ISSN 0361-7688, Programming and Computer Software, 2014, Vol. 40, No. 2, pp. 98–106. © Pleiades Publishing, Ltd., 2014.
Original Russian Text © S.A. Abramov, D.E. Khmelnov, 2014, published in Programmirovanie, 2014, Vol. 40, No. 2.

Regular Solutions of Linear Differential Systems with Power Series Coefficients

S. A. Abramov and D. E. Khmelnov

Computing Center, Russian Academy of Sciences, ul. Vavilova 40, Moscow, 119333 Russia
e-mail: sergeyabramov@mail.ru, dennis_khmelnov@mail.ru

Received October 17, 2013

Abstract—The following problem is considered: given a system of linear ordinary differential equations of arbitrary order with power series coefficients, to recognize whether it has regular solutions at point 0 and, if it does, to find them. An algorithm for solving this problem is proposed. Each power series that is a coefficient of the original system is specified by a procedure that computes the series coefficient by the index of this coefficient. The original system is assumed to have full rank; i.e., the equations of the system are independent. The algorithm is implemented in Maple.

DOI: 10.1134/S0361768814020029

1. INTRODUCTION

In [9], an algorithm for finding all regular solutions of a full-rank linear differential system of arbitrary order with polynomial coefficients was proposed (see also [1, Section 7.4]). In this paper, we rely on more general assumptions on the original differential system, supposing that all its coefficients are power series.

Let K be a number field: $\mathbb{Q} \subseteq K \subseteq \mathbb{C}$. The ring of polynomials and the field of rational functions of x over K are denoted as $K[x]$ and $K(x)$, respectively. The ring of formal power series of x over K is denoted as $K[[x]]$, and the field of formal Laurent series, as $K((x))$. For a nonzero element $a(x) = \sum a_i x^i$ from $K((x))$, its *valuation* $\mathrm{val}_x a(x)$ is defined as

$$\mathrm{val}_x a(x) = \min\{i : a_i \neq 0\}, \tag{1}$$

with $\mathrm{val}_x 0 = \infty$.Valuation of a vector or matrix with series entries is assumed to be equal to the minimum of valuations of the components.

If R is a ring (in particular, a field), then $\mathrm{Mat}_m(R)$ denotes the ring of square matrices of order m with entries from R. I_m denotes the identity matrix of order m, and M^T denotes the transpose of a matrix M.

It is convenient to write differential systems in terms of the operation $\theta = x\frac{d}{dx}$ rather than $\frac{d}{dx}$ (the transition from one notation to the other presents no difficulties). We consider systems of the form

$$A_r(x)\theta^r y + A_{r-1}(x)\theta^{r-1} y + \ldots + A_0(x) y = 0, \tag{2}$$

where $y = (y_1, y_2, \ldots, y_m)$ is a column of unknown functions of x. As for the coefficients

$$A_0(x), A_1(x), \ldots, A_r(x), \tag{3}$$

we assume that $A_i(x) \in \mathrm{Mat}_m(K[[x]])$, $i = 0, 1, \ldots, r$, with $A_r(x)$ (the *leading* matrix of the system) being nonzero. We consider systems the equations of which are independent over $K((x))[\theta]$. Such systems are called *full-rank* systems. For full-rank systems, we propose an algorithm for constructing their regular solutions, i.e., solutions of the form

$$y(x) = x^\lambda w(x), \tag{4}$$

where $\lambda \in \bar{K}$, $w(x) \in \bar{K}((x))^m[\ln x]$, and $\bar{K}$ is the algebraic closure of the field K. Each solution of this kind can be written as

$$x^\lambda \sum_{s=0}^{k} g_s(x) \ln^s x, \tag{5}$$

where $k \in \mathbb{N}$ (i.e., k is a nonnegative integer) and $g_s(x) \in \bar{K}((x))^m$, $s = 0, 1, \ldots, k$. If $\min_{s=0}^{k} \mathrm{val}_x g_s(x) = 0$, then λ is called an *exponent* of solution (4); otherwise, it is called an *exponent modulo* 1. If λ is an exponent modulo 1 for a regular solution $y(x)$, we will say that $y(x)$ *admits the factor* x^λ or that x^λ is an *admissible factor* of the solution $y(x)$. Clearly, if x^λ is an admissible factor of some nonzero regular solution, then $x^{\lambda'}$ is an admissible factor of this solution if and only if $\lambda - \lambda' \in \mathbb{Z}$. Transition from x^λ to $x^{\lambda'}$ in (5) changes valuations of the series $g_s(x)$.

The set

$$x^{\lambda_1}, x^{\lambda_2}, \ldots, x^{\lambda_s} \tag{6}$$

is called a *complete* set of admissible factors of regular solutions of system S if (i) no exponents of the elements of set (6) differ by an integer; (ii) each element x^{λ_i} of set (6) is an admissible factor for some nonzero regular solution of system S; (iii) for every nonzero regular solution of system S, set (6) contains an admissible factor for this solution.

All regular solutions of a given system that admit one and the same factor form a linear space over $\mathbb{C}$. Our algorithm finds admissible factors (up to integer addends in the exponents) for a given system and constructs bases of the corresponding solution spaces. To make problem statement more specific, it is required to specify the way the infinite series in the original system are given and the form the regular solutions are output by the algorithm. As for the input data, we rely on a quite universal approach that assumes that each power series that is a coefficient of the system is given algorithmically: for any entry $a(x) = \sum_{i=0}^{\infty} a_i x^i$ of a matrix from (3), there is an algorithm Ξ_a that, for $i \in \mathbb{N}$, calculates the value

$$\Xi_a(i) = a_i. \qquad (7)$$

It follows from [5, Proposition 2] that, for the given representation of system coefficients, both verification of independence of the equations over $K[[x]][\theta]$ and verification of solution existence in $K((x))^m \backslash \{0\}$ (i.e., existence of a nonzero Laurent solution) are algorithmically undecidable problems. It should be noted that, as can easily be seen, the Laurent solutions are the simplest regular solutions. However, the original system is assumed to have full rank (i.e., the equations of the system are independent over $K[[x]][\theta]$). For this case, an algorithm for constructing the Laurent solutions is presented in [4]. This algorithm will play an important role in the subsequent discussion, which is further addressed in Section 2.

Here, we note that, from the standpoint of the subsequent discussion, it does not matter whether Ξ_a is a real algorithm, whose description is available, or just a black box. Not all conceivable series can be represented algorithmically; our algorithm works with other series as well. In what follows, we will speak of algorithmic series representation having in mind that the actual situation is as was outlined above.

Let us clarify the representation of regular solutions output by the algorithm. We will start from the concept that is very important in the following discussion. Let $l \in \mathbb{Z} \cup \{-\infty\}$ and $a(x) \in \bar{K}((x))$. We define the *l-truncation* $a^{\langle l \rangle}(x)$ of the series $a(x)$ to be the result of zeroing out all coefficients of powers that are greater than or equal to l; if $l = -\infty$, then $a^{\langle l \rangle}(x) = 0$. Thus, $a^{\langle l \rangle}(x)$ is always a Laurent polynomial, i.e., an element of the ring $\bar{K}[x, x^{-1}]$. Similarly, the *l-truncation of a system* is the system obtained from the original one by the replacement of all series containing in its coefficients by their l-truncations. Let $W_S(\lambda)$ denote the space of the regular solutions of system S that admit the factor x^λ, and let $W_S^{\langle l \rangle}(\lambda)$ denote the space obtained from $W_S(\lambda)$ by the replacement of each element of form (5) with

$$x^\lambda \sum_{s=0}^{k} g_s^{\langle l \rangle}(x) \ln^s x$$

(thus, $W_S(\lambda)$ coincides with $W_S(\lambda + n)$ for any $n \in \mathbb{Z}$; however, this cannot be guaranteed for $W_S^{\langle l \rangle}(\lambda)$ and $W_S^{\langle l \rangle}(\lambda + n)$). Since the space $W_S(\lambda)$ is finite-dimensional, the valuations of the series $g_s(x)$ and $g_s^{\langle l \rangle}(x)$ are clearly bounded from below. In this case, for all sufficiently large l, we have $\dim W_S(\lambda) = \dim W_S^{\langle l \rangle}(\lambda)$.

Now, let us formulate the problem of construction of regular solutions that is solved by the algorithm proposed in the paper. We call it the $\mathbf{P}_R$ problem. It is assumed that a full-rank system S of form (2) and $d \in \mathbb{N}$ are given.

$\mathbf{P}_R$: Find a complete set of admissible factors of nonzero solutions of system S. Determine $l_0 \in \mathbb{Z}$ such that, for each x^λ from this set, $W_S(\lambda) = \dim W_S^{\langle l \rangle}(\lambda)$ for all $l \geq l_0$ and find a basis of the space $W_S^{\langle l_0 + d \rangle}(\lambda)$.

It makes sense to recall that, in the scalar case, the problem of finding regular solutions can be solved by means of algorithms known from the theory of differential equations. The Frobenius algorithm relies on studying roots of the indicial equation [2, Ch. IV; 16; 19, Ch. V]. When constructing solutions by this algorithm, not only values of the roots of the indicial equations, but also their multiplicities, are taken into account, as well as the existence of roots differing from one another by integers. The Heffter algorithm [17, Chs. II and VIII; 19, Ch. V] constructs a (possibly, empty) basis of regular solutions that admit the factor x^λ not taking into account the multiplicity of the root λ and the possibility of existence of other roots differing from λ by an integer. Owing to this, the Heffter algorithm turns out more convenient from the standpoint of generalization to systems of arbitrary order: the algebraic equation—an analogue of the indicial equation in the scalar case—one manages to construct for a differential system may contain, for example, redundant roots that bear no information on the space of system solutions.

In [14], the Heffter algorithm was extended to the first-order systems $y'(x) = A(x)y(x)$. The generalization of the Heffter algorithm to the case of abitrary-order systems with polynomial coefficinets was proposed in [9]. If the leading matrix of the original differential system is nonsingular, then one can apply the algorithm from [13] to systems of arbitrary order, which is based on a different approach. Note that, although the coefficients are assumed to be infinite series, the method of representation of such series is not discussed in [13, 14] (in the examples considered in these publications, the coefficients are, basically, rational functions). It is assumed in those works that, for each series, one can check whether it is equal to zero (independent of whether this series is predefined or obtained as a result of some operations on other series).

In this paper, we extend the Heffter approach to the arbitrary-order systems with the coefficients given by algorithmically specified series. In Section 4, an algorithm for solving problem $\mathbf{P}_R$ is proposed. Its implementation in Maple [20] is discussed in Section 5.

2. CONSTRUCTION OF LAURENT SOLUTIONS

2.1. Truncated Laurent Solutions

Let V_S denote the space of the Laurent solutions of a system S and $V_S^{\langle l \rangle}$, $l \in \mathbb{Z}$, be the space whose elements are l-truncations of the corresponding elements of the space V_S. The algorithm from [4] solves the $\mathbf{P}_L$ problem, which is stated below. It is assumed that a full-rank system S of form (2) and $d \in \mathbb{N}$ are given.

$\mathbf{P}_L$: Determine $l_0 \in \mathbb{Z}$ such that $\dim V_S = \dim V_S^{\langle l \rangle}$ for all $l \geq l_0$ and find a basis of the space $V_S^{\langle l_0 + d \rangle}$.

The algorithm for solving the $\mathbf{P}_L$ problem is based on the consideration of a recurrence system for a sequence of coefficients of arbitrary Laurent solutions of the original differential system. These recurrence systems are discussed in Section 2.2. The recurrence systems are reduced to a "convenient" form by the EG algorithm [1, 3, 6, 7] (more precisely, by a special version of this algorithm [4], which is outlined in Section 2.3).

2.2. Sequence of Coefficients of Laurent Solution

We will use notation E for the *shift operator*. The application of this operator to a two-sided sequence $a(n)$ yields the two-sided sequence $b(n) = a(n + 1)$, $n \in \mathbb{Z}$.

The transformation

$$x \longrightarrow E^{-1}, \quad x^{-1} \longrightarrow E, \quad \theta \longrightarrow n \tag{8}$$

determines the isomorphism

$$\mathcal{M}: \mathcal{D}_m \longrightarrow \mathcal{E}_m \tag{9}$$

of the ring of the differential operators $\mathcal{D}_m = \mathrm{Mat}_m(K((x)))[\theta]$ onto the ring of the recurrence operators $\mathcal{E}_m = \mathrm{Mat}_m(K[n])((E^{-1}))$ and converts the original differential system S to the induced recurrence system

$$B_0(n)z(n) + B_{-1}(n)E^{-1}z(n) + \ldots = 0, \tag{10}$$

which can also be written as

$$B_0(n)z(n) + B_{-1}(n)z(n-1) + \ldots = 0,$$

where

- $z(n) = (z_1(n), \ldots, z_m(n))^T$ is a column of unknown sequences such that $z_i(n) = 0$ for all negative integers n for which $|n|$ is big enough, $i = 1, 2, \ldots, m$;
- $B_0(n), B_{-1}(n), \ldots \in \mathrm{Mat}_m(K[n])$, with the matrix entries being polynomials of degree r or less;
- $B_0(n)$ is a nonzero matrix (the leading matrix of system (10)).

In the scalar case, such a transformation yielding the recurrence relation was considered in [8, 11, 12]; for the case of systems, it was considered in [4].

The induced system R obtained by the application of $\mathcal{M}$ is of full rank (i.e., the equations of system (10) are independent over $K(n)[[E^{-1}]]$) if and only if the original differential system S is of full rank [4]. The system S has a Laurent solution $y(x) = z(\nu)x^\nu + z(\nu + 1)x^{\nu+1} + \ldots$ if and only if the two-sided sequence

$$\ldots, 0, 0, z(\nu), z(\nu+1), \ldots$$

of vector coefficients satisfies the induced recurrence system R of form (10), i.e., the following equalities hold:

$$B_0(\nu)z(\nu) = 0,$$

$$B_0(\nu+1)z(\nu+1) + B_{-1}(\nu+1)z(\nu) = 0,$$

$$B_0(\nu+2)z(\nu+2) + B_{-1}(\nu+2)z(\nu+1) + B_{-2}(\nu+2)z(\nu) = 0,$$

$$\ldots$$

If the matrix $B_0(n)$ is nonsingular, then the set of roots of its determinant yields a finite superset of the set of valuations of all Laurent solutions of system S. However, in many cases, this matrix is singular, even when the leading matrix $A_r(x)$ of system S is nonsingular.

2.3. EG Algorithm

Let us discuss the basic idea of the special version of the EG algorithm suggested in [4] (in what follows, it will be referred to as simply the EG algorithm). Its purpose is to convert system (10) to a recurrence system with a nonsingular leading matrix.

Along with transforming the induced system itself, we will modify vector $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_m)$ with nonnegative integer components, which is initially set equal to $\gamma = (r, r, \ldots, r)$.

The "reduction + shift" step of the transformation of the recurrence system consists of the following three substeps:

(a) If the rows of the leading matrix are linearly dependent over $K(n)$ with the coefficients

$$v_1(n), v_2(n), \ldots, v_m(n) \in K[n], \tag{11}$$

then set

$$\mu = \max_{\substack{0 \leq j \leq m \\ v_j(n) \neq 0}} (\gamma_j + \deg v_j(n)).$$

Choose i such that

$$0 \leq i \leq m, \quad v_i(n) \neq 0, \quad \gamma_i + \deg v_i(n) = \mu \tag{12}$$

and replace the ith equation of the induced recurrence system by the linear combination of all equations of the system with the coefficients $v_1(n), v_2(n), \ldots, v_m(n)$. (As a result, the ith row of the leading matrix vanishes. This stage is called *reduction*.)

(b) Apply operator E to the ith equation of the system obtained after the reduction. (This stage is called *shift*.)

(c) Increase γ_i by deg $v_i(n)$, i.e., perform $\gamma_i := \mu$.

The repetition of the steps "reduction + shift" will never result in the equation with the zero left-hand side, i.e., to the equation $0 = 0$, since the equations of the system are assumed to be independent over $K(n)[[E^{-1}]]$. In [4], it is proved that this process terminates: at a certain step, the rows of the leading matrix turned out independent over $K(n)$.

The reduction substep can generate a set of *linear constraints* due to multiplications of the equations transformed by polynomials having integer roots. Let us assume that the ith equation is replaced by the linear combination of all equations of the system with the coefficients $v_1(n)$, $v_2(n)$, ..., $v_m(n)$, and let n_0 be the root $v_i(n)$. For any solution $y(x) = \sum_{n=\nu}^{\infty} z(n)x^n$, $\nu \le n_0$, the original system must satisfy the linear constraint

$$\begin{aligned}&[B_0(n_0)]_{i,*}\, z(n_0) + [B_{-1}(n_0)]_{i,*}\, z(n_0-1) + \\ &\quad \ldots + [B_{-n_0+\nu}(n_0)]_{i,*}\, z(\nu) = 0,\end{aligned} \tag{13}$$

where we used notation

$$[M]_{i,*}, \quad 1 \le i \le m,$$

for the $(1 \times m)$-matrix that is the ith row of an $(m \times m)$-matrix M.

The number n_0 in (13) is referred to as the *index* of this linear constraint.

Like in the case of systems with polynomial coefficients [1, Section 8.1], the algorithm of transformation of the induced recurrence system can be adapted for work with inhomogeneous systems. Let the original system for the solution of which the induced recurrence system is constructed have the form

$$A_r(x)\theta^r y + A_{r-1}(x)\theta^{r-1}y + \ldots + A_0(x)y = b(x),$$

with the left-hand side of the equation coinciding with the left-hand side of system (2) and the right-hand side being the vector with the components in the form of the Laurent series

$$b(x) = \sum_{n=\nu}^{\infty} r(n)x^n,$$

where ν is the valuation of the right-hand side and $r(n)$, $n \in \mathbb{Z}$, are vectors of the coefficients of the Laurent series. Then, the right-hand side of the corresponding induced recurrence system is equal to $r(n)$ (for $n < \nu$, we set $r(n) = 0$). Upon execution of the "reduction + shift" steps, the components of the right-hand side do not leave their places but take part in the reduction stage and are subjected to the action of the operator E at the shift stage. When constructing regular solutions of a full-rank linear differential system of arbitrary order with polynomial coefficients in [9], it was required to solve induced recurrence systems that differed from one another only by the right-hand sides. In order that all systems of this kind could be solved by a single application of the EG algorithm, the right-hand side was represented as a vector $r(n) = (r_1(n), r_2(n), \ldots, r_m(n))^T$, with the components being undefined functions $r_i(n)$, $i = 1, 2, \ldots, m$. A similar approach can be used in the case under consideration. When using the right-hand side of the generic form, each component of the transformed right-hand side is a linear combination of the (possibly, shifted) components of the original right-hand side. Note that the maximum shift of the components on the right-hand side is equal to the maximum number of shifts of one and the same equation in the course of the EG algorithm operation. Denoting this number as ξ, we obtain the following expressions for the components of the transformed right-hand side:

$$\tilde{r}_i(n) = \sum_{j=i}^{m}\sum_{k=0}^{\xi} \alpha_{ijk}(n) r_j(n+k), \tag{14}$$

$i = 1, 2, \ldots, m$, where $\alpha_{ijk}(n) \in K[n]$ are the coefficients corresponding to the transformations performed in the course of the execution of the "reduction + shift" steps (these coefficients can be expressed in terms of coefficients (11) obtained in the course of all reduction stages). Thus, one can use one and the same transformed recurrence systems for solving all systems with the same left-hand side and different right-hand sides. To this end, it is required to substitute components of the original right-hand side of the particular system to the transformed right-hand side in the form of particular values $r_j(n)$, $j = 1, 2, \ldots, m$. We will further use such a kind of the EG algorithm for solving the $\mathbf{P}_R$ problem. The arising linear constraints will also be inhomogeneous: the right-hand side of (13) becomes nonzero. As before, the index of this linear constraint is the number n_0. As in the case of systems with polynomial coefficients [1, Section 7.4], when searching regular solutions, it is not sufficient to use linear constraints with only integer indices (in this case, to calculate linear constraints (13) when eliminating the ith row on a reduction step, all, rather than only integer, roots of the coefficient $v_i(n)$ from (11) are taken). The set of all indices obtained is denoted by N.

2.4. Algorithm for Solving the $\mathbf{P}_L$ Problem

The algorithm for solving the $\mathbf{P}_L$ problem is based on the execution of the "reduction + shift" steps described in Section 2.3. However, system (10) to be transformed is infinite. The algorithm cannot work with all matrices B_{-t}, $t = 0, 1, \ldots$, simultaneously, and this is the point where lazy calculations storing information about all already performed reductions and shifts come to help. This allows us, if needed, to use matrices B_{-t} with the increasing values of t not repeating the already performed operations on matrices with lesser values of t.

Analyzing the roots of the determinant of the nonsingular leading matrix constructed and taking into account the linear constraints found, we can solve the $\mathbf{P}_L$ problem. Let e^* and e_* be maximal and minimal integer roots of this determinant, respectively (it may happen that $e^* = e_*$). Let N_Z be the set of all integer values from N. Then, the desired value to be determined in the $\mathbf{P}_L$ problem may be taken equal to

$$l_0 = \max(N_Z \cup \{e^*\}). \qquad (15)$$

If

$$l_1 = l_0 + d + \xi - e_*, \qquad (16)$$

then solution of the $\mathbf{P}_L$ problem for the original homogeneous system S coincides with the solution for the truncated system $S^{\langle l_1 \rangle}$. The details can be found in [4].

3. SEARCH FOR REGULAR SOLUTIONS BASED ON THE HEFFTER APPROACH: GENERAL SCHEME

In what follows, we write system (2) in the form $L(y) = 0$, where L is the differential operator

$$L = A_r(x)\theta^r + A_{r-1}(x)\theta^{r-1} + \ldots + A_0(x). \qquad (17)$$

For any integer $i \geq 0$, the result of application of L to $g(x)\ln^i(x)/i!$ has the form

$$L_{ii}(g)\frac{\ln^i x}{i!} + \ldots + L_{i1}(g)\frac{\ln x}{1!} + L_{i0}(g),$$

where $L_{i0}, L_{i1}, \ldots, L_{ii} \in \mathrm{Mat}_m(K[[x]])[\theta]$, $L_{00} = L$, and $L_{i+j,j} = L_{i0}$ for all $i, j \geq 0$ [17; 18, Section 3.2.1]. We will use the notation $L_i = L_{i0}$ $(= L_{i+j,j}$ for all $j \geq 0)$.

Proposition 1. For all $i \geq 0$, the following equality holds:

$$L_i = \sum_{k=i}^{r} A_k(x)\binom{k}{i}\theta^{k-i}, \qquad (18)$$

where $\binom{k}{i}$ are binomial coefficients.

Proof. The proposition follows from the Leibniz formula for the differentiation of the product of two functions and from the equality

$$\theta^k \frac{\ln^i x}{i!} = \begin{cases} \dfrac{\ln^{i-k} x}{(i-k)!} & \text{if } k \leq i, \\ 0 & \text{if } k > i. \end{cases}$$

□

From formula (18), it follows that $L_i = 0$ for all $i > r$.

The general scheme of finding regular solutions of the systems under consideration is similar to the scheme [9] used in the algorithm for finding all regular solutions of a full-rank linear differential system of an arbitrary order with polynomial coefficients (see also [1, Section 7.4]). This scheme itself is a generalization of the Heffter algorithm [17] and is based on the consideration of the sequence of systems

$$S_0, S_1, \ldots, \qquad (19)$$

where S_k is the system

$$L_0(g_i) = -\sum_{j=1}^{i} L_j(g_{i-j}), \quad i = 0, 1, \ldots, k \qquad (20)$$

(for $i = 0$ in (20), we have $L_0(g_0) = 0$). For a particular i, the system

$$L_0(g_i) = -\sum_{j=1}^{i} L_j(g_{i-j})$$

will also be referred to as the subsystem $\hat{S}_i$. Thus, the system S_k from sequence (19) consists of the subsystems

$$\hat{S}_0, \hat{S}_1, \ldots, \hat{S}_k,$$

and search for a solution to system S_{k+1} reduces to searching for a solution to subsystem $\hat{S}_{k+1}$ with regard to the earlier found solution $(g_0(x)^T, \ldots, g_k(x)^T)^T$ of system S_k. The following theorem is an analogue of the assertion proved by Heffter for the scalar case.

Theorem 1 ([9, 10]). *The set of nonnegative integer k for which system S_k has a Laurent solution*

$$(g_0(x)^T, g_1(x)^T, \ldots, g_k(x)^T)^T, \quad g_0(x) \neq 0,$$

is finite. If it is empty, then the equation $L(y) = 0$ has no nonzero solutions in $K((x))^m[\ln x]$. If this set is not empty and $\tilde{k}$ is its maximal element, then any solution of system $L(y) = 0$ belonging to $K((x))^m[\ln x]$ has the form

$$\sum_{s=0}^{\tilde{k}} g_{\tilde{k}-s}(x)\frac{\ln^s x}{s!}, \qquad (21)$$

where

$$(g_0(x)^T, g_1(x)^T, \ldots, g_{\tilde{k}}(x)^T)^T, \quad g_0(x) \neq 0, \qquad (22)$$

is a Laurent solution of system $S_{\tilde{k}}$. At the same time, any Laurent solution of system $S_{\tilde{k}}$ of form (22) *generates solution* (21) *to system $L(y) = 0$.*

In [9, 10], this theorem was proved for the case of systems with polynomial coefficients. Since the proof did not rely on the form of the coefficients, the same proof is applicable to the case of the series coefficients.

If the value of λ is known, substitution (4) reduces search for a regular solution to searching for solution $w(x) \in K((x))^m[\ln x]$. For the candidates on the role of λ, roots of the determinant of the nonsingular leading matrix of the induced recurrence system are used (after the application of the transformation described in Section 2.3 if the leading matrix of the induced recurrence system was originally singular).

Thus, we arrive at the following scheme.

1. For a given system S of form (2) with operator (17), construct the induced recurrence system and, by means of the EG algorithm described in Section 2.3, transform it to an equivalent system with a nonsingular leading matrix $\tilde{B}_0(n)$. Calculate all roots of the equation $\det\tilde{B}_0(n) = 0$. Assuming that two roots λ and λ' are equivalent if $\lambda - \lambda' \in \mathbb{Z}$, construct set Λ containing one representative from each equivalence class.

2. For every $\lambda \in \Lambda$, find regular solutions admitting the factor x^λ:

(a) Construct system $S(\lambda)$ by means of substitution (4) and subsequent multiplication by $x^{-\lambda}$.

(b) Construct Laurent solutions for the systems from (19) until the first system that has no Laurent solutions is met. This yields regular solutions $y(x)$ in form (21) for system $S(\lambda)$.

4. DETAILING THE HEFFTER SCHEME

For a fixed λ, the scheme from Section 3 reduces the problem of finding regular solutions to that of finding the Laurent solutions, for which the algorithm of its solution is known. The goal of this section is to elaborate details and to study possibilities of coordinated consideration of all values of λ belonging to Λ. This coordination improves efficiency of the algorithm.

4.1. Substitutions Performed

The left-hand sides of the inhomogeneous systems solved on step 2(b) of the scheme coincide with each other and are equal to the left-hand side of system $S(\lambda)$ obtained on step 2(a) from the original system S by means of substitution (4) and subsequent multiplication by $x^{-\lambda}$. Let the induced system R for the original system S transformed by the EG algorithm from Section 2.3 have the form

$$\tilde{B}_0(n)z(n) + \tilde{B}_1(n)z(n-1) + \ldots = \tilde{r}(n), \qquad (23)$$

and let N be the set of indices n_0 of the arising linear constraints. It is not difficult to see that the induced system $R(\lambda)$ for the system $S(\lambda)$ can be converted by the same transformations to the recurrence system

$$\begin{aligned}\tilde{B}_0(n+\lambda)z(n) + \tilde{B}_1(n+\lambda)z(n-1) + \ldots \\ = \tilde{r}(n+\lambda).\end{aligned} \qquad (24)$$

The corresponding linear constraints and the set of indices $N(\lambda)$ for (24) can be obtained by considering linear constraints arising upon transformation of system R.

Let Λ be the set determined on step 1 of the scheme from Section 3. For any $\lambda \in \Lambda$, the maximal and minimal integer roots of the equation

$$\det\tilde{B}_0(n+\lambda) = 0 \qquad (25)$$

are denoted as $e^*(\lambda)$ and $e_*(\lambda)$, respectively. If this equation has no integer roots, then the original differential system has no regular solutions admitting factor x^λ for the considered value of λ. In this case, the set Λ is modified by eliminating this value from the set (further, we assume that, for any $\lambda \in \Lambda$, equations (25) have integer roots). For $\lambda \in \Lambda$, we set

$$l_0(\lambda) = \max(N_Z(\lambda) \cup \{e^*(\lambda)\}),$$

where $N_Z(\lambda)$ contains all integer indices from $N(\lambda)$.

Proposition 2. Let Λ be the set determined on step 1 of the Heffter scheme. Let $\Lambda \neq \varnothing$ and, for any $\lambda \in \Lambda$, equation (25) have integer roots. Then, the desired value to be determined in the $\mathbf{P}_R$ problem can be taken as

$$l_0 = \max_{\lambda \in \Lambda} l_0(\lambda). \qquad (26)$$

Proof. According to (15), for a fixed λ for system $S(\lambda)$, we may take l_0 presenting in the statement of problem $\mathbf{P}_L$ equal to $l_0(\lambda)$. In this case, all subsystems $\hat{S}_i(\lambda)$ have left-hand sides coinciding with the left-hand side of the system $S(\lambda)$. □

4.2. Inhomogeneous Systems

Solution $g_0(x)$ of the subsystem $\hat{S}_0$, i.e., subsystem $L_0(g_0) = 0$, contains arbitrary constants. We use $g_0(z)$ for the calculation of the right-hand side of the subsystem $\hat{S}_1$, i.e., the subsystem $L_0(g_1) = -L_1(g_0)$; the above-mentioned arbitrary constants occur in the right-hand side linearly. Applying the same technique, as in the case of the scalar equation with a parameterized right-hand side (see, for example, [8]), we find, along with $g_1(x)$, linear relations for the constants occurring in $g_0(x)$ and $g_1(x)$. Continuing this process, on each step for the current subsystem, we obtain $\hat{S}_i$, $g_0(x)$, ..., $g_{i-1}(x)$, which contain unknown constants, and a linear algebraic system for these constants. In accordance with Theorem 1, condition $g_0(x) \neq 0$ guarantees that this process terminates.

All these subsystems for the given λ have one and the same left-hand side. Therefore, their induced recurrence systems have the same left-hand sides, but their right-hand sides are different. In order to transform the induced system only once, these transformations are applied to the induced system with the generic right-hand side, as described in Section 2.3.

Theorem 2. *Let ξ be the greatest number of shifts of one equation in the course of application of the EG algorithm to the induced recurrence system, $d \in \mathbb{N}$, and let l_0 and $e_*(\lambda)$ be defined like in Proposition 2. Then, to find $(l_0 + d)$-truncations of the Laurent solution of the subsystem $\hat{S}_k$, it is sufficient to calculate all solutions $g_j(x)$ of the preceding subsystems $\hat{S}_j$, $j = 0, \ldots, k-1$, as t_{kj}-truncations with*

$$t_{kj} = l_0 + d + (k-j)\xi. \qquad (27)$$

In this case, the left-hand side of the subsystem $\hat{S}_k(\lambda)$ *can be taken in the form of the* l_1*-truncation with*

$$l_1 = \max_{\lambda \in \Lambda}(l_0 + d + \xi - e_*(\lambda)), \quad (28)$$

and the left-hand side of the preceding subsystems $\hat{S}_j$, $j = 0, \ldots, k-1$, *as the* l_{1kj}*-truncation with*

$$l_{1kj} = l_1 + (k-j)\xi. \quad (29)$$

Proof. According to (16), if $l_1(\lambda) = l_0(\lambda) + d + \xi - e_*(\lambda)$, then solution of the $\mathbf{P}_L$ problem for the original homogeneous system $S(\lambda)$ coincides with the solution for the truncated system $S(\lambda)^{\langle l_1(\lambda)\rangle}$. The desired $(l_0 + d)$-truncation of the corresponding Laurent solution is constructed by means of the transformed induced recurrence system, and, to this end, its equations for the values of n from $e_*(\lambda)$ to $l_0 + d$ are used. Accordingly, in the case of the subsystem $\hat{S}_k(\lambda)$, for construction of the $(l_0 + d)$-truncation of the corresponding Laurent solution, coefficients of the right-hand side of the transformed induced recurrence system up to the degree $l_0 + d$ are required. Components of the right-hand side can contain up to ξ results of shifts of the components of the right-hand side of the original system. Accordingly, the truncated right-hand side of the original subsystem can be considered up to the degree $l_0 + d + \xi$. Then, it follows that the solution $g_{k-1}(x)$ of the previous subsystem $\hat{S}_{k-1}(\lambda)$ may be calculated up to the degree $l_0 + d + \xi$. Increasing d by ξ and repeating the reasoning, we find that the solution $g_{k-2}(x)$ of the subsystem $\hat{S}_{k-2}(\lambda)$ may be calculated up to the degree $l_0 + d + 2\xi$. Continuing the process, we find that it is sufficient to find solution $g_j(x)$ of the subsystem $\hat{S}_j(\lambda)$ in the form of the t_{kj}-truncation, where $t_{kj} = l_0 + d + (k-j)\xi$. Note that, for any $\lambda \in \Lambda$, in the search for the $(l_0 + d)$-truncation of the solution of $\hat{S}_k(\lambda)$, one can use an l_1-truncation $\hat{S}_k(\lambda)$ with $l_1 = \max_{\lambda \in \Lambda} l_1(\lambda)$. Accordingly, when searching for t_{kj}-truncations of the solutions of the preceding subsystems $\hat{S}_j$, $j = 0, \ldots, k-1$, one can use the l_{1kj}-truncation $\hat{S}_j(\lambda)$ with $l_{1kj} = l_1 + (k-j)\xi$. □

4.3. Algorithm

Having found the set Λ by means of the EG algorithm and the values $e_*(\lambda)$, $e^*(\lambda)$, $l_0(\lambda)$ for each $\lambda \in \Lambda$, we can determine l_0 by formula (26). Note that the elements of the set Λ are roots of algebraic equations, and it is assumed that the exact representations for these roots are available (in Maple, with the help of `RootOf`). In the course of application of the EG algorithm, the value of ξ is also computed: for each equation, the number of its shifts is counted, and, then, the maximal value among them is taken. These calculations correspond to step 1 of the Heffter scheme. It should be emphasized that, on this step, the EG algorithm is applied to the induced recurrence system with the right-hand side of the generic form, as discussed in Section 2.3.

The subsequent application of this scheme (steps 2(a) and 2(b)) is based on searching for truncated Laurent solutions for the truncated systems by means of the transformed inhomogeneous induced recurrence systems (an analogue of the algorithm from [4]). Formula (24) allows us not to perform all steps of the EG algorithm repeatedly for each system. On step 2(b), solution of the current system S_k reduces to searching the $(l_0 + d)$-truncation of the Laurent solution of the additional subsystem $\hat{S}_k$, which, in turn, requires calculation of the t_{kj}-truncations of the Laurent solutions of the preceding subsystems $\hat{S}_j$, $j = 1, \ldots, k-1$. The earlier found $t_{k-1,j}$-truncations can be prolonged since $t_{k-1,j} < t_{kj}$. This, in turn, requires prolongation of the truncations of the corresponding subsystems. In order that the coefficients of the truncated systems contain the desired number of terms, it will suffice to follow formulas (27)–(29).

5. IMPLEMENTATION

The algorithm is implemented in the computer algebra system `Maple` in the framework of package `EG` [1]. The implementation is partially based on the implementation of the algorithm for finding regular solutions of a full-rank linear differential system with polynomial coefficients [9] and on the implementation of the algorithms for solving the $\mathbf{P}_L$ problem.

For a given differential system of full rank with the coefficients in the form of the Laurent series, procedure `RegularSolution` finds solution to the $\mathbf{P}_R$ [4] problem.

The system $L(y) = 0$ is specified by the operator L represented in the matrix form as

$$\begin{pmatrix} L_{11} & \ldots & L_{1m} \\ \ldots & \ldots & \ldots \\ L_{m1} & \ldots & L_{mm} \end{pmatrix}, \quad (30)$$

with the operators L_{ij} being considered as power series in x with the coefficients from $K[\theta]$. The order of such a coefficient does not exceed the order of the system $L(y) = 0$. For fixed values of indices i and j, the operator L_{ij} is given by a function of an integer argument, for example, k, which computes the coefficient of x^k in this operator as a polynomial of θ. For all pairs of indices i, j, these functions can be defined by procedures. In simple cases, functions `if` and `piecewise` are used (see the example below). Besides the system given in this way, the procedures have three additional parameters: θ is the name of the operator $x\frac{d}{dx}$, x is the name of variable, and d is the value specified in the $\mathbf{P}_R$ problem. The result is $(l_0 + d)$-truncations of regular solutions of the system.

Example. Let $m = 3$ and matrix (30) be given as the sum of two matrices:

$$\begin{pmatrix} -x+(1+\theta-\theta^2)x^3 & \theta^3-2\theta-\theta^2x^3 & -1-x-x^2-x^3 \\ -x^3 & \theta^3-2\theta & -1-x^3 \\ -2x^3 & \theta^3-2\theta+\theta^2x & \theta-2-x^3 \end{pmatrix}$$

$$+\begin{pmatrix} \sum_{k=2}^{\infty}(\theta^2-1)x^k & \sum_{k=1}^{\infty}(\theta^2-1)x^k & 0 \\ 0 & \sum_{k=1}^{\infty}(\theta^2-1)x^k & 0 \\ 0 & \sum_{k=2}^{\infty}(\theta^2-1)x^k & \sum_{k=1}^{\infty}(\theta^2-1)x^k \end{pmatrix}.$$

In Maple, this sum can be written by means of `piecewise`:

```
L:=Matrix([[k->piecewise(k=0,0,k=1,
-1,k=3,theta,theta^2-1),
 k->piecewise(k=0,theta^3-2*theta,
             k=3,-1,theta^2-1),
 k->piecewise(k=0,-1,k=1,-1,k=2,
             -1,k=3,-1,0)],
 [k->piecewise(k=3,-1,0),
  k->piecewise(k=0,theta^3-2*theta,
             theta^2-1),
 k->piecewise(k=0,-1,k=3,-1,0)],
[k->piecewise(k=3,-2,0),
 k->piecewise(k=0,theta^3-2*theta,
           k=1,theta^2,theta^2-1),
 k->piecewise(k=0,theta-2,k=3,
     theta^2-2,theta^2-1)]]):
```

Let us find a regular solution of the corresponding system for $d = 0$ (for clearness, algebraic numbers are converted to radicals):

```
>convert(RegularSolution
(L,theta,x,0),radical);
```

$$\Big[x^{\sqrt{2}}\Big(\frac{1}{2}x\sqrt{2}c_3+O(x^2)\Big)+\ln(x)(xc_1+2x^2c_1+O(x^3))$$
$$+xc_2+O(x^2),$$
$$x^{\sqrt{2}}\Big(\Big(\frac{1}{14}\sqrt{2}c_3-\frac{2}{7}c_3\Big)x+c_3+O(x^2)\Big)$$
$$+\ln(x)\Big(xc_1-\frac{1}{4}x^2c_1+O(x^3)\Big)+c_1+xc_2+O(x^2),$$
$$x^{\sqrt{2}}\Big(-\frac{1}{2}x\sqrt{2}c_3+O(x^2)\Big)+\ln(x)(-xc_1-x^2c_1+O(x^3))$$
$$-xc_2+O(x^2)\Big]$$

In the given case, $l_0 = 1$, $\Lambda = \{0, \sqrt{2}\}$. The same system can be solved, for example, for $d = 2$:

```
>convert(RegularSolution
(L,theta,x,2),radical);
```

$$x^{\sqrt{2}}\Big(\frac{1}{2}x\sqrt{2}c_3+\Big(\frac{1}{7}\sqrt{2}c_3+\frac{3}{7}c_3\Big)x^2$$
$$+\Big(\frac{97}{112}\sqrt{2}c_3+\frac{625}{56}c_3\Big)x^3+O(x^4)\Big)$$
$$+\ln(x)\Big(xc_1+2x^2c_1+\frac{51}{8}x^3c_1+\frac{3743}{63}x^4c_1+O(x^5)\Big)$$
$$+xc_2+x^2(-c_1+2c_2)+x^3\Big(\frac{31}{8}c_1+\frac{51}{8}c_2\Big)+O(x^4),$$
$$x^{\sqrt{2}}\Big(c_3+\Big(\frac{1}{14}\sqrt{2}c_3-\frac{2}{7}c_3\Big)x+\Big(-\frac{37}{28}\sqrt{2}c_3+\frac{107}{56}c_3\Big)x^2$$
$$+\Big(\frac{143}{48}\sqrt{2}c_3-\frac{239}{56}c_3\Big)x^3+O(x^4)\Big)$$
$$+\ln(x)\Big(xc_1-\frac{1}{4}x^2c_1+\frac{19}{168}x^3c_1-\frac{1609}{28224}x^4c_1+O(x^5)\Big)$$
$$+c_1+xc_2+x^2\Big(\frac{9}{8}c_1-\frac{1}{4}c_2\Big)$$
$$+x^3\Big(-\frac{1525}{3528}c_1+\frac{19}{168}c_2\Big)+O(x^4),$$
$$x^{\sqrt{2}}\Big(-\frac{1}{2}x\sqrt{2}c_3+x^2\Big(\frac{19}{14}\sqrt{2}c_3-\frac{3}{7}c_3\Big)$$
$$+x^3\Big(\frac{25}{16}\sqrt{2}c_3-\frac{47}{8}c_3\Big)+O(x^4))$$
$$+\ln(x)\Big(-xc_1-x^2c_1+\frac{13}{8}x^3c_1-\frac{1531}{504}x^4c_1+O(x^5)\Big)$$
$$-xc_2+x^2(3c_1-c_2)+x^3\Big(-\frac{23}{8}c_1+\frac{13}{8}c_2\Big)+O(x^4)$$

ACKNOWLEDGMENTS

This work was supported in part by the Russian Foundation for Basic Research, project no. 13-01-00182-a.

REFERENCES

1. Abramov, S.A. and Khmelnov, D.E., Linear differential and difference systems: EG_δ and EG_δ-eliminations, *Programming Comput. Software*, 2013, vol. 39, no. 2, pp. 91–109.

2. Coddingthon, E.A. and Levinson, N., *Theory of Ordinary Differential Equations*, New York: McGraw-Hill, 1955.
3. Abramov, S.A., EG-eliminations, *J. Difference Equations Appl.*, 1999, vol. 5, pp. 393–433.
4. Abramov, S.A., Barkatou, M., and Khmelnov, D.E., On full rank differential systems with power series coefficients, *J. Symbolic Computation* (accepted).
5. Abramov, S.A., Barkatou, M., and Pflügel, E., Higher-order linear differential systems with truncated coefficients, *Proc. of CASC'2011*, 2011, pp. 10–24.
6. Abramov, S.A. and Bronstein, M., On solutions of linear functional systems, *Proc. of ISSAC'2001*, ACM, 2001, pp. 1–6.
7. Abramov, S.A. and Bronstein, M., Linear algebra for skew-polynomial matrices, *Rapport de Recherche INRIA, RR-4420, March 2002*, http://www.inria.fr/RRRT/RR-4420.html.
8. Abramov, S.A., Bronstein, M., and Petkovšek, M., On polynomial solutions of linear operator equations, *Proc. of ISSAC'95*, 1995, pp. 290–295.
9. Abramov, S.A., Bronstein, M., and Khmelnov, D.E., On regular and logarithmic solutions of ordinary linear differential systems, *Proc. of CASC'05*, 2005, pp. 1–10.
10. Abramov, S.A. and Khmelnov, D.E., A note on computing the regular solutions of linear differential systems, *Proc. of RWCA'04*, 2004, pp. 13–27.
11. Abramov, S. and Petkovšek, M., Special power series solutions of linear differential equations, *Proc. of FPSAC'96,* 1996, pp. 1–7.
12. Abramov, S.A., Petkovšek, M., and Ryabenko, A., Special formal series solutions of linear operator equations, *Discrete Math.,* 1997, vol. 210, pp. 1–8.
13. Barkatou, M.A, El Bacha, C., and Cluzeau, T., Simple forms of higher-order linear differential systems and their applications in computing regular solutions, *J. Symbolic Computation*, 2011, vol. 46, pp. 633–658.
14. Barkatou, M. and Pflügel, E., An algorithm computing the regular formal solutions of a system of linear differential equations, *J. Symbolic Computation*, 1999, vol. 28, pp. 569–587.
15. Cope, F.T., Formal solutions of irregular differential equations, *Am. J. Math.,* 1936, vol. 58, pp. 130–149.
16. Frobenius, G., Über die Integration der linearen Differentialgleichungen mit veränder Koefficienten, *J. für die reine und angewandte Mathematik*, 1873, vol. 76, pp. 214–235.
17. Heffter, L., *Einleitung in die Theorie der linearen Differentialgleichungen*, Leipzig: Teubner, 1894.
18. van der Hoeven, J., Fast evaluation of holonomic functions near and in regular singularities, *J. Symbolic Computation*, 2001, vol. 31, pp. 717–743.
19. Poole, E., *Introduction to the Theory of Linear Differential Equations*, New York: Dover, 1960.
20. Maple online help: http://www.maplesoft.com/support/help/

Translated by A. Pesterev