DIFFERENTIAL GALOIS THEORY
BANACH CENTER PUBLICATIONS, VOLUME 58
INSTITUTE OF MATHEMATICS
POLISH ACADEMY OF SCIENCES
WARSZAWA 2002

DECOMPOSING A 4TH ORDER LINEAR DIFFERENTIAL EQUATION AS A SYMMETRIC PRODUCT

MARK VAN HOEIJ
Department of Mathematics, Florida State University
Tallahassee, FL 32306-3027, U.S.A.
E-mail: hoeij@math.fsu.edu

Abstract. Let $L(y) = 0$ be a linear differential equation with rational functions as coefficients. To solve $L(y) = 0$ it is very helpful if the problem could be reduced to solving linear differential equations of lower order. One way is to compute a factorization of L, if L is reducible. Another way is to see if an operator L of order greater than 2 is a symmetric power of a second order operator. Maple contains implementations for both of these. The next step would be to see if L is a symmetric product of two lower order equations. In this document we will show how to find the formulas needed to solve this problem for the smallest case, where the order of L is 4. This case is already non-trivial; to find the formulas the help of a computer algebra system was needed.

1. Problem description. Let $L \in \mathbf{C}(x)[\partial]$ be a linear homogeneous differential operator of order n with rational functions as coefficients, where ∂ denotes $\frac{d}{dx}$. Denote by $V(L) = \{y \in \Omega | L(y) = 0\}$ the *solution space* of L. Here Ω is a differential field that contains $\mathbf{C}(x)$ as well as n linearly independent solutions of L, where n is the *order* of L (the highest derivative in L). We could take as Ω the field of fractions of the analytic functions at $x = p$, where p is chosen as a point that will be a regular point for all operators L under consideration. Then the dimension of $V(L)$ is $n = \mathrm{order}(L)$.

If L_1 and L_2 are operators, then the *symmetric product* $L = \mathbf{sp}(L_1, L_2)$ of L_1, L_2 is defined as the monic operator L such that:

$$V(L) = \mathrm{SPAN}\{y_1 y_2 | y_1 \in V(L_1), y_2 \in V(L_2)\}.$$

It is known that such an operator $L \in \mathbf{C}(x)[\partial]$ exists and is unique, see [2]. If n_1, n_2, n are the orders of L_1, L_2, L then:

$$n_1 + n_2 - 1 \leq n \leq n_1 n_2.$$

2000 *Mathematics Subject Classification*: Primary 34-04; Secondary 34A30.
The paper is in final form and no version of it will be published elsewhere.

Here is an example how the symmetric product L can be computed with Maple:

EXAMPLE A:

```
> restart:
> with(DEtools):
> _Envdiffopdomain := [Dx, x]:
> # remark: this means that d/dx is denoted by Dx
> L1 := Dx^2+2/x*Dx+x;
```

$$L1 := Dx^2 + \tfrac{2}{x}Dx + x$$

```
> L2 := Dx^2 - x;
```

$$L2 := Dx^2 - x$$

```
> L := symmetric_product(L1,L2);
```

$$L := Dx^4 + \frac{3}{x}Dx^3 - \frac{3}{x^2}Dx^2 + 4x^2$$

The equations corresponding to L_1 and L_2 are given by the command `diffop2de`. If $y_1(x)$ and $y_2(x)$ satisfy these equations:

```
> diffop2de(L1,y1(x))=0; diffop2de(L2,y2(x))=0;
```

$$xy_1(x) + \frac{2}{x}\left(\frac{\partial}{\partial x}y_1(x)\right) + \frac{\partial^2}{\partial x^2}y_1(x) = 0$$

$$-xy_2(x) + \frac{\partial^2}{\partial x^2}y_2(x) = 0$$

then $y(x) := y_1(x)y_2(x)$ will satisfy the equation of the symmetric product:

```
> diffop2de(L,y(x))=0;
```

$$4x^2y(x) - \frac{3}{x^2}\left(\frac{\partial^2}{\partial x^2}y(x)\right) + \frac{3}{x}\left(\frac{\partial^3}{\partial x^3}y(x)\right) + \frac{\partial^4}{\partial x^4}y(x) = 0$$

For brevity, write symmetric_product as **sp**. It is well known that:

$$\mathbf{sp}(\partial - d, \partial^n + a\partial^{n-1} + \cdots) = \partial^n + (a - nd)\partial^{n-1} + \cdots \tag{1}$$

where the dots refer to lower order terms. If y is non-zero solution of $\partial - d$ then $1/y$ is a solution of $\partial + d$, and so it follows that:

$$\mathbf{sp}(\mathbf{sp}(\partial + d, L_1), \mathbf{sp}(\partial - d, L_2)) \ = \mathbf{sp}(L_1, L_2) \tag{2}$$

for any operators L_1, $L_2 \in \mathbf{C}(x)[\partial]$ and any $d \in \mathbf{C}(x)$. If

$$L = \mathbf{sp}(\partial^2 + a_1\partial + b_1, \partial^2 + a_2\partial + b_2) \tag{3}$$

for some $a_1, a_2, b_1, b_2 \in \mathbf{C}(x)$, then by taking $d = (a_2 - a_1)/4$ in equation (2) it follows that there exist $a, B_1, B_2 \in \mathbf{C}(x)$ such that

$$L = \mathbf{sp}(\partial^2 + a\partial + B_1, \partial^2 + a\partial + B_2). \tag{4}$$

So we may assume that the coefficients of ∂^1 in the two second order operators are the same. Note that if a, B_1, B_2 satisfies (4) then so does a, B_2, B_1. We wish to reduce the number of possibilities. For this purpose we will rewrite (4) as:

$$L = \mathbf{sp}(L_1, L_2) = \mathbf{sp}(\partial^2 + a\partial + b + \sqrt{c}, \partial^2 + a\partial + b - \sqrt{c}) \tag{5}$$

where $a = (a_1 + a_2)/2$, $b = (B_1 + B_2)/2$, and $c = ((B_1 - B_2)/2)^2$ are elements of $\mathbf{C}(x)$. In this paper we are interested in solving fourth order equations, and L has order 4 if and only if $c \neq 0$, which we shall assume.

Main problem. The problem to be solved in this document is the following: Given a monic operator $L \in \mathbf{C}(x)[\partial]$ of order 4, decide if there exist $a, b, c \in \mathbf{C}(x)$ such that equation (5) holds. If so, find such a, b, c.

Application. If we can find such a, b, c, then solving L has been reduced to solving two second order operators L_1, L_2. A basis of $V(L)$ is then obtained by multiplying the elements of the bases of $V(L_1)$ and $V(L_2)$.

2. An easier subproblem: The case $a = 0$. We will first calculate formulas for finding b, c under the simplifying assumption that $a = 0$.

$$L = \mathbf{sp}(L_1, L_2) = \mathbf{sp}(\partial^2 + b + \sqrt{c}, \partial^2 + b - \sqrt{c}) \tag{6}$$

```
> L1, L2 := Dx^2 + b(x)+sqrt(c(x)), Dx^2 + b(x)-sqrt(c(x)):
> L := symmetric_product(L1, L2);
```

$$\begin{aligned} L := Dx^4 - \frac{1}{2}\frac{(\frac{\partial}{\partial x}c(x))Dx^3}{c(x)} + 4b(x)Dx^2 \\ - \frac{2((\frac{\partial}{\partial x}c(x))b(x) - 3(\frac{\partial}{\partial x}b(x))c(x))Dx}{c(x)} \\ + \frac{-(\frac{\partial}{\partial x}c(x))(\frac{\partial}{\partial x}b(x)) + 4c(x)^2 + 2(\frac{\partial^2}{\partial x^2}b(x))c(x)}{c(x)} \end{aligned}$$

Now we can write the coefficient of $Dx^3 = \partial^3$ as

$$C = -\frac{1}{2}\frac{c'}{c}, \quad \text{hence} \quad c' = -2Cc \tag{7}$$

so that we can substitute $c' = -2Cc$ into L:

```
> collect(subs(diff(c(x),x) = -2*C(x)*c(x),L),Dx,normal);
```

$$\begin{aligned} Dx^4 + C(x)Dx^3 + 4b(x)Dx^2 + (4C(x)b(x) + 6\tfrac{\partial}{\partial x}b(x))Dx \\ + 2C(x)\frac{\partial}{\partial x}b(x) + 4c(x) + 2\frac{\partial^2}{\partial x^2}b(x) \end{aligned}$$

So, if L is of the form (6), then the values of $C = C(x)$ and $b = b(x)$ are easily determined:

$$C(x) = \text{coeff}(L, Dx, 3) = \text{ coefficient of } Dx^3 \tag{8}$$

$$b(x) = \frac{1}{4}\text{coeff}(L, Dx, 2) = \frac{1}{4} \cdot \text{coefficient of } Dx^2 \tag{9}$$

At this point, we can use the coefficient of Dx^1 as a **first check**: If this coefficient is not equal to $4Cb + 6b'$ then L cannot be of the form (6) for any functions b, c in any differential field extension of $\mathbf{C}(x)$. Assume that the coefficient of Dx^1 passes the check.

Then we need to calculate $c = c(x)$. It is clear how to find it:

$$c = \frac{1}{4}(\mathrm{coeff}(L, Dx, 0) - 2b'C - 2b'')$$

Then equation (7) gives us a **second check:** test if C equals $-\frac{1}{2}\frac{c'}{c}$. Again, if this test fails, then L cannot have form (6) for any b, c. It is easy to see that if L does have form (6), then it will pass both checks and b, c will be found.

We will now illustrate the $a = 0$ subproblem by two examples. In the first example, L is of the form (6). In the second example it is not.

Example 1.

```
> L := Dx^4-2/x*Dx^3+4*Dx^2-8/x*Dx+4*x^4;
```

$$L := Dx^4 - \frac{2}{x}Dx^3 + 4Dx^2 - \frac{8}{x}Dx + 4x^4$$

```
> C:=coeff(L,Dx,3);  b:=1/4*coeff(L,Dx,2);
```

$$C := -2/x$$
$$b := 1$$

```
> 4*C*b+6*diff(b,x) = coeff(L,Dx,1);
```

$$-8/x = -8/x$$

Now the first check is that this equation holds, i.e. that lhs − rhs (lefthand side minus righthand side) equals 0:

```
> first_check := normal(lhs(%)-rhs(%));
```

$$\mathit{first\_check} := 0$$

```
> c := 1/4*(coeff(L,Dx,0) - 2*C*diff(b,x) - 2*diff(b,x,x) );
```

$$c := x^4$$

```
> C = -1/2*diff(c,x)/c;
```

$$-2/x = -2/x$$

```
> second_check := normal(lhs(%)-rhs(%));
```

$$\mathit{second\_check} := 0$$

Both checks pass, therefore L is the symmetric product of L_1 and L_2 below:

```
> sqc := sqrt(c, symbolic);
```

$$sqc := x^2$$

```
> L1:=Dx^2 + b + sqc; L2:=Dx^2 + b - sqc;
```

$$L1 := Dx^2 + 1 + x^2$$
$$L2 := Dx^2 + 1 - x^2$$

```
> symmetric_product(L1,L2);  # equals L
```

$$Dx^4 - \frac{2}{x}Dx^3 + 4Dx^2 - \frac{8}{x}Dx + 4x^4$$

Example 2 (same L as in example A).

```
> L := Dx^4+3/x*Dx^3-3/x^2*Dx^2+4*x^2;
```

$$L := Dx^4 + \tfrac{3}{x}Dx^3 - \tfrac{3}{x^2}Dx^2 + 4x^2$$

```
> C:=coeff(L,Dx,3):  b:=1/4*coeff(L,Dx,2):
> 4*C*b+6*diff(b,x) = coeff(L,Dx,1):
> first_check := normal(lhs(%)-rhs(%));
```

$$\mathit{first\_check} := 0$$

```
> c := 1/4*(coeff(L,Dx,0)-2*C*diff(b,x)-2*diff(b,x,x)):
> C = -1/2*diff(c,x)/c:
> second_check := normal(lhs(%) - rhs(%));
```

$$\mathit{second\_check} := 4\frac{1}{x}$$

The second check fails. So in example 2, there exist no b, c for which equation (6) holds. However, this operator is the same as in example A. And so there do exist a, b, c such that equation (5) holds.

There exists a well known trick to eliminate the coefficient of ∂^{n-1} where $n = \text{order}(L)$. This can be done by: $\mathbf{sp}(Dx - \text{coeff}(L, Dx, n-1)/n, L)$ if L is monic.

In many algorithms for linear differential equations, the input is first *normalized* with this trick. The two second order operators L_1, L_2 in equation (5) are normalized precisely when $a = 0$. We could normalize L_1, L_2 if we knew what L_1, L_2 are, but we only know L. We may hope that normalizing L would have the same effect, however, one easily finds out that it is not so by trying an example:

Example 2a (normalize L from example 2).

```
> n:=4:
> L:=symmetric_product(Dx - coeff(L,Dx,n-1)/n, L);
```

$$L := Dx^4 - \frac{15}{8x^2}Dx^2 + \frac{15}{8x^3}Dx + \frac{1024x^6 - 315}{256x^4}$$

```
> C:=coeff(L,Dx,3):  b:=1/4*coeff(L,Dx,2):
> 4*C*b+6*diff(b,x) = coeff(L,Dx,1):
> first_check := normal(lhs(%)-rhs(%));
```

$$\mathit{first\_check} := \frac{15}{4x^3}$$

The normalized L already fails the first check, so it is not of form (6).

Remark 1. $\mathbf{sp}(\partial - a, \mathbf{sp}(L_1, L_2)) = \mathbf{sp}(\mathbf{sp}(\partial - a/2, L_1), \mathbf{sp}(\partial - a/2, L_2))$. So if L is of form (5), and if we knew the value of a, then $\mathbf{sp}(\partial - a, L)$ is of form (6) and then the problem becomes easy.

Corollary 1. *Finding a (if it exists) is equivalent to solving the main problem.*

3. Formula for the general case. The corollary could be used in the following way: Apply a transformation $L := \mathbf{sp}(\partial - a(x), L)$ where $a(x)$ is an undetermined function. Then compute first_check and second_check. Equating both to zero gives two non-linear differential equations in $a(x)$. We could try to simplify these equations and hope to find a linear differential equation. Note that $a(x)$ is related to the ∂^{n-1} coefficient, and from equation (8) we see that we could also work with this coefficient by using $C(x)$ or $c(x)$ (see equation (7)). Taking $c(x)$ makes the equations easier (the equations derived from first_check and second_check), but they are still both non-linear which makes solving difficult. The question now is: can these equations be reduced to linear differential equations, and if so, how?

To answer this question, we took the easiest possible 4th order operator, which is $L = \partial^4$, and then computed all a, b, c for which equation (5) holds, see http://www.math.fsu.edu/~hoeij/papers/symprod/idea_symprod for details. The conclusion of this computation was that there exists no linear differential equation for $a(x)$, $b(x)$, or $c(x)$, and that there does exist a linear differential equation for $c(x)^{-1/4}$, at least for the special case $L = \partial^4$. This is the key for solving the problem, because all one has to do now is to use first_check to compute an equation for $c(x)^{-1/4}$ for an arbitrary 4th order operator L, and one obtains the following result:

PROPOSITION 1. *Let $L = \partial^4 + A_4\partial^3 + A_3\partial^2 + A_2\partial + A_1$. If L is of form (5), then $c^{-1/4}$ is a solution of the operator:*

$$L_3 = 20\partial^3 + (8A_3 - 12A_4' - 3A_4^2)\partial + 12A_3' - 8A_2 + 4A_3A_4 - 10A_4'' - A_4^3 - 9A_4A_4'.$$

Proof. To verify the proposition, compute L in formula (5) in terms of $a(x), b(x), c(x)$ and their derivatives. Then take the coefficients $A_1, \ldots, A_4$ of L and determine L_3 from the given formula. Then calculate $L_3(c(x)^{-1/4})$ and simplify it to zero. This would be a very tedious and non-instructive computation if done by hand, but it can be done with very little effort with a computer algebra system. A Maple worksheet that contains this computation can be downloaded from http://www.math.fsu.edu/~hoeij/papers.html ∎

Note: If L is normalized (i.e. if $A_4 = 0$) then we can use the shorter formula

$$L_3 = \frac{5}{2}\partial^3 + A_3\partial + \frac{3}{2}A_3' - A_2.$$

PROPOSITION 2. *i is a non-zero solution of L_3 if and only if*

$$\mathbf{sp}\left(\partial - \frac{1}{4}A_4 + \frac{1}{2}\frac{i'}{i}, L\right)$$

satisfies first_check.

Proof. Again, a very tedious and non-instructive computation if done by hand, but trivial with the help of a computer algebra system. Just take $L = \partial^4 + A_4(x)\partial^3 + A_3(x)\partial^2 + A_2(x)\partial + A_1(x)$, compute the symmetric product with $\partial - \frac{1}{4}A_4(x) + \frac{1}{2}i'(x)/i(x)$, then compute first_check like in section 2. The result is precisely the formula for L_3. This computation also shows how this formula was found. ∎

This leads to the following algorithm.

4. The algorithm

INPUT: $L = \partial^4 + A_4\partial^3 + A_3\partial^2 + A_2\partial + A_1 \in \mathbf{C}(x)[\partial]$.
OUTPUT: $a, b, c \in \mathbf{C}(x)$ such that equation (5) holds, if such a, b, c exist.
STEP 1: Compute L_3 with the formula from section 3.
STEP 2: Find all i for which $L_3(i) = 0$ and $i^4 \in \mathbf{C}(x)$.
STEP 3: Compute the set of all $i'/i \in \mathbf{C}(x)$ for all $i \neq 0$ from step 2.
STEP 4: For each such i'/i, test if $\mathbf{sp}(\partial - \frac{1}{4}A_4 + \frac{1}{2}\frac{i'}{i}, L)$ satisfies second_check.

STEP 5: If so, then we can find L_1, L_2 for which $\mathbf{sp}(\partial - \frac{1}{4}A_4 + \frac{1}{2}\frac{i'}{i}, L)$ meets condition (6). Replace L_1 and L_2 by $\mathbf{sp}(\partial + \frac{1}{8}A_4 - \frac{1}{4}\frac{i'}{i}, L_1)$ and $\mathbf{sp}(\partial + \frac{1}{8}A_4 - \frac{1}{4}\frac{i'}{i}, L_2)$. Then (5) holds.

REMARKS.

- *Step 2*: Recall that $i = c^{-1/4}$ must be a solution of L_3, and $c \in \mathbf{C}(x)$ hence i^4 must be in $\mathbf{C}(x)$. Note that one can replace $\mathbf{C}(x)$ in the algorithm by another differential field, provided that an algorithm for step 2 is available.
- *Step 3*: As an algebraic set, this set is either:
  1. A finite set with 0, 1, 2 or 3 points.
  2. A projective line, so the i'/i are parametrized by homogeneous parameters $(s : t) \in P^1(\mathbf{C})$.
  3. A disjoint union of a projective line and one point.
  4. A projective plane, then the i'/i are parametrized by $(s : t : u) \in P^2(\mathbf{C})$.
- *Step 4*: i'/i depends on parameters in cases 2, 3, and 4, which means we need to translate second_check into homogeneous polynomial equations for s, t (in cases 2, 3) or for s, t, u (in case 4).
- *Implementation*: The algorithm is implemented in dsolve in Maple7. To view the code type the following in Maple7:

  ```
  > interface(verboseproc=2):
  > print(`dsolve/diffeq/higherorder/is_sympr_o2`);
  ```

  Cases 2, 3, 4 have not been fully implemented (it will consider only 2 points on the line or 3 points on the plane). The reason is that if L is reducible then it will be treated by DFactor, and if L is irreducible then cases 2, 3, 4 are rare. However, in order to have a complete procedure these cases must be implemented. It is easy to do so, and the efficiency would not be bad because no Gröbner basis is needed to solve the polynomial equations because the number of homogeneous variables is at most 3, so they can be solved with resultants and gcd's.

5. An example

$$L(y) = \frac{4}{x^2}y(x) + \frac{4}{x^2}\frac{d^2}{dx^2}y(x) + \frac{5}{x}\frac{d^3}{dx^3}y(x) + \frac{d^4}{dx^4}y(x) = 0.$$

The formula for L_3 gives:

$$L_3(y) = -\frac{16}{x^3}y(x) + \frac{17}{x^2}\frac{d}{dx}y(x) + 20\frac{d^3}{dx^3}y(x) = 0$$

which has a basis of exponential solutions, but (up to constant factors) only one solution whose 4th power is rational: $i = x^{1/2}$. So $i'/i = \frac{1}{2x}$ and we find:

$$\mathbf{sp}\left(\partial - \frac{1}{4}A_4 + \frac{1}{2}\frac{i'}{i}, L\right) = \partial^4 + \frac{1}{x}\partial^3 + \frac{1}{x^2}\partial^2 - \frac{2}{x^3}\partial + 2\frac{2x^2+1}{x^4}.$$

The latter passes first_check and second_check, so it is of the form (6), and we find $L_1 = \partial^2 + (1/4 - x)/x^2$, $L_2 = \partial^2 + (1/4 + x)/x^2$. Then L must be of the form (5), and we find $L_1 = \partial^2 + \frac{1}{x}\partial - \frac{1}{x}$, $L_2 = \partial^2 + \frac{1}{x}\partial + \frac{1}{x}$. Solving these L_1, L_2 and multiplying their solutions results in the following basis of $V(L)$:

$$\{K(0, 2\sqrt{x}) \cdot J(0, 2\sqrt{x}), I(0, 2\sqrt{x}) \cdot J(0, 2\sqrt{x}),$$
$$K(0, 2\sqrt{x}) \cdot Y(0, 2\sqrt{x}), I(0, 2\sqrt{x}) \cdot Y(0, 2\sqrt{x})\}$$

where J, Y are the Bessel functions of the first and second kind, and I, K are the modified Bessel functions of the first and second kind. On a Pentium with 266 MHz, the computation time for finding L_1, L_2 from L was 0.25 seconds, of which 0.20 seconds was spent on finding the solution $i = x^{1/2}$ of L_3, and 0.05 seconds on computing symmetric products.

6. Related problems. One can also ask when a 4th order operator L can be *transformed* into some operator M (meaning that the D-modules for L and M are isomorphic) where M is a symmetric product of second order operators. A solution to this problem is given in [1]. Since this problem is more general than the problem discussed in this document, it is to be expected that its solution will cost more CPU time. Indeed, this problem reduces to factoring a 6th order operator (the second exterior power of L) which takes considerably longer than solving a 3rd order operator L_3.

Another interesting problem is how to transform a 3rd order operator into a symmetric power of a 2nd order operator whenever possible. Singer showed that this is equivalent to finding a point, with coefficients in $\mathbf{C}(x)$, on a certain conic. Doing this efficiently (no Gröbner basis) requires an algorithm from number theory for finding a point on a conic.

The algorithm presented in this document is fast, and so it is of practical value for a differential solver because if it fails (if L is not a symmetric product) then little computation time has been lost. Similar problems for higher order operators will (unless one spends considerably more effort to optimize the algorithms) take much more computation time, making it less practical for a differential solver.

References

[1] E. Compoint and J.-A. Weil, *Absolute reducibility of differential operators and Galois groups*, in progress.

[2] M. F. Singer, *Solving homogeneous linear differential equations in terms of second order linear differential equations*, Amer. J. Math. 107 (1985), 663–696.