Differential Galois theory and algorithms for linear ODE's.

Marius van der Put

April 2000

1 Introduction. The questions

An excellent survey on the subject of the title has been given in [S]. The aim of this lecture is to present a selection from [S] supplemented by some newer results, short descriptions of the topics and some comments. The limitations of this presentation are due to space, time and of course the knowledge of the author.

C denotes a field of characteristic 0 and $C(z)$ will be the differential field with respect to the differentiation $f \mapsto f' = \frac{df}{dz}$. In the sequel we will also consider other differential fields. A linear differential equation (in matrix form) over $C(z)$ is an equation $y' = Ay$, where the $n \times n$ matrix A has coefficients in the field $C(z)$. The C-vector space of solutions $y \in C(z)^n$ has dimension $\leq n$. In general this dimension is $< n$ and one wants to find "symbolic solutions". The Picard-Vessiot formalism makes this precise. Under the assumption that C is algebraically closed, there is an extension $K \supset C(z)$ of differential fields such that:

(PV1) The field of constants of K is again C.

(PV2) There is a "fundamental matrix" $F \in \mathrm{GL}(n, K)$, i.e., $F' = AF$.

(PV3) The field K is generated over $C(z)$ by the entries of F.

The solution space $V \subset K^n$ is defined as $\{y \in K^n |\, y' = Ay\}$. This is a vector space over C of dimension n and the columns of F fom a basis of V. The differential automorphisms of $K/C(z)$ form a group G. This group acts in the obvious way on K^n and leaves V (as a set) invariant. The restriction of this action of G to V induces a homomorphism $G \to \mathrm{GL}(V) = \mathrm{GL}(n, C)$. This homomorphism is injective and its image is an algebraic subgroup of

GL(n, C). In this way we identify, G with a linear algebraic group. This group is called the *differential Galois group* of the equation. The *direct problem* is to make K and G as explicit as possible. Once the structure of K and G is known, many questions concerning rational solutions, algebraic solutions, Liouvillian solutions, algebraic relations between solutions, etc. are easily answered.

The *inverse problem* gives an algebraic subgroup $G \subset$ GL(n, C) and asks for an equation having differential Galois group G and satisfying some additional requirements.
A first answer to the inverse problem, namely: every linear algebraic group is the differential Galois group of a differential equation over $\mathbf{C}(z)$ with regular singularities, was given by C. Tretkoff and M. Tretkoff.
The asymptotic theory of singularities of complex differential equations by J. Martinet and J.-P. Ramis, led to a very precise answer for the complex case by J.-P. Ramis. (see [R,P2]). The all important *local theorem* can be formulated as follows:

Theorem 1.1 J.-P. Ramis' local theorem
(a) *A linear differential equation $y' = Ay$ over the field of convergent Laurent series $\mathbf{C}(\{z\})$ determines formal data and Stokes matrices.*
(b) *From these data the differential Galois group can be read off.*
(c) *Any compatible system of formal data and Stokes matrices comes from a unique equation (up to local analytic isomorphism).*
(d) *The possible differential Galois groups are the linear groups G such that $G/L(G)$ is topologically generated by one element. Here $L(G)$ denotes the (normal algebraic) subgroup generated by all maximal tori of G.*

Unfortunately, this result is not constructive. We note that any linear differential equation over the differential field $\mathbf{C}(\{z\})$ is isomorphic to an equation $zy' = My$ where the coefficients of M are in $\mathbf{C}[z^{-1}]$. This suggest the *problem* of constructing, for a given group G as in (d), an equation $zy' = My$. A special case, namely where G is supposed to be connected, has been succesfully treated by C. Mitschi and M.F. Singer, [M-S].

2 The Risch algorithm

We recall the definition of *elementary function* over $\overline{\mathbf{Q}}(z)$. One considers towers of differential fields $\overline{\mathbf{Q}}(z) = K_0 \subset K_1 \subset \cdots \subset K_n$, where each $K_i = K_{i-1}(f_i)$ and one of the following is satisfied:
(i) f_i is algebraic over K_{i-1}.
(ii) $f_i' = h' f_i$ for some $h \in K_{i-1}$. (Loosely stated, $f_i = exp\,(h)$).
(iii) $f_i' = \frac{h'}{h}$ for some $h \in K_{i-1}$. (Loosely stated $f_i = \log(h)$).
An elementary function b is an element occuring in some tower as above. The problem is to decide whether the equation $y' = b$, for elementary b, has an elementary solution and to produce this elementary solution if it exists. We note that the above problem seems to have no translation into differential Galois theory.
Risch gave a complete algorithm "on paper". Many people have contributed to further theoretical questions and implementations, e.g., M. Rothstein, B. Trager, M. Bronstein, J. Davenport (see [B]).

The implementation has only got halfway. The "transcendental part", i.e., the elementary b only involves steps of type (ii) and (iii), has been implemented in Axiom, Maple and Mathematica. There is an incomplete implementation in Axiom and Maple of the "simple algebraic part", i.e., b involves only step (i). The more difficult "general algebraic part", i.e., b involves steps (i) and steps (ii) or (iii), has only a partial implementation in Axiom.

The Risch' algorithm involves the study of solutions of the *Risch differential equation* $y' = ay + b$ over some differential field K, which are either in K or are algebraic over K. Recently a special case, namely $y' = ay$ with a algebraic over $\overline{\mathbf{Q}}(z)$, has been implemented by M. van Hoeij [H].
We note that some heuristics for the Risch equation may be derived from a new result of Y. André (see [A]):

Theorem 2.1 *Consider the equation $y' = ay$ with a an algebraic function over $\overline{\mathbf{Q}}(z)$. This has an algebraic solution if and only if for almost all primes p the reduction modulo p of the equation has a rational solution (in other words the p-curvature is 0).*

3 Kovacic' algorithm

This concerns the order two equation $y'' + ay' + by = 0$ with, say, $a, b \in C(z)$ and $C \subset \overline{\mathbf{Q}}$. It suffices to consider the reduced form $y'' = ry$ with $r \in C(z)$. The question is to produce Liouvillian solutions (if any), which amounts in this case to produce possible algebraic functions u satisfying the Riccati equation $u' + u^2 = r$. Kovacic has presented a complete algorithm, which is based on the classification of the algebraic subgroups of SL(2). There are many papers on this subject. The two main obstructions for a succesful algorithm are:

(a) One has to control the algebraic extensions of C which occur. The lecture of M. Berkenbosch, at this CATHODE 2 workshop, explains how to find those extensions a priori, i.e., without solving the equation.

(b) A straightforward calculation of the symmetric powers, required by the algorithm, of the operator $\partial^2 - r$ is very costly.

The existing implementation in Maple is rather insufficient. Algorithms for order two and order three equations have been developed by M. van Hoeij, F. Ragot, F. Ulmer and J.-A. Weil (see [H-R-U-W]). This group has also produced Maple code which, using some heuristics, attacks order two and three differential equations. M. Bronstein has implemented in Bernina an algorithm for order two equations, based upon a paper of F. Ulmer and J.-A. Weil [U-W], on papers of M. Bronstein, Th. Mulders, J.-A. Weil concerning symmetric powers (ISSAC97) and a paper [P1] of M. van der Put concerning modular factorization. The implementation of order three equations in Bernina is based upon [H-R-U-W]. The above algorithms detect the differential Galois groups involved. However not all questions, even for order two equations, can be handled by the above implementations.
Finally, the latest version of MATHEMATICA seems to have a reasonable implementation of the Kovacic algorithm.

4 Factorization of differential operators

A differential operator L is an expression of the form $a_n\partial^n + a_{n-1}\partial^{n-1} + \cdots + a_1\partial + a_0$, where ∂ stands for the operator $\frac{d}{dz}$ and where the coefficients a_i

belong to some given differential field K. The problem is to decompose L as a product of operators of lower degree in ∂. One allows here expressions which are algebraic over K.

(a) *Rational solutions.* Here one asks for solutions of the equation in K (or with coordinates in K). For scalar equations there are implementations by S.A. Abramov and M. Bronstein (in Maple and Bernina). For matrix equations there is an implementation by M.A. Barkatou (in Isolde). Both cases are restricted to $\overline{\mathbf{Q}}(z)$, in particular there are problems with parameters. (See also section 5.)

(b) *Factorization.*
(b1) *The eigenring method.* The eigenring of a differential equation L is the finite dimensional C-algebra of all the endomorphism of the equation. This eigenring is the set of all rational solutions of another differential equation associated to L. If this eigenring is not too trivial, then some factorizations of L can be deduced from it. The method was introduced by M.F. Singer and has seen small improvements by M. van Hoeij. Clearly one needs here part (a) for the computations.
(b2) *First order factors or exponential solutions.* For the formal case, M. van Hoeij implemented classical methods. Factorizations over $\overline{\mathbf{Q}}(z)$ are based on the formal local case.
(b3) *Full factorization.* Recent work of S.P. Tsarev combines the local formal factorizations of (b2) with the classical work of Beke.

A combination of the three above is used in the implementation by van Hoeij in Maple. The program is still in need of improvements.

(c) *Liouvillian solutions.* For order two equations the methods and bounds for the degrees are given by Kovacic (with small improvements by J.-A. Weil). For order 3, these are given by M.F. Singer and F. Ulmer. Algorithms of certain steps are:
(c1) *Computing invariants.* Non-implemented algorithms are given by M. van Hoeij, J.-A. Weil. Presently, M. Bronstein is working at an implementation.
(c2) *Computing symmetric powers* of an order two equation are implemented by M. Bronstein, M. van Hoeij, Th. Mulders, J.A. Weil.
(c3) *Factoring an invariant* into linear factors and computation of the mini-

mal polynomial of an algebraic solution of Riccati is present in [H-R-U-W].

Implementations by the last four authors exist in Maple. In particular, J.-A. Weil has simplified a code of M. van Hoeij. In Bernina there are implementations by M. Bronstein.

(d) *Algebraic solutions*. Apart from [F], the only known method is that of Liouvillian solutions combined with the algebraic case of Risch' algorithm. There are no implementations.

5 Computability of the direct problem

The question is: given a differential equation L, is there at least a *theoretical algorithm* which computes the Picard-Vessiot extension and the differential Galois group (and related things like rational solutions, algebraic solutions etc.)

There is a *negative answer* to this by J. Lipschitz, in the case where L contains "enough" parameters. The reasoning is as follows. A "language" involving differential fields in characteristic zero contains a language for the integers. In particular, diophantine equations can be formulated in this language. The negative answer to Hilbert's 10th problem, namely the undecidebility for diophantine equations, can be used to show that questions like the existence of rational solutions for differential equations are equally undecidable.

Positive answers are given by E. Compoint, M.F. Singer, F. Beukers and S. Bermann (see [C-S]), in the case where there are no parameters and the differential Galois group is either reductive or an extension of a reductive group by a vector group. In the general situation, say over the field $\overline{\mathbf{Q}}(z)$, it is not known whether the differential Galois group (and the Picard-Vessiot extension) is computable.

6 Construction of differential equations

More precisely, a linear algebraic group $G \subset \mathrm{GL}(n, C)$ is given and one asks for a matrix (or scalar) differential equation of order n over $C(z)$, such that the action of its differential Galois group on the solution space coincides with the given action of G on C^n. There are two *extreme cases*:
(i) G is a connected linear algebraic group (C. Mitschi and M.F. Singer).
(ii) G is a finite group (F. Ulmer and M. van der Put).

ad (i): The *constructive answer* is a matrix differential equation

$$y' = (\sum_{i=1}^{d} \frac{A_i}{z - a_i} + \sum_{j=0}^{e} B_j z^j)y,$$

where the numbers d and e can be computed from the structure of the group G. The position of the distinct regular singular points $a_1, \ldots, a_d$ is arbitrary. The point ∞ is irregular singular if some matrix B_j is not zero. The $A_1, \ldots, A_d, B_0, \ldots, B_e$ are explicitly constructed $n \times n$-matrices.

The construction uses the Lie algebra $\underline{G}$ of G. The matrices $A_1, \ldots, B_e$ are chosen in this Lie algebra. This assures that the differential Galois group of the equation is at least a subgroup of G. If the choice is "generic", then one has avoided relations which may result in a differential Galois group which is smaller than G.

A *striking special case* is: Any semi-simple connected $G \subset \mathrm{GL}(n, C)$ can be realized by an explicit equation $y' = (B_0 + B_1 z)y$ with $B_0, B_1 \in \underline{G}$. The choice of B_0, B_1 is dictated by the root space decomposition of $\underline{G}$ and the weights of the representation $\underline{G} \subset \mathrm{End}(C^n)$. Of course there are many good choices for B_0, B_1.

ad (ii): *The finite group* G is given as a subgroup of $\mathrm{GL}(n, C)$. The aim is to construct a scalar linear equation

$$L(y) := y^{(n)} + a_{n-1}y^{(n-1)} + \cdots + a_1 y^{(1)} + a_0 y = 0.$$

with all $a_j \in C(z)$, such that the given representation of G coincides with the action of the differential Galois group on the solution space $V \subset K$ of the equation. Here $K \supset C(z)$ denotes the Picard-Vessiot field, which is a finite Galois extension and has Galois group G.

Since G is finite, the equation can only have regular singularities. In the above problem, $G \subset \mathrm{GL}(n, C)$ can be seen as a finite monodromy group and one requires an *explicit* equation with this monodromy. This is the constructive, algebraic version of the Riemann-Hilbert problem.

The yet unknown Picard-Vessiot field $K \supset C(z)$ corresponds with a ramified Galois cover of curves $X \to \mathbf{P}^1_C$ with group G. One prescribes a ramification locus $S \subset \mathbf{P}^1_C$. Then X and K are determined by the choice of a surjective group homomorphism $H : \pi_1(\mathbf{P}^1 \setminus S) \to G$, where π_1 denotes the fundamental group. Although X and K are determined by H, there does not result an explicit formula for them. The next step is to consider the C-vector space $\Omega(X)$ of the holomorphic differentials on X. The group G acts on $\Omega(X)$ and the above data suffice to determine whether the given representation $G \subset \mathrm{GL}(n, C)$ is present as a subspace W of $\Omega(X)$. If so, then one continues by considering the C-vector space $V \subset K$ defined by $V \cdot dz = W$. The differential operator L with coefficients in $C(z)$ is defined by $\ker(L) = V$. For the local exponents of L at each of the singular points there are only finitely many possibilities. Using somewhat ad hoc data one hopes to find L explicitly. This method has met succes for $n = 2$ and $n = 3$. We will discuss the algorithm for $n = 3$ and $S = \{0, 1, \infty\}$.

A finite irreducible subgroup $G \subset \mathrm{GL}(3, C)$ is the input for the problem. The fundamental group of $\mathbf{P}^1 \setminus \{0, 1, \infty\}$ is given its usual presentation as the group generated by three elements $\sigma_0, \sigma_1, \sigma_\infty$ with relation $\sigma_0\sigma_1\sigma_\infty = 1$, namely three loops around $0, 1, \infty$. The homomorphism H is given by three elements $g_0, g_1, g_\infty \in G$, the images of $\sigma_0, \sigma_1, \sigma_\infty$. The possible triples (g_0, g_1, g_∞) should generate G and satisfy $g_0 g_1 g_\infty = 1$.

The algorithm

Step 1. Calculate (all) equivalence classes of triples (g_0, g_1, g_∞) as above. For a large group G this can be rather costly.

Step 2. Calculate for a given triple (g_0, g_1, g_∞) whether the given representation is present in $\Omega(X)$. This is easy and can almost be done by hand.

Step 3. Suppose that the representation occurs then one chooses one of the few possibilities for the local exponents of L. This determines L up to one constant, the *accessory parameter* μ. In lucky cases, i.e., integer exponent differences or an extra symmetry of the data (g_0, g_1, g_∞), one easily calculates the value(s) of μ. One proceeds with step 5. If not then one goes to step 4.

Step 4. One expands the local solutions of L at (say) the point $z = 0$. These Puiseux expansions contain the accessory parameter. Invariant theory for the finite group produces a symmetric power $\mathrm{sym}^k L$ of L which has a rational solution. Some linear algebra determines μ. If no value for μ is found, then an accident has happened, namely an apparent singularity is present in L. This means the end of the algorithm. If μ exists, then one goes to step 5.

Step 5. A verification that L has the required differential Galois group. This is superfluous if, after the choice of (g_0, g_1, g_∞), no further choices were needed. In the contrary case one can make a local verification at (say) $z = 0$. Moreover, the direct methods of [H-R-U-W] can be used here.

Comments.

(1) A good portion of the equations of order three found with this algorithm happens to be closely related to the generalized hypergeometric equations constructed by F. Beukers and G. Heckman. The reason is that some of the irreducible subgroups $G \subset \mathrm{GL}(3, C)$ are almost hypergeometric subgroups.

(2) The theoretical method presented here, especially its Tannakian aspect, gives rise to various other algorithms producing equations of higher order with prescribed finite differential Galois group.

7 Methods in positive characteristic

The situation is the following: K is a field of characteristic $p > 0$, such that $[K : K^p] = p$. Take any element $z \in K \setminus K^p$. Then $K = K^p(z)$ and there is a unique differentiation $f \to f'$ such that $z' = 1$. The basic examples are:

$K = k(z)$ or $K = k((z))$ with $k = k^p$ and the differentiation is given by $f' = \frac{df}{dz}$.

A linear differential equation reads again $y' = Ay$, where A is a $n \times n$-matrix with coefficients in K. The corresponding differential operator has the form $\nabla := \frac{d}{dz} - A$ and acts upon $M := K^n$. The important point is that the classification of the differential operators ∇ is very easy, even compared with the formal classification in the complex case. The algorithm for the classification of the differential operator has two steps:

(1) Compute the *p-curvature*, which is the K-linear operator ∇^p. For small primes p, say $p < 100$, and modest n this is feasible. For large p this seems very costly. The resulting ∇^p is K-linear and its matrix $\mathcal{C}$ has coefficients in K^p for a suitable basis of M.

(2) Compute the Jordan normal form of $\mathcal{C}$. This Jordan normal form classifies the differential operator ∇. In particular, all questions concerning solutions of $y' = Ay$ can be answered using this Jordan normal form.

Application to differential equations in characteristic 0
Let be given a differential operator

$$L = \partial^n + a_{n-1}\partial^{n-1} + \cdots + a_1\partial + a_0 \text{ with coefficients in } \mathbf{Q}(z).$$

For a suitable prime p one considers the operator $L_p \in \mathbf{F}_p(z)[\partial]$ obtained by reducing L (or rather is coefficients) modulo p. The classification of L_p determines the possible factorization of L_p and yields information on the factorization of L itself. By combining several primes, or by Hensel methods involving powers p^m, one can try to factor L itself.

Theory and proposals for algorithms are available. Only a minor part of this has been implemented by M. Bronstein in Bernina.

References

[A] Y. André - *Sur la conjecture des p-courbures de Grothendieck-Katz* - Institut de Mathématiques de Jussieu, Prépublication 134, Octobre 1997.

[B] M. Bronstein -?

[C-S] E. Compoint and M.F. Singer - *Computing Galois groups of Completely Reducible Differential Equations* - J. Symb. Comp. **11**, 1-22, 1998.

[F] W. Fakler - *Algebraische Algoritmen zur Lösung von linearen Differentialgleichungen* - Ph.D. thesis, University of Paderborn, 1998.

[H] M. van Hoeij - *The Minimum Polynomial of an Algebraic Solution of Abel's problem* - preprint January 2000.

[H-R-U-W] M. van Hoeij, J.-F. Ragot, F. Ulmer and J.-A. Weil - *Liouvillian solutions of linear differential equations of order three and higher* - Institut de recherche mathématiques de Rennes, prépublication 97-21, Octobre 1997.

[K] J. Kovacic - *An algorithm for solving second order linear homogeneous equations* - J. Symb. Comp. **2**, p. 3-43, 1986.

[M-S] C. Mitschi and M.F. Singer - *Connected Linear Groups as Differential Galois Groups* - Journal of Algebra **184**, 1996, 333-361.

[P1] M. van der Put - *Reduction modulo p of differential equations* - Indag. Mathem., N.S., 7(3), 367-387, 1996.

[P2] M. van der Put - *Recent work on differential Galois theory* - Sém. Bourbaki, 1997-1998, no 849, Astérisque **252**, 1998.

[P-U] M. van der Put and F. Ulmer - *Differential equations and finite groups* - to appear in Journal of Algebra, preprint available from www.msri.org (1998).

[R] J.-P. Ramis - *About the Inverse Problem in Differential Galois theory: The Differential Abhyankar Conjecture* - Manuscript for Astérisque, version 16-1-95.

[S], M.F. Singer - *Direct and Inverse Problems in Differential Galois Theory* - In: "The collected works of E. Kolchin".

[T] S.P. Tsarev - *An algorithm for Complete Enumeration of All Factorizations of a linear Ordinary Differential Operator* - ISSAC'96 Proceedings.

[U-W] F. Ulmer and J.-A. Weil - *Note on Kovacic's algorithm* - J. Sym. Comp. vol 22, no 2, p. 179-200.