Liouvillian Solutions of Irreducible Linear Difference Equations

Yongjae Cha
Florida State University
Tallahassee, FL, USA
ycha@math.fsu.edu

Mark van Hoeij*
Florida State University
Tallahassee, FL, USA
hoeij@math.fsu.edu

ABSTRACT

In this paper we give a new algorithm to compute Liouvillian solutions of linear difference equations. Compared to the prior algorithm by Hendriks and Singer, our main contribution consists of two theorems that significantly reduce the number of combinations that the algorithm will check.

Categories and Subject Descriptors

G.2.1 [**Combinatorics**]: [Recurrences and difference equations]; I.1.2 [**Symbolic and Algebraic Manipulation**]: Algorithms—*Algebraic algorithms*

General Terms

Algorithms

1. INTRODUCTION

Let C be a field of characteristic zero and $\overline{C}$ be its closure. A linear difference operator

$$L = a_n\tau^n + a_{n-1}\tau^{n-1} + \cdots + a_0\tau^0$$

where $a_i \in C(x)$ and τ is the shift operator, corresponds to a difference equation or recurrence equation

$$a_n(x)u(x+n)+a_{n-1}(x)u(x+n-1)+\cdots+a_0(x)u(x) = 0. \quad (1)$$

The set of all such linear difference operators is denoted by $C(x)[\tau]$. A solution of L is a function u which satisfies equation (1).

In this paper we aim to find Liouvillian solutions for linear difference equations. Starting with an operator L of order n, we give an algorithm to solve L whenever L is gauge equivalent to an operator of the form $\tau^n + c\phi \in C(x)[\tau]$ (details are provided only for $n = 2$ and $n = 3$, but it is easy to generalize to higher order). Here c is a constant and ϕ is a monic rational function (a quotient of monic polynomials).

*Supported by NSF grant 0728853

ISSAC'09, July 28–31, 2009, Seoul, Korea.

For irreducible L, finding such $\tau^n + c\phi$ is equivalent to computing Liouvillian solutions, see [9, Lemma 4.1], [8, Prop. 3.1], or [5, Prop. 55]. Computing Liouvillian solutions is already solved in [9, 11, 2, 8] where this problem is reduced to computing hypergeometric solutions of some other operator $\tilde{L}$ (or system, in [8]) which has n times more finite singularities (defined in [10]) than the original operator L. The problem is that computing a hypergeometric solution is done with a combinatorial algorithm [12, 7] where the number of combinations depends exponentially on the number of singularities. We give a more direct approach, based on Theorems 2 and 3, that avoids introducing $\tilde{L}$ and the corresponding increase in the number of singularities. This way there are much fewer combination to be checked.

In essence, the efficiency problem in the existing algorithm for Liouvillian solutions is this: If one computes hypergeometric solutions of $\tilde{L}$ (denoted P_i in [9, Lemma 5.3]), and if one does not exploit the fact that the singularities of $\tilde{L}$ come from singularities of L, then the algorithms for computing hypergeometric solutions [7, 12] will try many more combinations than necessary. The contribution of this paper is to solve this problem with Theorems 2 and 3.

The paper is organized as follows. In Section 2, we review definitions and some properties of gauge transformations and valuation growths from [9], [7] and [10]. In Section 3, we define n-equivalence and show how it is related to gauge equivalence. This section also contains Theorems 2 and 3 on which our algorithm is based. Section 4 gives the algorithm for order 2 and 3. In Section 5 we work out an example of order 2 and compare with the algorithm in [9]. In Section 6, we discuss order > 3 and possible improvements for our algorithm, some of which are already incorporated in our implementation [6].

2. PRELIMINARIES AND DEFINITIONS

The statement in Theorem 1 in subsection 2.2 below was already mentioned in [7, Section 4] but a proof was not given there so we will give a proof in this paper. Apart from this proof and definition 6, everything else in this section and its subsections comes from [9], [7], [10] and [13]. The main new results in this paper are Theorems 2 and 3 in Section 3 because those are the results that make our algorithm efficient.

Let C be a field of characteristic zero and $\overline{C}$ be its algebraic closure. Existence and uniqueness (up to difference isomorphisms) of a universal extension V for difference equations with coefficients in $\overline{C}$ is proved in Section 6.2 of [13]. We will view the solutions of $L(y) = 0$ as elements of this universal extension V. This way, the solution space of L (which we will denote as $V(L) \subset V$) will be a vector space of dimension $\mathrm{ord}(L) =: \max\{i \mid a_i \neq 0\} - \min\{i \mid a_i \neq 0\}$. We will assume $a_0 \neq 0$ in this paper so that $\mathrm{ord}(L) = \max\{i \mid a_i \neq 0\}$.

One can view V as a subring of the ring $\boldsymbol{S}$ which is defined as follows

$$\boldsymbol{S} = \{[f] \mid f \in \overline{C}^{\mathbb{N}}\}$$

where $[f]$ is the equivalence class of all $\tilde{f} \in \overline{C}^{\mathbb{N}}$ for which $f - \tilde{f}$ has finite support. If f is a function that is defined on all but finitely many elements of $\mathbb{N}$ then its image $[f] \in \boldsymbol{S}$ is well defined. This way $\overline{C}(x)$ can be viewed as a subring of $\boldsymbol{S}$.

2.1 Gauge equivalence

Definition 1. *If two operators L_1 and L_2 in $C(x)[\tau]$ have same order then they are called* gauge equivalent *if there exists an operator $G \in C(x)[\tau]$ such that $G(V(L_1)) = V(L_2)$. Then G is called a* gauge transformation *from L_1 to L_2 and we denote $L_1 \sim_g L_2$.*

If L_1 and G are given then we can find $G', S \in C(x)[\tau]$ for which $G'G + SL_1 = 1$ with the extended Euclidean algorithm for $C(x)[\tau]$. Then G' is the inverse gauge transformation, that is $G'(V(L_2)) = V(L_1)$. In this way one can see that gauge equivalence is an equivalence relation.

If there exists a gauge transformation G from L_1 to L_2 (in other words, if L_1 and L_2 are gauge equivalent) then there is also a gauge transformation G_{rem} from L_1 to L_2 with $\mathrm{ord}(G_{\mathrm{rem}}) < \mathrm{ord}(L_1)$, namely take G_{rem} as the remainder of G after right division by L_1. So given a gauge transformation G from L_1 to L_2 we may assume w.l.o.g. $\mathrm{ord}(G) < \mathrm{ord}(L_1) = \mathrm{ord}(L_2)$.

Definition 2. *Let $L = \sum_{i=0}^{n} a_i \tau^i$, $a_i \in C(x)$ and $a_n = 1$ be linear difference operator. Then we can form the system $\tau(Y) = A_L Y$ where*

$$A_L = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{bmatrix}.$$

We call the matrix A_L the companion matrix *of the equation $L(y) = 0$.*

So, y is solution of $L(y) = 0$ if and only if

$$Y = (y, \tau(y), \dots, \tau^{n-1}(y))^T$$

is solution of $\tau(Y) = A_L Y$. Suppose two linear difference operators L and M are gauge equivalent. Then there is a gauge transformation G from the solutions of L to the solutions of M. Let A_L, A_M be companion matrices for L, M and A_G be corresponding matrix for G such that multiplying by A_G is a bijection from the solution space of $\tau(Y) = A_L Y$ to the solution space of $\tau(Z) = A_M Z$. Then

$$\begin{aligned} \tau(Z) &= A_M Z \\ \tau(A_G Y) &= A_M A_G Y \\ \tau(A_G)\tau(Y) &= A_M A_G Y \\ \tau(Y) &= \tau(A_G)^{-1} A_M A_G Y \end{aligned}$$

Thus

$$A_L = \tau(A_G)^{-1} A_M A_G. \tag{2}$$

2.2 Valuation growths of difference equations

Definition 3. *Let $L = a_n\tau^n + \cdots + a_0\tau^0$. After multiplying L on the left by a suitable element of $C(x)$, we may assume that the a_i are in $C[x]$ and $gcd(a_0, \dots, a_n) = 1$. Then $q \in \overline{C}$ is called a* problem point *of L if q is a root of the polynomial $a_0(x)a_n(x-n)$ and $p \in \overline{C}/\mathbb{Z}$ is called a* finite singularity *of L if L has a problem point in p (i.e. $p = q + \mathbb{Z}$ for some problem point q).*

Definition 4. *Let $p \in \overline{C}/\mathbb{Z}$. For $a, b \in p \subset \overline{C}$ we say $a > b$ iff $a - b$ is a positive integer.*

Let $L = \sum_{i=0}^{n} a_i(x)\tau^i$, $a_i(x) \in C(x)$ with $a_0 \neq 0, a_n \neq 0$ be a difference operator. We define $L_\varepsilon = \sum_{i=0}^{n} a_i(x+\varepsilon)\tau^i$ which is substituting x by $x + \varepsilon$ in L. The map $L \mapsto L_\varepsilon$ defines an embedding (as non-commutative rings) of $C(x)[\tau]$ in $C(x,\varepsilon)[\tau]$, so if $L = MN$ then $L_\varepsilon = M_\varepsilon N_\varepsilon$.

Definition 5. *Let $a \in \overline{C}(\epsilon)$. The ε-valuation $v_\varepsilon(a)$ of a at $\varepsilon = 0$ is the element of $\mathbb{Z} \cup \infty$ defined as follows: if $a \neq 0$ then $v_\varepsilon(a)$ is the largest integer $m \in \mathbb{Z}$ such that $a/\epsilon^m \in \overline{C}[[\epsilon]]$, and $v_\varepsilon(0) = \infty$.*

Let $p \in \overline{C}/\mathbb{Z}$. We denote

$$V_p(L_\varepsilon) = \{\tilde{u} : p \to \overline{C}(\varepsilon) \mid L_\varepsilon(\tilde{u}) = 0\}.$$

Choosing q_l, q_r in p. Let q_l be the smallest root (by the ordering from Definition 4) of $a_0(x)a_n(x-n)$ in p, so q_l is the smallest problem point in p. Likewise we define q_r to be the largest root of $a_0(x)a_n(x-n)$ in p. If p is not singularity, that is, if a_0 and a_n have no roots in p, then choose two arbitrary elements in p and define q_l, q_r to be those two elements.

Definition 6. *For non-zero $\tilde{u} \in V_p(L_\varepsilon)$ and for $a, b \in \mathbb{C}$ if $b = a + n - 1$, we define the* box-valuation

$$v_b^a(\tilde{u}) = \min\{v_\varepsilon(\tilde{u}(m)) | m = a, a+1, \dots, b\}.$$

Lemma 1. *With q_l, q_r chosen as above, we have*

$$v_{q-1}^{q-n}(\tilde{u}) = v_{q_l-1}^{q_l-n}(\tilde{u}) \quad \textit{for all } q \in \{q_l - 1, q_l - 2, q_l - 3, \dots\},$$

$$v_{q+n}^{q+1}(\tilde{u}) = v_{q_r+n}^{q_r+1}(\tilde{u}) \quad \textit{for all } q \in \{q_r + 1, q_r + 2, q_r + 3, \dots\}$$

.

Proof. We will only prove the first equation, the second equation can be proved likewise. Given two consecutive boxes $[q-n, \dots, q-1]$ and $[q-(n-1), \dots, q]$ the values of $\tilde{u}$ at one box can be computed from the values of $\tilde{u}$ at the other box using the relation $a_n(x+\varepsilon)\tilde{u}(x+n) + \cdots + a_0(x+\varepsilon)\tilde{u}(x) = 0$ for $x = q - n$. This computation involves a division either by $a_n(q-n+\varepsilon)$ or by $a_0(q-n+\varepsilon)$. If $q \in \{q_l-1, q_l-2, q_l-3, \dots\}$ then $a_n(q-n+\varepsilon)$ and $a_0(q-n+\varepsilon)$ have ε-valuation 0, and hence this division does not decrease the box valuation. So the valuation of each box can not be lower than the valuation of the other box, hence the boxes $[q-n, \dots, q-1]$ and $[q-(n-1), \dots, q]$ have the same box valuation. By repeating this one can check that the box valuation $v_{q-1}^{q-n}(\tilde{u})$ and $v_{q_l-1}^{q_l-n}(\tilde{u})$ must be equal for all $q \in \{q_l-1, q_l-2, q_l-3, \dots\}$. □

We define $v_{\varepsilon,l}(\tilde{u})$ as $v_{q_l-1}^{q_l-n}(\tilde{u})$ which, by Lemma 1, equals the box valuation of any box on the left of q_l. Likewise we define $v_{\varepsilon,r}(\tilde{u})$ as $v_{q_r+n}^{q_r+1}(\tilde{u})$.

DEFINITION 7. *Define the* valuation growth *of non-zero* $\tilde{u} \in V_p(L_\varepsilon)$ *as*

$$g_{p,\varepsilon}(\tilde{u}) = v_{\varepsilon,r}(\tilde{u}) - v_{\varepsilon,l}(\tilde{u}) \in \mathbb{Z}.$$

Define the set of valuation growths *of* L *at* p *as*

$$\overline{g}_p(L) = \{g_{p,\varepsilon}(\tilde{u}) \mid \tilde{u} \in V_p(L_\varepsilon), \tilde{u} \neq 0\} \subset \mathbb{Z}.$$

DEFINITION 8. *Let* L *be a difference operator and* $p \in \overline{C}/\mathbb{Z}$ *be a finite singularity of* L. *If* $\overline{g}_p(L) = \{0\}$ *then* p *is called* apparent singularity. *If* $\overline{g}_p(L)$ *has more than one element then* p *is called* essential singularity.

Note: this definition of apparent singularity is related, but not quite equivalent, to the definition in [1].

THEOREM 1. *If* L_1 *and* L_2 *are gauge equivalent then* $\overline{g}_p(L_1) = \overline{g}_p(L_2)$ *for every* $p \in \overline{C}/\mathbb{Z}$.

PROOF. Let $G = c_{n-1}\tau^{n-1} + \cdots + c_0 \in C(x)[\tau]$ be a gauge transformation from L_1 to L_2 and let $p \in \overline{C}/\mathbb{Z}$. Choose any non-zero $\tilde{u} \in V_p(L_{1,\varepsilon})$, and let $\tilde{v} = G_\varepsilon(\tilde{u})$ be the corresponding solution in $V_p(L_{2,\varepsilon})$. Then

$$\tilde{v}(q) = c_0(q+\varepsilon)\tilde{u}(q) + \cdots + c_{n-1}(q+\varepsilon)\tilde{u}(q+n-1) \quad (3)$$

for all $q \in p$. We can take $q' \in p$ such that for all $q \in \{q'-1, q'-2, q'-3, \ldots\}$

(i) $q + 2n - 2$ is smaller (Definition 4) than any problem point of L_1 and L_2

(ii) $c_0(q+\varepsilon), \ldots, c_{n-1}(q+\varepsilon)$ have ε-valuation ≥ 0.

If $q < q'$ then, by Lemma 1, $\min\{v_\varepsilon(\tilde{u}(m)) | m = q, \ldots, q + (n-1)\} = v^q_{q+(n-1)}(\tilde{u}) = v_{\varepsilon,l}(\tilde{u})$ and by (ii) and Equation (3) $v_\varepsilon(\tilde{v}(q)) \geq v^q_{q+(n-1)}(\tilde{u})$. Hence $v_\varepsilon(\tilde{v}(q)) \geq v_{\varepsilon,l}(\tilde{u})$. Repeating this for $q, q+1, \ldots, q+(n-1)$ we get $v_{\varepsilon,l}(\tilde{v}) = \min\{v_\varepsilon(\tilde{v}(m)) \mid m = q, \ldots, q+(n-1)\}) \geq v_{\varepsilon,l}(\tilde{u})$. Since gauge equivalence is an equivalence relation there is a gauge transformation $G' \in C(x)[\tau]$ from L_2 to L_1. Using this G' we can get the opposite inequality $v_{\varepsilon,l}(\tilde{v}) \leq v_{\varepsilon,l}(\tilde{u})$. In all, $v_{\varepsilon,l}(\tilde{v}) = v_{\varepsilon,l}(\tilde{u})$. In the same way one can show $v_{\varepsilon,r}(\tilde{v}) = v_{\varepsilon,r}(\tilde{u})$. Thus, $\tilde{v}$ and $\tilde{u}$ have the same valuation growth, and hence $\overline{g}_p(L_1) = \overline{g}_p(L_2)$. $\square$

LEMMA 2. *If* $p \in \overline{C}/\mathbb{Z}$ *is not a finite singularity of* L *(i.e. if* a_0 *and* a_n *have no roots in* p*) then* $\overline{g}_p(L) = \{0\}$.

PROOF. Let $\tilde{u} \in V_p(L_\varepsilon)$, $\tilde{u} \neq 0$. Following the proof of Lemma 1 one can see that $v^{q-n}_{q-1}(\tilde{u})$ is the same for every $q \in p$. Hence the valuation growth of $\tilde{u}$ is 0. $\square$

3. *N*-EQUIVALENCE CLASS

DEFINITION 9. *We say* $a, b \in C(x)$ *are* n*-equivalent if* $\frac{a}{b} = \frac{\tau^n(r)}{r}$ *for some non-zero* $r \in C(x)$ *and denote* $a \sim_n b$.

Note that n-equivalence is similar to [8, Section 3.2.1] (the new results in this section are found after the Problem Statement below). A rational function is said to be monic if it is a quotient of monic polynomials.

Notation We write a non-zero rational function as $c\phi$ where ϕ is a monic rational function and $c \in C^*$.

EXAMPLE 1. *If* $\phi(x) = (x-q_1)^{n_1}R(x)$ *and* $\tilde{\phi}(x) = (x - q_1 - n)^{n_1}R(x)$ *then* $\phi, \tilde{\phi}$ *are* n*-equivalent. This means that up to* n*-equivalence one can shift roots or poles by multiples of* n*. If all roots and poles of* $c\phi(x)$ *are in* $\mathbb{Z}$ *then* $c\phi(x)$ *is 1-equivalent to a function of the form* cx^n *and 3-equivalent to a function of the form* $cx^{n_1}(x-1)^{n_2}(x-2)^{n_3}$ *for some* $n_1, n_2, n_3 \in \mathbb{Z}$ *with* $n = n_1 + n_2 + n_3$.

We will denote $\det(L)$ as the determinant of the companion matrix of L. Let $L = a_n\tau^n + \cdots + a_0\tau^0$ then $\det(L) = (-1)^n\frac{a_0}{a_n}$. If $L \sim_g M$ and G is gauge transformation from L to M then $\det(M) = \frac{\tau(\det(A_G))}{\det(A_G)}\det(L)$ by Equation (2). Thus,

$$\det(L) \sim_1 \det(M). \quad (4)$$

If $M = \tau^n + c\phi$ with ϕ monic then Equation (4) implies

$$\frac{a_0}{a_n} \sim_1 c\phi \text{ and } c = \mathrm{lc}(a_0)/\mathrm{lc}(a_n) \quad (5)$$

where $\mathrm{lc}(a_i)$ denotes the leading coefficient of a_i. Note that this is similar to the proof of Lemma 3.13 in [8].

LEMMA 3. *If* $\phi \sim_n \tilde{\phi}$ *and if* L *is gauge equivalent to* $\tau^n + c\phi$ *then* L *is also gauge equivalent to* $\tau^n + c\tilde{\phi}$.

PROOF. $\tilde{\phi}/\phi = \tau^n(r)/r$ for some $r \in C(x)$ by definition of n-equivalence. Then $M := \tau^n + c\phi$ and $\widetilde{M} := \tau^n + c\tilde{\phi}$ are gauge equivalent because multiplying by r is a bijection from $V(M)$ to $V(\widetilde{M})$. Since gauge equivalence is an equivalence relation, $L \sim_g \widetilde{M}$. $\square$

Problem Statement
Operators of the form $\tau^n + c\phi$ can be solved easily (see subsection 3.1 for details). If L is gauge equivalent to an operator of the form $M = \tau^n + c\phi$ then we can solve L as well. However, given only L, not M, we need to find $c\phi$ up to n-equivalence (see Lemma 3) but a_0/a_n only provides it up to 1-equivalence.

LEMMA 4. *Let* $M = \tau^3 + cx^{n_1}(x-1)^{n_2}(x-2)^{n_3}$ *for some* $c \in C^*$ *and let* $p = \mathbb{Z} \in \overline{C}/\mathbb{Z}$ *then* $\min(\overline{g}_p(M)) = \min\{n_1, n_2, n_3\}$ *and* $\max(\overline{g}_p(M)) = \max\{n_1, n_2, n_3\}$.

PROOF. By Lemma 1,

$$v_{\varepsilon,l}(\tilde{u}) = \min\{v_\varepsilon(\tilde{u}(-3)), v_\varepsilon(\tilde{u}(-2)), v_\varepsilon(\tilde{u}(-1))\}$$

and

$$v_{\varepsilon,r}(\tilde{u}) = \min\{v_\varepsilon(\tilde{u}(3)), v_\varepsilon(\tilde{u}(4)), v_\varepsilon(\tilde{u}(5))\}$$

for all non-zero $\tilde{u} \in V_p(M_\varepsilon)$. Now

$$v_\varepsilon(\tilde{u}(3)) = v_\varepsilon(\tilde{u}(-3)) + n_1$$

$$v_\varepsilon(\tilde{u}(4)) = v_\varepsilon(\tilde{u}(-2)) + n_2$$

$$v_\varepsilon(\tilde{u}(5)) = v_\varepsilon(\tilde{u}(-1)) + n_3.$$

The values of $v_\varepsilon(\tilde{u}(-3))$, $v_\varepsilon(\tilde{u}(-2))$, $v_\varepsilon(\tilde{u}(-1))$ in $\mathbb{Z} \bigcup \{\infty\}$ can chosen arbitrarily by choosing suitable $\tilde{u} \in V_p(M_\varepsilon)$. Doing so it is easy to check from the five equations above that the smallest resp. largest possible value one can obtain for

$$g_{p,\varepsilon}(\tilde{u}) = v_{\varepsilon,r}(\tilde{u}) - v_{\varepsilon,l}(\tilde{u})$$

is $\min\{n_1, n_2, n_3\}$ resp. $\max\{n_1, n_2, n_3\}$. So $\min(\overline{g}_p(M)) = \min\{n_1, n_2, n_3\}$ and $\max(\overline{g}_p(M)) = \max\{n_1, n_2, n_3\}$. $\square$

Definition 10. *Let $L = a_n\tau^n + \cdots + a_0$ and let $p_1, \ldots, p_k$ be the finite singularities for L. Write $p_i = q_i + \mathbb{Z}$ for some $q_i \in \overline{C}$ then $a_0/a_n \sim_1 c\prod_{i=1}^{k}(x-q_i)^{n_i}$ for some $c \in C$ and $n_i \in \mathbb{Z}$. We call this n_i the $\sim_1$-exponent of L at p_i.*

We will first sketch the key idea before giving the Theorem below. Suppose that the operator L is gauge equivalent to some unknown $M = \tau^3 + c\phi$, and suppose for example that M is as in Lemma 4. The $\sim_1$-exponent of M (and hence of L by equation (4)) at p is $n_1 + n_2 + n_3$. Our strategy is now this: to find M, our algorithm needs to compute numbers n_1, n_2, n_3 at every singularity p. It is easy to compute the sum of these three numbers by taking the $\sim_1$-exponent of L. But we can also compute the minimum and the maximum of these three numbers using Lemma 4 combined with Theorem 1. Knowing the minimum, maximum, and sum, of three numbers, that determines those numbers up to a permutation. That is the key idea of our algorithm for order 3, and in the Theorem below.

Theorem 2. *Let $L = a_3\tau^3 + a_2\tau^2 + a_1\tau + a_0$ where $a_i \in C[x]$. Let $\{p_1, \ldots, p_k\} \subseteq \overline{C}/\mathbb{Z}$ be the set of finite singularities of L. Write $p_i = q_i + \mathbb{Z}$ for some $q_i \in \overline{C}$. Let $M_i = \max(\overline{g}_{p_i}(L))$, $m_i = \min(\overline{g}_{p_i}(L))$ and $e_i = n_i - M_i - m_i$ where n_i is the $\sim_1$-exponent of L at p_i. If L is gauge equivalent to an operator of the form $\tau^3 + c\phi$ for some monic rational function $\phi \in C(x)$ and $c \in C^*$ (c is given in equation (5)) then*

$$\phi \sim_3 \prod_i^k (x-q_i)^{n_{i,1}}(x-(q_i+1))^{n_{i,2}}(x-(q_i+2))^{n_{i,3}}$$

where $(n_{i,1}, n_{i,2}, n_{i,3})$ is a permutation of (M_i, m_i, e_i).

Proof. Let $M = \tau^3 + c\phi$ and gauge equivalent to L. We may assume that the singularities of M are a subset of $\{p_1, \ldots, p_k, p_{k+1}, \ldots, p_l\}$ for some $p_{k+1}, \ldots, p_l \in \overline{C}/\mathbb{Z}$. Write $p_i = q_i + \mathbb{Z}$ for some $q_i \in \overline{C}$. Now,

$$\phi \sim_3 \prod_{i=1}^{l}(x-q_i)^{n_{i,1}}(x-(q_i+1))^{n_{i,2}}(x-(q_i+2))^{n_{i,3}} \quad (6)$$

for some $n_{i,1}, n_{i,2}$ and $n_{i,3}$ as explained in Example 1. Then

$$\frac{a_0}{a_3} \sim_1 c\phi \sim_1 c\prod_{i=1}^{l}(x-q_i)^{n_i},$$

see Equation (5) and Example 1, with

$$n_i = n_{i,1} + n_{i,2} + n_{i,3}. \quad (7)$$

By Theorem 1 and Lemma 4,

$$M_i = \max(\overline{g}_{p_i}(L)) = \max(\overline{g}_{p_i}(\tau^3 + c\phi)) = \max\{n_{i,1}, n_{i,2}, n_{i,3}\}$$
$$m_i = \min(\overline{g}_{p_i}(L)) = \min(\overline{g}_{p_i}(\tau^3 + c\phi)) = \min\{n_{i,1}, n_{i,2}, n_{i,3}\}$$

For $i > k$ we have $\overline{g}_{p_i}(L) = \{0\}$ by Lemma 2 (p_i is not a singularity of L if $i > k$) and so $\max\{n_{i,1}, n_{i,2}, n_{i,3}\} = 0$ and $\min\{n_{i,1}, n_{i,2}, n_{i,3}\} = 0$. In all, $n_{i,1} = n_{i,2} = n_{i,3} = 0$ for all $i > k$. Thus, we can replace l in equation (6) by k:

$$\phi \sim_3 \prod_i^k (x-q_i)^{n_{i,1}}(x-(q_i+1))^{n_{i,2}}(x-(q_i+2))^{n_{i,3}}.$$

The maximum of $n_{i,1}, n_{i,2}, n_{i,3}$ is M_i and the minimum is m_i, and so the remaining number must be $e_i := n_i - M_i - m_i$ by Equation (7). This determines $n_{i,1}, n_{i,2}, n_{i,3}$ up to a permutation. □

Remark. If $p \in \overline{C}/\mathbb{Z}$ is a apparent singularity (Definition 8) of L then $n_1 = n_2 = n_3 = 0$ so then p will not be a singularity of $\tau^3 + c\phi$. Hence such p are not needed for constructing ϕ in our algorithm.

Example 2. *Suppose $L = a_3\tau^3 + a_2\tau^2 + a_1\tau + a_0$ and $p = \mathbb{Z} \in \overline{C}/\mathbb{Z}$ is the only singularity of L. If $L \sim_g (\tau^3 + c\phi)$ for some monic rational function $\phi(x) \in C(x)$ and $c \in C^*$ then for some integers n_1, n_2, n_3 one has $cx^{n_1}(x-1)^{n_2}(x-2)^{n_3} \sim_3 c\phi(x) \sim_1 \frac{a_0}{a_3} \sim_1 cx^n$, where $n = n_1 + n_2 + n_3$. Let $M = \max(\overline{g}_p(L))$, $m = \min(\overline{g}_p(L))$ and $e = n - M - m$. Then the ordered triple (n_1, n_2, n_3) is a permutation of M, m and e. If M, m and e are all distinct numbers this leaves $3! = 6$ possibilities for the ordered triple (n_1, n_2, n_3).*

More generally, if there are k singularities then we have $\leq 6^k$ combinations, with equality when M_i, m_i and e_i are all distinct for each singularity.

Theorem 3. *Let $L = a_2\tau^2 + a_1\tau + a_0$ where $a_i \in C[x]$. Suppose the singularities are $\{p_1, \ldots, p_k\} \subseteq \overline{C}/\mathbb{Z}$. Write $p_i = q_i + \mathbb{Z}$ for some $q_i \in \overline{C}$. Let $M_i = \max(\overline{g}_{p_i}(L))$ and $m_i = \min(\overline{g}_{p_i}(L))$. If L is gauge equivalent to operator of the form $\tau^2 + c\phi$ then*

$$\phi \sim_2 \prod_i \begin{cases} (x-q_i)^{M_i}(x-(q_i+1))^{m_i} \\ \text{or} \\ (x-q_i)^{m_i}(x-(q_i+1))^{M_i} \end{cases}$$

and c is as in Equation (5).

The proof is similar to the proof of Theorem 2. As an example, if $p = \mathbb{Z}$ is the only singularity of L then $\phi \sim_2 x^M(x-1)^m$ or $x^m(x-1)^M$. More generally, the number of combinations that the algorithm need to check is 2^l where l is the number of p_i for which $M_i \neq m_i$. For each combination we find a candidate ϕ up to 2-equivalence.

3.1 Solutions of $\tau^n + c\phi$

Solutions of $L = \tau^n + c\phi$ can be found easily. First find a solution $u(x)$ of $\tau + c\phi(nx)$. Let $v(x) = u(x/n)$ then

$$\begin{aligned} v(x+n) &= u((x+n)/n) = u(x/n+1) \\ &= -c\phi(nx/n)u(x/n) \\ &= -c\phi(x)v(x). \end{aligned}$$

Thus $v(x)$ is a solution of $L = \tau^n + c\phi(x)$, and $(\xi)^x v(x)$ is also solution of L for any $\xi \in \overline{C}$ with $\xi^n = 1$. We obtain a basis of $V(L)$ this way.

4. ALGORITHMS FOR ORDER 2 AND 3

Given $L = a_2\tau^2 + a_1\tau + a_0 \in C(x)[\tau]$, after clearing denominators we may assume that $a_0, a_1, a_2 \in C[x]$. Algorithm Tausqsols resp. Taucbsols below uses Theorem 3 resp. 2 to compute a set comb, the set of all candidates for ϕ. It then checks each $\phi \in$ comb.

Note that the two algorithms below only search for ϕ defined over the field C, and that C must be given in the input. If there exist Liouvillian solutions with ϕ defined not over C but over some algebraic extension C' of C, then in order to find these solutions, we need to call the algorithm with C' instead of C in the input. The problem of finding these field extensions C' of C has already been solved for hypergeometric solutions in [7, Section 8] and the same approach works here as well. The only difference is that here we have additional information that can be used to further reduce the search for C', for instance, unlike for hypergeometric solutions, in our situation the minimal field extension needed to find ϕ must necessarily be Galois over C with cyclic Galois group (this restriction is of course only useful for order > 2 because an extension of degree 2 is always cyclic). Our implementation for order 2 uses the same approach as in [7] to determine the fields C' for which we have to call Tausqsols(C', L) in order to find all Liouvillian solutions.

Algorithm Tausqsols
Input: A field C of characteristic 0, and an operator $L = a_2\tau^2 + a_1\tau + a_0$ with $a_0, a_1, a_2 \in C[x]$ and $a_2 \neq 0, a_0 \neq 0$.
Output: A basis of solutions of L if there exists an operator of the form $\tau^2 + c\phi \in C(x)[\tau]$ that is gauge equivalent to L. Otherwise the empty set.

1. Let S be the irreducible factors of a_2a_0 over C up to 1-equivalence.
2. $c := \mathrm{lc}(a_0)/\mathrm{lc}(a_2)$ as in equation (5).
3. comb $:= \{1\}$.
4. For $s \in S$ do
   (a) $p :=$ a root of s.
   (b) $m := \min(\overline{g}_p(L))$, $M := \max(\overline{g}_p(L))$.
   (c) $T := \{s(x)^m s(x-1)^M, s(x)^M s(x-1)^m\}$.
   (d) comb $:= \{ij \mid i \in \text{comb}, j \in T\}$.
5. For each $\phi \in$ comb, check if there exists a gauge transformation from $\tau^2 + c\phi$ to L, and if so, then
   (a) Compute a basis of solutions of $\tau^2 + c\phi$.
   (b) Apply the gauge transformation to the solutions of $\tau^2 + c\phi$.
   (c) Return the result of step 5b as output and stop the algorithm.
6. Return $\emptyset$.

Algorithm Taucbsols
Input: A field C of characteristic 0, and an $L = a_3\tau^3 + a_2\tau^2 + a_1\tau + a_0$ with $a_0, a_1, a_2, a_3 \in C[x]$ and $a_3 \neq 0, a_0 \neq 0$.
Output: A basis of solutions of L if there exists an operator of the form $\tau^3 + c\phi \in C(x)[\tau]$ that is gauge equivalent to L. Otherwise the empty set.

1. Let S be the irreducible factors of a_3a_0 over C up to 1-equivalence.
2. $c := \mathrm{lc}(a_0)/\mathrm{lc}(a_3)$ as in equation (5).
3. We can write
$$\frac{a_0}{a_3} = c \prod_{\substack{s \in S \\ i \in \mathbb{Z}}} s(x-i)^{n_{i,s}}$$
with only finitely many $n_{i,s} \neq 0$. Then for each $s \in S$ let $l_s := \sum_i n_{i,s} \in \mathbb{Z}$.
4. comb $:= \{1\}$.
5. For $s \in S$ do
   (a) $p :=$ root of s.
   (b) $m := \min(\overline{g}_p(L))$, $M := \max(\overline{g}_p(L))$, $e := l_s - M - m$.
   (c) $E :=$ the set of all permutations of $[m, M, e]$.
   (d) $T := \{s(x)^i s(x-1)^j s(x-2)^k \mid [i, j, k] \in E\}$.
   (e) comb $:= \{ij \mid i \in \text{comb}, j \in T\}$.
6. For each $\phi \in$ comb, check if there exists a gauge transformation from $\tau^3 + c\phi$ to L, and if so, then
   (a) Compute a basis of solutions of $\tau^3 + c\phi$.
   (b) Apply the gauge transformation to the solutions of $\tau^3 + c\phi$.
   (c) Return the result of step 6b as output and stop the algorithm.
7. Return $\emptyset$.

Finding a gauge transformation can be reduced to finding a rational solution of a system of recurrence equations, which can be done with [4] or [3]. See Section 4 of [7] for computing the set of valuation growths of a difference operator (an implementation is available in Maple as the undocumented command `‘LREtools/g_p‘`).

5. EXAMPLE

We will follow Algorithm Tausqsols with the operator $L =$
$$(3+2x)(x+4)(x+3)\tau^2 - (8x^2+32x+36)\tau - 16x(2x+5)(x+1).$$
First we get $S = \{x, x-1/2\}$. Let $p = 0 + \mathbb{Z}$ and $p' = 1/2 + \mathbb{Z}$, then
$$\overline{g}_p(L) = \{-2, -1, 0, 1, 2\} \text{ and } \overline{g}_{p'}(L) = \{0\}.$$
So p' is apparent singularity of L and it has no role in constructing ϕ. Thus
$$\text{comb} = \left\{ \frac{(x-1)^2}{x^2}, \frac{x^2}{(x-1)^2} \right\}$$
and
$$c = \frac{\mathrm{lc}(a_0(x))}{\mathrm{lc}(a_2(x))} = \frac{\mathrm{lc}(-16x(2x+5)(x+1))}{\mathrm{lc}((3+2x)(x+4)(x+3))} = -16.$$
So we have two candidates $\tau^2 - 16\frac{(x-1)^2}{x^2}$ and $\tau^2 - 16\frac{x^2}{(x-1)^2}$, and the algorithm checks if any of these candidates is gauge equivalent to L. It finds that $\tau^2 - 16\frac{x^2}{(x-1)^2}$ is gauge equivalent to L and finds the gauge transformation
$$g_1(x)\tau + g_0(x) = \frac{1}{x^3 + 2x^2}\tau + \frac{4x}{(x^2-1)^2}. \quad (8)$$
Using Section 3.1 we get a basis of solutions of $\tau^2 - 16\frac{(x+1)^2}{x^2}$, namely
$$v(x) \text{ and } (-1)^x v(x), \text{ where } v(x) = \frac{16^{\frac{x}{2}}\Gamma(\frac{x}{2}+\frac{1}{2})^2}{\Gamma(\frac{x}{2})^2}. \quad (9)$$
By applying the gauge transformation (8) to the solution (9) we get
$$g_1(x)v(x+1) + g_0(x)v(x),$$
$$(-1)^{x+1}g_1(x)v(x+1) + (-1)^x g_0(x)v(x)$$
as a basis of solutions of L, where $g_1(x), g_0(x)$ are given in equation (8).

The algorithm presented in [9] would construct an operator $\tilde{L}$ and then compute its hypergeometric solutions. In the example L given above, we find (we used Khmelnov's [11] Maple implementation to compute $\tilde{L}$)

$$\begin{aligned}\tilde{L} = &(x+3)(x+2)(8x^2+16x+9)(5+2x)^2\tau^2\\ &+(-8100-35904x-1024x^6-9216x^5\\ &-66112x^2-63744x^3-33664x^4)\tau\\ &+256x(x+1)(8x^2+32x+33)(1+2x)^2.\end{aligned}$$

Apparent singularities of L can become non-singular in the operator $\tilde{L}$, and non-singular points can become apparent singularities, but this does not matter because neither apparent singularities nor non-singular points contribute to the combinatorial problem.

Concerning the singularities that do contribute to the combinatorial problem, each singularity $p = q + \mathbb{Z}$ of L corresponds to $n := \mathrm{ord}(L)$ singularities of $\tilde{L}$, namely $p_1 = q/n + \mathbb{Z}$, $p_2 = (q+1)/n + \mathbb{Z}$, $p_3 = (q+2)/n + \mathbb{Z}, \ldots$, $p_n = (q+n-1)/n + \mathbb{Z}$. The set $\overline{g}_p(L)$ at the singularity p of L is the same as the set $\overline{g}_{p_i}(\tilde{L})$ at each of the n singularities $p_1, \ldots, p_n$ of $\tilde{L}$.

So in this example, singularity $p = 0+\mathbb{Z}$ of L corresponds to two singularities $p_1 = 0+\mathbb{Z}$ and $p_2 = 1/2+\mathbb{Z}$ of $\tilde{L}$, each of which has the same set of valuation growths $\{-2,-1,0,1,2\}$ as L has at p. We verified with a Maple computation that $\overline{g}_{p_i}(\tilde{L})$, for $i = 1, 2$, is indeed equal to $\{-2,-1,0,1,2\}$. Note that $\tilde{L}$ has another singularity, given by a root of $8x^2+16x+9$ (or of $8x^2+32x+33$ which is 1-equivalent to it). Since this singularity corresponds to a regular point of L, we conclude that this must be an apparent singularity, and indeed, we verified with a Maple computation that the set of valuation growths is $\{0\}$ at this singularity.

If we solve $\tilde{L}$ with the algorithm *hypergeomsols* in Maple, then it has to choose an element of $\overline{g}_{p_1}(\tilde{L})$ and an element of $\overline{g}_{p_2}(\tilde{L})$, and there are $5 \times 5 = 25$ ways to make such choices. Thus, the number of combinations coming from the finite singularities is 25. In contrast, our algorithm had only 2 combinations to check. For order 3, the algorithm in [9] calls hypergeomsols several times (see [2] to reduce the number of such calls) and if N is the number of combinations that hypergeomsols has to check in one such call, then the number of combinations in our algorithm is at most N^r where $r = \max\{\log(3!)/\log(3^3), \log(3)/\log(2^3)\} = 0.54....$ So we have reduced the combinatorial problem by roughly the square root.

6. POSSIBLE IMPROVEMENTS

For equations of order $d \geq 4$, the situation is similar to order 3, in the sense that for each singularity we need to determine d numbers $n_1, \ldots, n_d$. Again, from a_0/a_d we can compute $n_1 + \cdots + n_d$, while from $\overline{g}_p(L)$ we can obtain the minimum and maximum of $n_1, \ldots, n_d$. This time, however, these three pieces of data (minimum, maximum, and sum) are no longer sufficient to determine $n_1, \ldots, n_d$ up to a permutation. However, they do allow to find a finite set of candidates for $(n_1, \ldots, n_d)$. This way one obtains an algorithm for $d \geq 4$. The multiplicity of the minimum resp. maximum in $n_1, \ldots, n_d$ can be found by computing the rank of the linear maps $E_{p,r}$ resp. $E_{p,l}$ defined in [10]. This helps to further decrease the combinatorial problem.

In [9], computing Liouvillian solutions of L is reduced to computing hypergeometric solutions of some other operator that we denoted as $\tilde{L}$. Although we did not use $\tilde{L}$ in our algorithm, one can nevertheless interpret our algorithm as a way to reduce the combinatorial problem that occurs when computing the hypergeometric solutions of $\tilde{L}$. This way one can see that techniques to speed up computation of hypergeometric solutions can be adapted to speed up our algorithm as well. For instance, the p-curvature can be used in the same way as in [7].

For many irreducible operators one can quickly rule out the existence of Liouvillian solutions by computing the p-curvature, or by computing local data at infinity (the dominant term in the formal solutions). Such local data can be computed quickly, so implementing such a test improves the overall performance of the algorithm.

Another issue is that even though we have drastically reduced the number of combinations to be checked, the CPU time per combination in our approach is more than the CPU time per combination in the $\tilde{L}$ approach. In our approach, for each combination the algorithm checks if there is a gauge transformation, while in the $\tilde{L}$ approach, for each combination one has to check if there is a polynomial solution. It is not difficult to address this issue because one can translate a combination in our approach to a combination for $\tilde{L}$, and after this, the cost per combination is reduced to the cost of computing polynomial solutions. In essence this approach would be quite similar to [9] except that one has much fewer combinations to check. To properly implement this, one needs to combine this approach with [2], because [9] perform some other unnecessary computations as well (other than checking too many combinations when solving $\tilde{L}$) and those other problems are addressed in [2].

For order 2 there is another way to find Liouvillian solutions that is even faster than the approach in this paper; the number of combinations is only 1. Unfortunately, this approach does not generalize to order > 2. Given $L = a_2\tau^2 + a_1\tau + a_0 \in C(x)[\tau]$, with $a_0a_1a_2 \neq 0$ and L irreducible, compute an operator $M \in C(x)[\tau]$ of order 3 such that $u^2v \in V(M)$ for every $u \in V(L)$ and $v \in V(a_0\tau - a_2)$. Then (assuming L is irreducible) L has non-zero Liouvillian solutions iff M has a non-zero rational solution. Moreover, the Liouvillian solutions of L can be quickly computed from a rational solution of M. This approach has been implemented by graduate student Giles Levy.

7. REFERENCES

[1] S.A. Abramov, M. A. Barkatou, M. van Hoeij, *Apparent Singularities of Linear Difference Equations with Polynomial Coefficients*, AAECC, **17**, p. 117–133, (2006).

[2] S.A. Abramov, M.A. Barkatou, D.E. Khmelnov, *On m-Interlacing Solutions of Linear Difference Equations*, preprint (2009).

[3] S. A. Abramov and M. Bronstein, *On solutions of linear functional systems*, ISSAC'2001, p. 1–6, (2001).

[4] M. A. Barkatou, *Rational Solutions of Matrix Difference Equations: The Problem of Equivalence and Factorization*, ISSAC'1999, p. 277–282, (1999).

[5] R. Bomboy, *Réductibilité et résolubilité des équations aux différences finies*, PhD thesis, (2001). `www-sop.inria.fr/cafe/personnel/Raphael.Bomboy`

[6] Y. Cha, M. van Hoeij, *Implementation for Tausqsols*, `www.math.fsu.edu/~hoeij/files/DifferenceLiouv` (2008).

[7] T. Cluzeau, M. van Hoeij, *Computing hypergeometric solutions of linear difference equations*, AAECC, **17(2)**, p. 83–115, (2006).

[8] R. Feng, M. F. Singer, M. Wu, *Liouvillian Solutions of Linear Difference-Differential Equations*, preprint, arXiv:0810.1574v1, (2008).

[9] P. A. Hendriks, M. F. Singer, *Solving difference equations in finite terms*, J. Symbolic Comput., **27**, p. 239–259, (1999).

[10] M. van Hoeij, *Finite singularities and hypergeometric solutions of linear recurrence equations*, J. Pure Appl. Algebra, **139**, p. 109–131, (1999).

[11] D. Khmelnov, *Search for Liouvillian solutions of linear recurrence equations in the MAPLE computer algebra system*, Programming and Computer Software, **34**, No. 4, p. 204–209, (2008).

[12] M. Petkovšek, *Hypergeometric solutions of linear difference equations with polynomial coefficients*, J. Symbolic Comput., **14**, p. 243–264, (1992).

[13] M. van der Put, M. F. Singer, *Galois Theory of Difference Equations* , Springer-Verlag, **1666**, Lecture Notes in Mathematics, (1997).