Rational Heuristics for Rational Solutions of Riccati Equations

Manuel Bronstein
INRIA
2004, Route des Lucioles, B.P. 93
F-06902 Sophia Antipolis, France
Manuel.Bronstein@sophia.inria.fr

Thom Mulders
Institute of Scientific Computing
ETH Zentrum
CH-8092 Zürich, Switzerland
Thom.Mulders@inf.ethz.ch

Abstract

We describe some new algorithm and heuristics for computing the polynomial and rational solutions of bounded degree of a class of ordinary differential equations, which includes generalized Riccati equations. As a consequence, our methods can be used for factoring linear ordinary differential equations. Since they generate systems of algebraic equations in at most n unknowns, where n is the order of the differential equation, they are particulary effective for first-order Riccati equations, which is confirmed by experimental timings for that case. Combined with the Ulmer-Weil algorithm and a modular heuristic for guessing the degree bounds, our method yield a rational heuristic alternative to Kovacic's algorithm for solving second order linear ordinary differential equations.

Introduction

Consider ordinary differential equations of the form

$$p(x)\frac{d^n y}{dx^n} = q\left(x, y, \frac{dy}{dx}, \ldots, \frac{d^{n-1}y}{dx^{n-1}}\right) \tag{1}$$

where p and q are polynomials with coefficients in some field K. An important class of such equations are the generalized Riccati equations that are satisfied by the logarithmic derivatives of solutions of linear ordinary differential equations:

$$a_n(x)R_n(y) + \cdots + a_1(x)R_1(y) + a_0(x)R_0(y) = 0 \tag{2}$$

where

$$R_0(y) = 1 \quad \text{and} \quad R_{i+1}(y) = \frac{dR_i(y)}{dx} + yR_i(y) \text{ for } i \geq 0\,. \tag{3}$$

A straightforward approach for finding polynomial and rational solutions of bounded degrees of equations of type (1) is to replace y by a quotient of polynomials with undetermined coefficients and equate both sides of (1). This yields a system of algebraic equations for the coefficients, with a correspondence between its solutions and the solutions of (1). Even for relatively small degree bounds and for first order equations, the nonlinear systems produced by this approach are usually beyond the capacity of algebraic solvers. We propose in this paper a generalization of the series method of [1] to equations of type (1) that produces systems in n indeterminates regardless of the degree bounds. In the case of Riccati equations, this method is similar to the one in [10], except that we can choose to expand at an ordinary point of the equation rather than at a singularity, which yields better performances when the singularities are algebraic rather than rational numbers. In addition, when coupling this method with a modular heuristic based on [9] for guessing the number of extra singularities, our method yields a rational heuristic factorizer for second order linear ordinary differential operators, which is particularly efficient for non completely reducible operators.
Notation: throughout this paper, we write $'$ for d/dx and $y^{(i)}$ for d^iy/dx^i.

1 Series Solutions

Repeatedly differentiating (1), we can construct a sequence of polynomials q_m in $K[x][X_1, \ldots, X_{n+m}]$ for $m \geq 0$ such that

$$py^{(n+m)} = q_m\left(y, y', \ldots, y^{(n+m-1)}\right)\,. \tag{4}$$

Indeed, we have $q_0 = q$ and once q_m is computed, differentiating (4) yields

$$\begin{aligned} py^{(n+m+1)} + p'y^{(n+m)} \quad &= \\ &\frac{\partial q_m}{\partial x}\left(y, y', \ldots, y^{(n+m-1)}\right) + \sum_{i=1}^{n+m} \frac{\partial q_m}{\partial X_i}\left(y, y', \ldots, y^{(n+m-1)}\right) y^{(i)} \end{aligned}$$

so

$$q_{m+1} = \frac{\partial q_m}{\partial x} - p'X_{n+m+1} + \sum_{i=1}^{n+m} \frac{\partial q_m}{\partial X_i} X_{i+1}\,. \tag{5}$$

Let now $a \in K$ be such that $p(a) \neq 0$ and y be a solution of (1). If a is not a singularity of y, then the formal series expansion of y around $x = a$ is of the form $\sum_{k \geq 0} S_k (x-a)^k$ where $k!S_k = y^{(k)}(a)$. It follows from (4) that

$$S_{n+m} = \frac{q_m(a)\left(S_0, S_1, 2S_2, \ldots, (n+m-1)!S_{n+m-1}\right)}{(n+m)!p(a)} \quad \text{for } m \geq 0\,. \tag{6}$$

Formulas (5) and (6) yield an algorithm for computing a formal series solution of (1) around $x = a$ given $S_0, \ldots, S_{n-1}$.

2 Polynomial Solutions

Given an integer $N \geq 0$, consider the problem of computing the polynomial solutions of degree at most N of (1). Let $A_0, \ldots, A_{n-1}$ be indeterminates and $a \in K$ be such that $p(a) \neq 0$. Using the method of Section 1, we can compute a formal series solution $\sum_{k \geq 0} S_k (x-a)^k$ with $S_i = A_i$ for $0 \leq i < n$. Note that $S_k \in K[A_0, \ldots, A_{n-1}]$ for $k \geq 0$. Replacing y by $\sum_{k=0}^N S_k(x-a)^k$ in (1) and equating both sides yields a system of algebraic equations for $A_0, \ldots, A_{n-1}$ whose solution set in any algebraic extension E of K is isomorphic to the set of polynomial solutions of degree at most N in $E[x]$. Since polynomials are not singular at $x = a$, this method yields an algorithm for computing all the polynomial solutions of bounded degree. The following heuristic reduces the number of algebraic equations to be solved: let I be the ideal of $K[A_0, \ldots, A_{n-1}]$ generated by $(S_k)_{k>N}$, and $V_E(I)$ its set of zeroes in some algebraic extension E of K. Then $V_E(I)$ is isomorphic to the set of solutions of (1) in $E[x]$ of degree at most N via the morphism that takes $(a_0, \ldots, a_{n-1}) \in V_E(I)$ to $\sum_{k=0}^N S_k(a_0, \ldots, a_{n-1})(x-a)^k$. Let I_m be the ideal $(S_{N+1}, \ldots, S_{N+m})$ for $m > 0$. We have $I_1 \subseteq I_2 \subseteq \cdots \subseteq I$, so we compute $I_1, I_2, \ldots$ (via a Gröbner basis for each of them) until we obtain m such that $I_m = I_{m+1}$ (this is guaranteed to happen since $K[A_0, \ldots, A_{n-1}]$ is Noetherian). At this point, we replace y by $\sum_{k=0}^N S_k(a_0, \ldots, a_{n-1})(x-a)^k$ in (1) where a_i is the image of A_i in $K[A_0, \ldots, A_{n-1}]/I_m$ and equate both sides. If both sides are identically equal, which is the expectation of the heuristic, then the set of polynomial solutions of degree at most N is isomorphic to the zero set of I_m. Otherwise, we obtain additional algebraic equations that generate an ideal J such that the set of polynomial solutions of degree at most N is isomorphic to the zero set of $I_m \cup J$. This heuristic is particularly effective in the first order case, where computing $I_1, I_2, \ldots$ is reduced to setting $g_1 = S_{N+1} \in K[A_0]$ and computing $g_{m+1} = \gcd(g_m, S_{N+m+1})$ until $g_{m+1} = g_m$.

As a consequence, we can use the method of this section to compute rational solutions of bounded degree with a known denominator: given $d \in K[x]$ and an integer $N \geq 0$, to compute the solutions of (1) of the form $y = z/d$ where z is a polynomial of degree at most N. One approach is to replace y by z/d in (1), which yields an equation of the same type for z, and then look for polynomial solutions of degree at most N of that equation. Alternatively, if we have chosen a point $a \in K$ such that $p(a)d(a) \neq 0$, we can multiply the power series $\sum_{k \geq 0} S_k(x-a)^k$ by d, yielding a series $\sum_{k \geq 0} T_k(x-a)^k$, and then use $\sum_{k=0}^N T_k(x-a)^k$ as candidate solution, in which case I_m is the ideal $(T_{N+1}, \ldots, T_{N+m})$. Together with the bounding methods of [3], this yields an algorithm for finding the rational solutions with no extra poles (*i.e.* whose poles are among the singularities of the equation) of generalized Riccati equations of the type (2). A method for finding the rational solutions with extra poles is developped in the rest of this paper.

3 Rational Solutions

Given integers $L, M \geq 0$, we consider in this section the problem of computing the rational solutions of equation (1) of the form $y = P/Q \in K(x)$, where $P, Q \in K[x]$ are such that $\deg(P) \leq L$ and $\deg(Q) \leq M$. Our method for this problem is similar to the algorithm of Section 2, except that we use Pade approximation to transform a general power series solution of (1) into a rational function. It is a heuristic however, because it can fail to compute solutions, in particular any solution for which $Q(a) = 0$, where $a \in K$ is the expansion point we picked (but also solutions for which $Q(a) \neq 0$ can be missed). We describe two different methods, both based on Pade approximation. The first method computes the full Pade approximation, before extracting polynomial equations for the parameters involved, while the second only performs part of the Pade approximation, then extract equations for the parameters and use these to reduce the computational effort to compute the full Pade approximation. The methods described in this section specialize to the methods of Section 2 when the bound on the degree of the denominator is 0.

We proceed as follows: as in Section 2, we compute a power series

$$S(A_0, \ldots, A_{n-1}) = \sum_{k \geq 0} S_k(A_0, \ldots, A_{n-1})(x-a)^k$$

with $S_i = A_i$ for $0 \leq i < n$, where $A_0, \ldots, A_{n-1}$ are indeterminates and $a \in K$ is such that $p(a) \neq 0$. If there is an algebraic extension E of K and a solution $y \in E(x)$ of (1), which is nonsingular at $x = a$, then its power series expansion at $x = a$ is also a solution of (1), so there are $a_0, \ldots, a_{n-1} \in E$ such that $y = \sum_{k \geq 0} S_k(a_0, \ldots, a_{n-1})(x-a)^k$. If the degrees of the numerator and denominator of y are bounded by L and M respectively, then we can compute y from its power series by computing its $[L/M]$ Pade approximation [2]. Of course we do not know $a_0, \ldots, a_{n-1}$ a priori so we try to find algebraic equations whose solutions give candidate values for $a_0, \ldots, a_{n-1}$.

Let $N \geq 0$ be an integer. Using Pade approximation we can find polynomials $\mathcal{P}, \mathcal{Q} \in K[x][A_0, \ldots, A_{n-1}]$ such that $\deg_x(\mathcal{P}) \leq L + N, \deg_x(\mathcal{Q}) \leq M$ and

$$\frac{\mathcal{P}}{\mathcal{Q}} = S(A_0, \ldots, A_{n-1}) + \mathcal{O}(x^{L+N+M+1}).$$

Let $P_i, Q_i \in K[A_0, \ldots, A_{n-1}]$ be the coefficients of $\mathcal{P}$ and $\mathcal{Q}$ respectively in x^i, and I_0 the ideal generated by $(P_{L+1}, \ldots, P_{L+N})$. Let R be the result of replacing y by $\mathcal{P}/\mathcal{Q}$ in (1) and performing all the computations modulo I_0. The coefficients of the various powers of x in the numerator of R are again in $K[A_0, \ldots, A_{n-1}]$. Let I_1 be the ideal of $K[A_0, \ldots, A_{n-1}]$ generated by these coefficients.

Now suppose that there is an algebraic extension E of K and $a_0, \ldots, a_{n-1}$ in E such that $y = \mathcal{P}(x, a_0, \ldots, a_{n-1})/\mathcal{Q}(x, a_0, \ldots, a_{n-1})$. Since $y = P/Q$ it follows from the degree bounds that $\deg(\mathcal{P}(x, a_0, \ldots, a_{n-1})) \leq L$ and so $(a_0, \ldots, a_{n-1})$ is a zero of I_0. Since y is a solution of (1) it follows that $(a_0, \ldots, a_{n-1})$ is also a zero of I_1. It is clear that for any zero $(a_0, \ldots, a_{n-1})$ of $I_0 \cup I_1$ such that $Q(x, a_0, \ldots, a_{n-1}) \neq 0$, $\mathcal{P}(x, a_0, \ldots, a_{n-1})/\mathcal{Q}(x, a_0, \ldots, a_{n-1})$ is a solution of (1). Note that this does not mean that we find in this way all the solutions $y \in E(x)$ of the form $y = P/Q$ with $P, Q \in E[x]$ and $\deg(P) \leq L, \deg(Q) \leq M$ since it may happen that for a particular y there are no $a_0, \ldots, a_{n-1} \in E$ such that $y = \mathcal{P}(x, a_0, \ldots, a_{n-1})/\mathcal{Q}(x, a_0, \ldots, a_{n-1})$, even though y is nonsingular at $x = a$.

Example 1 *we look for solutions* $y = P/Q$ *with* $P, Q \in K[x]$ *and* $\deg(P) = 0, \deg(Q) \leq 1$ *of*

$$y'' = 1 - y\,. \tag{7}$$

The power series solution at $x = 0$ *is*

$$y = A_0 + A_1 x + \frac{1 - A_0}{2} x^2 + \cdots$$

Taking $L = 0, M = 1$ *and* $N = 1$ *in our method yields the following equations:*

$$\left\{\begin{array}{lcl} P_0 & = & A_0 Q_0 \\ P_1 & = & A_0 Q_1 + A_1 Q_0 \\ 0 & = & A_1 Q_1 + (1 - A_0) Q_0 / 2 \end{array}\right.$$

A solution to those equations is

$$\begin{array}{lcllcl} Q_0 & = & A_1; & P_0 & = & A_0 A_1; \\ Q_1 & = & (A_0 - 1)/2; & P_1 & = & (A_0^2 - A_0 + 2A_1^2)/2\,. \end{array}$$

Equating P_1 *to* 0 *gives the relation*

$$A_0^2 - A_0 + 2A_1^2 = 0\,. \tag{8}$$

The solution we get is $\mathcal{P} = P_0 = A_0 A_1$ *and* $\mathcal{Q} = Q_0 + Q_1 x = A_1 + (A_0 - 1)x/2$. *It is easy to see that there are no* $a_0, a_1 \in K$ *such that* $\mathcal{P}(x, a_0, a_1)/\mathcal{Q}(x, a_0, a_1) = 1$ *although* $y = 1$ *is an appropriate solution of (7).*

If the zero set of I_0 is finite, then the process of replacing y by $\mathcal{P}/\mathcal{Q}$ in (1) and performing the computations modulo I_0 can be split into replacing y by $\mathcal{P}(x, v)/\mathcal{Q}(x, v)$ for each zero v of I_0 with $\mathcal{Q}(x, v) \neq 0$. These computations are very efficient for rational points since no parameters or algebraic numbers are involved. In this case we have v is also a zero of I_1 if and only if $\mathcal{P}(x, v)/\mathcal{Q}(x, v)$ is a solution of (1).

The above method has two drawbacks. In order to describe them and the way we can eliminate them we first have to describe how we compute the polynomials $\mathcal{P}$ and $\mathcal{Q}$: the coefficients of $\mathcal{P}$ and $\mathcal{Q}$ have to satisfy the following system of linear equations:

$$\left\{\begin{array}{lcl} P_0 & = & S_0Q_0 \\ P_1 & = & S_1Q_0 + S_0Q_1 \\ \vdots & \vdots & \vdots \\ P_{L+N} & = & S_{L+N}Q_0 + S_{L+N-1}Q_1 + \cdots + S_0Q_{L+N} \\ 0 & = & S_{L+N+1}Q_0 + S_{L+N}Q_1 + \cdots + S_{L+N-M+1}Q_M \\ \vdots & \vdots & \vdots \\ 0 & = & S_{L+N+M}Q_0 + S_{L+N+M-1}Q_1 + \cdots + S_{L+N}Q_M \end{array}\right. \tag{9}$$

where $Q_i = 0$ if $i > M$ and $S_i = 0$ if $i < 0$.

In order to compute $\mathcal{P}$ and $\mathcal{Q}$ we first solve the last M equations for $Q_0, \ldots, Q_M$ using a fraction–free algorithm, and then substitute a solution into the first $L + N + 1$ equations to get $P_0, \ldots, P_{L+N}$. The first drawback is that during these computations we always compute with polynomials in $K[A_0, \ldots, A_{n-1}]$. The second is the fact that when we want to increase N (in order to get more equations in $A_0, \ldots, A_{n-1}$) we may have to solve a completely new linear system.

We can eliminate these two drawbacks as follows: suppose that there are $\mathcal{P}, \mathcal{Q} \in K[x][A_0, \ldots, A_{n-1}]$ such that $\deg_x(\mathcal{P}) \leq L, \deg_x(\mathcal{Q}) \leq M$ and $y = \mathcal{P}(x, a_0, \ldots, a_{n-1})/\mathcal{Q}(x, a_0, \ldots, a_{n-1})$ for some $a_0, \ldots, a_{n-1}$ in an algebraic extension E of K. Let $p_i = P_i(a_0, \ldots, a_{n-1})$, $q_i = Q_i(a_0, \ldots, a_{n-1})$ and $s_i = S_i(a_0, \ldots, a_{n-1})$. The following linear equalities then hold:

$$\left\{\begin{array}{lcl} p_0 & = & s_0q_0 \\ p_1 & = & s_1q_0 + s_0q_1 \\ \vdots & \vdots & \vdots \\ p_L & = & s_Lq_0 + s_{L-1}q_1 + \cdots + s_0q_L \\ 0 & = & s_{L+1}q_0 + s_Lq_1 + \cdots + s_{L-M+1}q_M \\ \vdots & \vdots & \vdots \\ 0 & = & s_{L+M}q_0 + s_{L+M-1}q_1 + \cdots + s_Lq_M \\ 0 & = & s_{L+M+i}q_0 + s_{L+M+i-1}q_1 + \cdots + s_{L+i}q_M \quad (\forall i \geq 1) \end{array}\right. \tag{10}$$

Since $(q_0, \ldots, q_M) \neq (0, \ldots, 0)$ is a solution of all the equations of (10) with a 0 on the left side, we see that

$$\begin{vmatrix} s_{L+1} & \cdots & s_{L-M+1} \\ \vdots & & \vdots \\ s_{L+M} & \cdots & s_L \\ s_{L+M+i} & \cdots & s_{L+i} \end{vmatrix} = 0$$

for any $i \geq 1$. Therefore, $(a_0, \ldots, a_{n-1})$ is a zero of the determinant

$$D_i = \begin{vmatrix} S_{L+1} & \cdots & S_{L-M+1} \\ \vdots & & \vdots \\ S_{L+M} & \cdots & S_L \\ S_{L+M+i} & \cdots & S_{L+i} \end{vmatrix}$$

for any $i \geq 1$. Performing fraction-free Gaussian elimination on the matrix

$$\begin{bmatrix} S_{L+1} & \cdots & S_{L-M+1} \\ \vdots & & \vdots \\ S_{L+M} & \cdots & S_L \\ S_{L+M+1} & \cdots & S_{L+1} \\ \vdots & & \vdots \\ S_{L+M+N} & \cdots & S_{L+N} \end{bmatrix} \tag{11}$$

we can transform it into the form

$$\begin{bmatrix} * & \cdots & \cdots & * & * \\ 0 & \ddots & & \vdots & \vdots \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & * & * \\ 0 & \cdots & \cdots & 0 & D_1 \\ \vdots & & & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 & D_N \end{bmatrix}$$

giving us $D_1, \ldots, D_N$. Note that during the elimination process we only use the first M rows to search for pivots. If at some stage we cannot find a nonzero pivot in the first M rows, we stop and the heuristic fails.

Let I_2 be the ideal of $K[A_0, \ldots, A_{n-1}]$ generated by $D_1, \ldots, D_N$. We now compute a solution of the linear system

$$\left\{ \begin{array}{lcl} P_0 & = & S_0 Q_0 \\ P_1 & = & S_1 Q_0 + S_0 Q_1 \\ \vdots & \vdots & \vdots \\ P_L & = & S_L Q_0 + S_{L-1} Q_1 + \cdots + S_0 Q_L \\ 0 & = & S_{L+1} Q_0 + S_L Q_1 + \cdots + S_{L-M+1} Q_M \\ \vdots & \vdots & \vdots \\ 0 & = & S_{L+M} Q_0 + S_{L+M-1} Q_1 + \cdots + S_L Q_M \end{array} \right. \tag{12}$$

but now performing all the computations modulo the ideal I_2. Let R be the result of replacing y by $\mathcal{P}/\mathcal{Q}$ in (1) and performing all the computations modulo I_2. The coefficients of the various powers of x in the numerator of R generate an ideal I_3 in $K[A_0, \ldots, A_{n-1}]$. If there are $a_0, \ldots, a_{n-1} \in E$ such that $y = \mathcal{P}(x, a_0, \ldots, a_{n-1})/\mathcal{Q}(x, a_0, \ldots, a_{n-1})$ then $(a_0, \ldots, a_{n-1})$ is a zero of $I_2 \cup I_3$.

The advantage of this second method is that we only have to compute the triangularization of the matrix (11), and not the full $\mathcal{P}$ and $\mathcal{Q}$ with coefficients in $K[A_0, \ldots, A_{n-1}]$. For the computation of $\mathcal{P}$ and $\mathcal{Q}$ themselves, we already have some equations that can be used to reduce the computational cost. The triangularization of (11) can be done row by row and from this we see that when we increase N the computation takes the same effort as if we would have started with a larger N from the beginning. This allows us to increase N and enlarge the ideal I_2 until it stabilizes, in an analogous fashion to what was done in Section 2. In the first order case, computing with ideals reduces to computing greatest common divisors in $K[A_0]$.

4 Generalized Riccati Equations

We consider now the generalized Riccati equation (2), which is a special instance of (1). Let $a_n z^{(n)} + \cdots + a_1 z' + a_0 z = 0$ be a linear ordinary differential equation, and $y = z'/z$. Then, $z^{(i)} = R_i(y)z$ for $i \geq 0$ where the R_i's are given by (3), so z is a solution of the linear equation if and only if y is a solution of the generalized Riccati equation (2). Although the general methods of the previous sections are applicable directly to Riccati equations, their special structure can be used to improve the method, and we describe the modifications in this section.

Let $y \in E(x)$ be a solution of (2) where E is an algebraic extension of K. For any irreducible factor p of a_n in $E[x]$, one can compute an upper bound m_p for the power of p that appears in the denominator of y. This is classical and goes back to Beke [6, 7]. It is in fact possible to compute m_p for all the factors of a_n and $D = \prod_{p|a_n} p^{m_p}$ without factoring a_n [3]. In a similar way, we can compute an upper bound m_∞ for the degree of the polynomial part of y. Then, y can be written as

$$u = P + \frac{A}{D} + \frac{Q'}{Q} \tag{13}$$

where $P, A, Q \in E[x]$ are such that $\deg(A) < \deg(D), \deg(P) \leq m_\infty$ and $\gcd(Q, D) = 1$. There are methods to compute an upper bound for the degree of Q (for example [10]), but those methods require computing with algebraic numbers (the zeroes of a_n), and the bounds they give can be very pessimistic. Assume that we have an upper bound m on $\deg(Q)$ (either a computed bound or a heuristically determined bound, as in the next section). We can use the methods from the previous section to search for solutions $y = P_1/P_2$ of (2), where P_1, P_2 are polynomials satisfying $\deg(P_1) \leq m + \deg(D) + m_\infty$ and $\deg(P_2) \leq m + \deg(D)$. In order to compute y in this way we have to compute Pade approximants of type $[m + \deg(D) + m_\infty / m + \deg(D)]$, which means that we have to compute $2m + 2\deg(D) + m_\infty$ terms of the series solutions of (2), and have to consider a system like (12) with approximately the same number of equations.

We can decrease the amount of work by not searching for the solution y, but by writing $y = v/D$. Setting $z'/z = v/D$, we get $D^i z^{(i)} = T_i(v)z$, where the T_i's are given by

$$T_0(v) = 1 \text{ and } T_{i+1}(v) = D\frac{dT_i(v)}{dx} - i\frac{dD}{dx}T_i(v) + vT_i(v) \text{ for } i \geq 0\,. \tag{14}$$

Hence, v is a solution of the equation

$$a_n T_n(v) + D a_{n-1} T_{n-1}(v) + \cdots + D^{n-1} a_1 T_1(v) + D^n a_0 T_0(v) = 0\,. \tag{15}$$

This is again a special case of (1), so we can use the methods of the previous sections to search its polynomial and rational solutions. Given that $v = yD$, we only have to compute the solutions $v = P_1/P_2$ of (15) with P_1 and P_2 polynomials satisfying $\deg(P_1) \leq m + \deg(D) + m_\infty$ and $\deg(Q) \leq m$. For this we have to compute $2m + \deg(D) + m_\infty$ terms of the series solutions of (15), and have to consider a system like (12) with approximately the same number of equations. If $\deg(D)$ is large this can significantly decrease the amount of work for computing y.

Since (15) is an equation of type (1), we can use the method described in Section 1 to compute its formal series solution S. There is however a more efficient method that uses the special structure of (15): when we know the coefficients of $1, (x-a), \ldots, (x-a)^t$ of $T_i(S)$, we can compute the coefficient of $(x-a)^{t-1}$ of $T_{i+1}(S)$ using the recurrence (14). Setting for $0 \leq i \leq n-2$ the i^{th} coefficient of $T_1(S) = S$ to A_i, we can compute in this way for $j = 2, \ldots, n-1$ the coefficients of $T_j(S)$ up to $(x-a)^{n-1-j}$. If we have at some point for $j = 1, \ldots, n$ the coefficients of $T_j(S)$ up to $(x-a)^{t-j}$ we can set the coefficient of $(x-a)^t$ of $T_1(S)$ to an indeterminate α, compute the next coefficient of $T_j(S)$ for $j = 2, \ldots, n$ using the recurrence (14) and compute the coefficient of $(x-a)^{n-j}$ of the left side of (15). This is an expression of the form $c\alpha + f$ with $c \in K$ and $f \in K[A_0, \ldots, A_{n-2}]$ and equating this to 0 gives the coefficient of $(x-a)^t$ of S. Since (3) is a special case of (14) with $D = 1$, this method is also applicable when computing the formal series solution of (2).

All the methods described up to this point use formal series expansions around a point $a \in K$ such that $a_n(a) \neq 0$, and can only find rational solutions y such that a is not a pole of y. In the case of Riccati equations, if $a \in K$ is such that $a_n(a) = 0$, we can also try to compute the Laurent series solutions of (2) around $x = a$, for example with the methods of [3, 10]. Using a Pade method one can then also try to find a rational solution y of (2), but now y has a pole at $x = a$. This is essentially the method used in [10] to compute right factors of a linear ordinary differential operator, but we only use it at the singularities in K of (2), not at singularities in algebraic extensions of K, for which the computational effort may increase considerably.

Finally, it is also possible to expand around $x = \infty$ and perform the whole method there regardless of whether it is a singularity of the equation. The advantange of choosing ∞ is that if it is not a singularity of the equation, then $m_\infty = 0$, so $P \in K$ in (13). Therefore, ∞ cannot be a singularity of a rational solution (while this may happen if we pick a random ordinary point in K).

5 First Order Riccati Equations

We now apply our methods to the first-order Riccati equation

$$a_2(y' + y^2) + a_1 y + a_0 = 0 \tag{16}$$

with $a_0, a_1, a_2 \in K[x]$. The existence of a solution $y \in E(x)$ of (16), where E is an algebraic extension of K, is equivalent to the reducibility of the differential operator

$$L = a_2 \frac{d^2}{dx^2} + a_1 \frac{d}{dx} + a_0 \tag{17}$$

in $E(x)[d/dx]$. As mentionned in Section 4, any solution $y \in E(x)$ of (16) can be written in the form (13), where $P, A, D, Q \in E[x]$ and $\gcd(Q, D) = 1$. Furthermore, there are rational algorithms that yield a candidate for D as well as an upper bound on $\deg(P)$. We use the modular factorization algorithm of [9] in the following way in order to heuristically compute a bound on $\deg(Q)$: for p an odd prime, we say that the equation (16) is *specializable modulo* p if there is a subring $\mathcal{O}_p$ of K containing all the coefficients of a_0, a_1, a_2 and a ring homomorphism $\sigma_p : \mathcal{O}_p \to \mathbb{F}_p$, where $\mathbb{F}_p$ is the prime field of characteristic p (note that when K is a finitely generated extension of the rational numbers $\mathbb{Q}$, then the equation is specializable modulo all but a finite number of primes). For such a prime p, we extend σ_p to $\mathcal{O}_p[x]$ via $\sigma_p(\sum_j c_j x^j) = \sum_j \sigma_p(c_j) x^j$, and define the reduction of (17) to be

$$L_p = \sigma_p(a_2) \frac{d^2}{dx^2} + \sigma_p(a_1) \frac{d}{dx} + \sigma_p(a_0) \in \mathbb{F}_p(x)[\frac{d}{dx}] .$$

Let p be an odd prime such that (16) is specializable modulo p and for which $\sigma_p(a_2) \neq 0$. Writing $\overline{a}_i = \sigma_p(a_i)$ and $\partial = d/dx + \overline{a}_1/(2\overline{a}_2)$, we get $L_p = \overline{a}_2(\partial^2 - r)$ where

$$r = \frac{1}{2} \frac{\overline{a}_1'}{\overline{a}_2} - \frac{\overline{a}_0}{\overline{a}_2} - \overline{a}_1 \frac{2\overline{a}_2' - \overline{a}_1}{4\overline{a}_2^2} .$$

Let the right-remainder of d^p/dx^p by $d^2/dx^2 - r$ in $\mathbb{F}_p(x)[d/dx]$ be $f d/dx + g$ for $f, g \in \mathbb{F}_p(x)$. If $f = 0$, then L_p has infinitely many factorizations over $\mathbb{F}_p(x)$ and the prime p yields no information. Otherwise, let

$$\Delta = \frac{1}{4} \left(\frac{f'}{f} \right)^2 - \frac{1}{2} \frac{f''}{f} + r .$$

If $\Delta = \delta^2$ for some $\delta \in \mathbb{F}_p(x)$, then

$$L_p = \overline{a}_2(\partial + u)(\partial - u) = \overline{a}_2 \left(\frac{d}{dx} + u + \frac{\overline{a}_1}{2\overline{a}_2} \right) \left(\frac{d}{dx} - u + \frac{\overline{a}_1}{2\overline{a}_2} \right)$$

where

$$u = \frac{1}{2} \frac{f'}{f} \pm \delta \quad \in \mathbb{F}_p(x) .$$

Let d_+ and d_- be the denominators of $f'/(2f) + \delta + \overline{a}_1/(2\overline{a}_2)$ and $f'/(2f) - \delta + \overline{a}_1/(2\overline{a}_2)$ respectively, $q_+ = d_+/\gcd(d_+/\sigma_p(D))$ and $q_- = d_-/\gcd(d_-/\sigma_p(D))$. Then $G_p = \{\deg(q_+), \deg(q_-)\}$ is the set of "guesses" for the degree of Q in (13) that are obtained from the prime p. In practice, we perform this reduction at several primes until we find G_p and G_q with a nonempty intersection and use $\max\{G_p \cap G_q\}$ as our bound for $\deg(Q)$. If Δ is not the square of an element of $\mathbb{F}_p(x)$, then L_p factors over a quadratic extension of $\mathbb{F}_p(x)$ but not over $\mathbb{F}_p$, and the prime p yields no information.

Once the above heuristic yields a bound on $\deg(Q)$, we apply the method of Section 4, choosing as expansion point a zero $a \in K$ of a_2 if there is one, ∞ otherwise (even if the equation is singular at $x = \infty$). The ideals generated are all in $K[A_0]$, so the nonlinear equations that arise are solved with greatest common divisor computations. When solving a unimodular second order linear ordinary differential equation of the form $a_2 y'' + a_1 y' + a_0 y = 0$, if its symmetric square

$$a_2^2 y''' + 3a_1 a_2 y'' + (2a_1^2 + 4a_0 a_2 + a_1' a_2 - a_1 a_2') y' + (4a_0 a_1 + 2a_0' a_2 - 2a_0 a_2') y = 0$$

does not have nontrivial rational solutions in $K(x)$, then it is reducible if and only if it is not completely reducible [8], so if the equation (16) has a solution $y \in \overline{K}(x)$, where $\overline{K}$ is the algebraic closure of K, then it has a unique such solution, which must in fact be in $K(x)$. This means that when computing the power series solution around a singularity of the equation (either $x = \infty$ or a zero $a \in K$ of a_2) we only have to consider the zeros in K of the characteristic equations, avoiding any computation with algebraic numbers. In addition the ideal I of Section 1 is either (1) or $(x - \alpha)$ for some $\alpha \in K$, so, together with the algorithm of [8] for the completely reducible cases, this yields a rational heuristic alternative to Kovacic's algorithm [5].

6 Timings and Comparisons

We have implemented in the MAPLE computer algebra system the method described in Section 5 for computing the rational solutions of first order Riccati equations. The heuristic bound on the number of extra singularities is computed from modular factorizations, which are done via a call to the `bernina`[1] server [4]. We have used the following examples to test the effectiveness of our method:

$$L_1 = \left(\frac{d}{dx} + \frac{x^2+2}{x^7-3}\right)\left(\frac{d}{dx} + \frac{x+5}{x^8+4}\right)$$

$$L_2 = \left(\frac{d}{dx} + \frac{x^6-2}{x^{13}+3x^{11}-2x^2-3}\right)\left(\frac{d}{dx} + \frac{x^7+5}{x^{13}+5x^{10}-3x^3+1}\right)$$

$$L_3 = \left(\frac{d}{dx} + \frac{3x^2-2}{x^5+3x^3-2x^2-2}\right)\left(\frac{d}{dx} - \frac{3x^2-1}{x^{12}+5x^7-3x^3-2}\right)$$

$$L_4 = \left(\frac{d}{dx} + \frac{1}{x^5-7}\right)\left(\frac{d}{dx} + \frac{1}{x^{12}-2}\right)$$

$$L_5 = \frac{d}{dx}^2 - x^2 - 11\,, L_6 = \frac{d}{dx}^2 - x^2 - 41\,, L_7 = \frac{d}{dx}^2 - x^2 - 201\,.$$

We have compared our implementation of the algorithm with the packages `diffop`[2] and ISOLDE[3]. Timings in CPU seconds on a DEC alpha 564 are

[1] `http://www.inria.fr/safir/WHOSWHO/Manuel.Bronstein/bernina.html`
[2] `http://klein.math.fsu.edu/~hoeij/compalg/diffop/`
[3] `http://www-lmc.imag.fr/CF/LOGICIELS/isolde_maple/isolde.html`

in the following table, where n_p is the number of modular factorizations needed to guess a bound on $\deg(Q)$ and t_p is the time needed by `bernina` to compute the modular factorizations.

	heuristic	`diffop`	ISOLDE	n_p	t_p	$\deg(Q)$
L_1	7.0	28.6	8.1	2	6.6	0
L_2	21.9	>2000	23.1	2	19.6	0
L_3	9.3	3.1	26.9	2	8.5	0
L_4	9.7	44.7	6.6	2	9.3	0
L_5	0.5	0.5	2.7	2	0.3	5
L_6	0.8	1.9	3.4	2	0.4	20
L_7	4.6(FAIL)	64.5	6.9	10	2.6	100

Our method failed to find a solution for L_7, because the guess for the bound on $\deg(Q)$ was too low. It appears that our method is faster for operators that do not have singularities in $K \cup \{\infty\}$, in which case it avoids introducing algebraic numbers. For other operators it seems that a polyalgorithm is needed that tries various strategies. Since most of the time needed by our method is spent on modular factorizations, it could be sped up significantly by trying a fixed heuristic bound first, like `diffop` does. It could also be worthwhile to first try the method of Section 2 with a bound of 0 on $\deg(Q)$.

References

[1] Sergei Abramov, Manuel Bronstein, and Marko Petkovšek. On polynomial solutions of linear operator equations. In *Proceedings of ISSAC'95*, pages 290–296. ACM Press, 1995.

[2] George A. Baker Jr. and Peter Graves-Morris. *Pade approximants.* Cambridge University Press, Cambridge, 1996.

[3] Manuel Bronstein. Linear ordinary differential equations: breaking through the order 2 barrier. In Paul Wang, editor, *Proceedings of ISSAC'92*, pages 42–48. ACM Press, 1992.

[4] Manuel Bronstein. Σ^{IT}– a strongly-typed embeddable computer algebra library. In Jacques Calmet, editor, *Proceedings of DISCO'96*, LNCS 1128, pages 22–33. Springer, 1996.

[5] J.J. Kovacic. An Algorithm for Solving Second Order Linear Homogeneous Differential Equations. *J. Symbolic Computation*, 2(1):3–43, March 1986.

[6] L. Schlesinger. *Handbuch der Theorie der linearen Differentialgleichungen.* Teubner, Leipzig, 1895.

[7] Fritz Schwarz. A factorization algorithm for linear ordinary differential equations. In Gaston Gonnet, editor, *Proceedings of ISSAC'89*, pages 17–25. ACM Press, 1989.

[8] Félix Ulmer and Jacques-Arthur Weil. Note on kovacic's algorithm. *Journal of Symbolic Computation*, 22:179–200, 1996.

[9] Marius van der Put. Reduction modulo p of differential equations. *Indagationes mathematicae*, 7:367–387, 1996.

[10] Mark van Hoeij. Factorization of differential operators with rational functions coefficients. *Journal of Symbolic Computation*, 24:537–561, 1997.