergeometric solutions of q-difference equations. In Leclerc (B.) and Thibon (J. Y.) (editors), *Formal power series and algebraic combinatorics*. pp. 1–10. – Université de Marne-la-Vallée, 1995. Proceedings SFCA'95.

[4] Abramov (Sergei A.), Bronstein (Manuel), and Petkovšek (Marko). – *On Polynomial Solutions of Linear Operator Equations*. – Technical Report n° 33-468, Institute of Mathematics, Physics and Mechanics, University of Ljubljana. Slovenia., April 1995. 16 pages.

[5] Bronstein (M.) and Petkovšek (M.). – On Ore rings, linear operators and factorisation. *Programmirovanie*, n° 1, 1994, pp. 27–44. – Also available as Research Report 200, Informatik, ETH Zürich.

[6] Gosper (R. William). – Decision procedure for indefinite hypergeometric summation. *Proceedings of the National Academy of Sciences USA*, vol. 75, n° 1, January 1978, pp. 40–42.

[7] Singer (Michael F.). – Liouvillian solutions of linear differential equations with Liouvillian coefficients. *Journal of Symbolic Computation*, vol. 11, 1991, pp. 251–273.