Symbolic-Numeric Approach to Dealing with Differential Operators

Alexandre Goyer [1]

[1]Inria, France (team MATHEXP)

Visiting team CRIStAL

Lille, France, November 9th, 2022

Linear ordinary differential equations

Denote $K = \mathbb{C}(x)$. Let $r \geqslant 1$ and $a_0, \dots, a_r \in K$ with $a_r \neq 0$.

$$(E) \quad a_r(x) f^{(r)}(x) + \cdots + a_1(x) f'(x) + a_0(x) f(x) = 0$$

"Exact resolution" of (E):

- closed form solution
  - rational, algebraic, hyperexponential, Liouvillian, ...
- structure of the solution space
  - algebraic relation between solutions,
  - subspace which is the solution space of a smaller equation,
  - direct sum of solution spaces of smaller equations, ...

Algebraic viewpoint on linear ODEs

Denote $K = \mathbb{C}(x)$. Let $r \geqslant 1$ and $a_0, \ldots, a_r \in K$ with $a_r \neq 0$.

$$\boxed{(E) \quad a_r(x)f^{(r)}(x) + \cdots + a_1(x)f'(x) + a_0(x)f(x) = 0}$$

$$\Updownarrow$$

$$\boxed{P = a_r\partial^r + \cdots + a_1\partial + a_0 \in K\langle\partial\rangle} \qquad \partial = \frac{d}{dx}$$

Let denote $\mathcal{D} = K\langle\partial\rangle$ the algebra of linear differential operators.

Commutation rule: $\partial x = x\partial + 1$ Leibniz rule: $(xf)' = xf' + f$

Interest: many properties of $\mathbb{F}[X]$ are still available in $K\langle\partial\rangle$ ☺

The differential Galois group

The differential Galois group associated to P:

- measures what algebra can see about the solutions,
- is (very) hard to compute [Hrushovski, 2002],
- is not (so) hard to approximate numerically.

Common thread of this talk

From this approximation, how to deduce properties about the exact solutions?
And, is it possible to certify the result rigorously?

Initiated by: [Chudnovsky brothers, 1990], [van der Hoeven, 2007]

1) The differential Galois group

2) Factorization of differential operators

3) Algebraicity of the solutions

1) The differential Galois group

Assumption: $0 \notin \bigcup_{i=0}^{r-1} \left\{ \text{poles of } \frac{a_i}{a_r} \right\}$ (singularities)

Cauchy-type theorem:
There are $f_1, \ldots, f_r \in \mathbb{K}[[x]]$ linearly independent solutions of P.

Definitions: $\mathrm{Sol}(P) := \mathrm{Span}_{\mathbb{C}}(f_1, \ldots, f_r) \xleftrightarrow{\sim} \mathbb{C}^r$

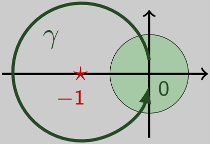
$E := \mathbb{C}(x)(f_1, \ldots, f_r, f_1', \ldots, f_r^{(r-1)})$ (Picard–Vessiot extension)

$\mathrm{Gal}_{\mathrm{diff}}(P) := \{ \sigma \in \mathrm{Aut}\,(E/\mathbb{C}(x)) \mid \forall f \in E,\ \sigma(f') = \sigma(f)' \}$
↳ acts linearly on $\mathrm{Sol}(P)$

Theorem: $\mathrm{Gal}_{\mathrm{diff}}(P)$ is a linear algebraic group.

$\mathbb{C}^{r\times r} \supset \mathcal{G} :=$ a matrix representation of $\mathrm{Gal}_{\mathrm{diff}}(P)$

Example: $Q = (x+1)\partial^2 + \partial$ admits $f_1 = 1$, $f_2 = \log(1+x)$ as solutions



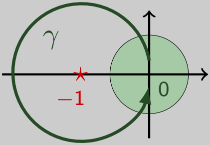
$$af_1 + bf_2 \xrightarrow[\text{along } \gamma]{\text{analytic continuation}} (a + 2i\pi b)f_1 + bf_2$$

$$\begin{pmatrix} a + 2i\pi b \\ b \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 2i\pi \\ 0 & 1 \end{pmatrix}}_{\text{local monodromy matrix around } -1} \begin{pmatrix} a \\ b \end{pmatrix}$$

Monodromy

e.g. [Mitschi, Sauzin, 2016]

Example: $Q = (x+1)\partial^2 + \partial$ admits $f_1 = 1$, $f_2 = \log(1+x)$ as solutions



$$af_1 + bf_2 \xrightarrow[\text{along } \gamma]{\text{analytic continuation}} (a + 2i\pi b)f_1 + bf_2$$

$$\begin{pmatrix} a + 2i\pi b \\ b \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 2i\pi \\ 0 & 1 \end{pmatrix}}_{\text{local monodromy matrix around } -1} \begin{pmatrix} a \\ b \end{pmatrix}$$

$\mathcal{M} :=$ group generated by all the local monodromy matrices of P

Theorem [Schlesinger, 1885]
If P is Fuchsian then $\mathcal{G} = \overline{\mathcal{M}}$ (Zariski topology).

computing these matrices $\simeq$ numerical evaluation of the solutions

2) Factorization of differential operators

joint work with
Marc Mezzarobba and Frédéric Chyzak
(ISSAC 2022)

Factorization algorithms

1894 Beke: hyperexponential solutions + exterior power method

1996 Singer: eigenring method

1997 van Hoeij: local-to-global method

↳ Maple DEtools package 

2004 Cluzeau, van Hoeij: modular approach

2007 van der Hoeven: symbolic-numeric approach

2013 Johansson, Kauers, Mezzarobba: hyperexponential solutions by symbolic-numeric approach

2014 Llorente: Liouvillian solutions by symbolic-numeric approach

Improvements of Beke's algorithm

1989 Schwarz

1990 Grigor'ev (complexity analysis)

1994 Bronstein

1996 Tsarev

Factorization algorithms

1894 **Beke:** hyperexponential solutions + exterior power method

1996 **Singer:** eigenring method

1997 **van Hoeij:** local-to-global method

↳ Maple DEtools package

2004 **Cluzeau, van Hoeij:** modular approach

2007 **van der Hoeven:** symbolic-numeric approach

2013 **Johansson, Kauers, Mezzarobba:** hyperexponential solutions by symbolic-numeric approach

2014 **Llorente:** Liouvillian solutions by symbolic-numeric approach

2022 **Chyzak, G., Mezzarobba (this work):** hybrid algorithm

↳ SageMath package

Factorization algorithms

1894 Beke: hyperexponential solutions + exterior power method

1996 Singer: eigenring method

exponential parts

1997 van Hoeij: local-to-global method

↳ Maple DEtools package

2004 Cluzeau, van Hoeij: modular approach

monodromy

2007 van der Hoeven: symbolic-numeric approach

2013 Johansson, Kauers, Mezzarobba: hyperexponential solutions by symbolic-numeric approach

2014 Llorente: Liouvillian solutions by symbolic-numeric approach

2022 Chyzak, G., Mezzarobba (this work): hybrid algorithm

↳ SageMath package

Module point of view

e.g. [Singer, van der Put, 2003]

$$\boxed{\mathcal{A} := \mathbb{C}[\mathcal{M}]} \subset \mathbb{C}^{r\times r} \quad \text{(group algebra)}$$

Proposition: There is a one-to-one correspondence:

$$\text{right-hand factor } R \text{ of } P \quad \xleftrightarrow[U \simeq \mathrm{Sol}(R)]{} \quad \mathcal{A}\text{-submodule } U \text{ of } \mathbb{C}^r$$

Strategy for finding a right-hand factor [van der Hoeven, 2007]

1. Loop over precision until success:
2. Compute approximations of the local monodromy matrices
3. Search for an approximate nontrivial proper $\mathcal{A}$-submodule U
4. If no such U exists then
5. Exploit error bounds to ensure the irreducibility
6. Else guess a symbolic $R \in \overline{\mathbb{K}}(x)\langle\partial\rangle$ from U and check it

Finding an $\mathcal{A}$-submodule

e.g. [Holt, Rees, 1994]

Norton's criterion (variant):
Let $\mathcal{A}$ be a matrix algebra and $M \in \mathcal{A}$ with a simple eigenvalue λ.
Let nonzero $u \in \mathbb{C}^{r\times 1}$, $v \in \mathbb{C}^{1\times r}$ be such that $Mu = \lambda u$, $vM = \lambda v$.
Then the left $\mathcal{A}$-module $\mathbb{C}^{r\times 1}$ is irreducible iff $\mathcal{A}u = \mathbb{C}^{r\times 1}$ and $v\mathcal{A} = \mathbb{C}^{1\times r}$.

Note: $\mathcal{A}U \subset U$ iff $V\mathcal{A} \subset V$ where $V = \{v \in \mathbb{C}^{1\times r} \mid \forall u \in U, \langle u, v\rangle = 0\}$.

Proof. Let $U, W \subset \mathbb{C}^{r\times 1}$ be such that $\mathcal{A}U \subset U$ and $\mathbb{C}^{r\times 1} = U \oplus W$.

With respect to an adapted basis: $\mathcal{A} \subset \left\{ \left(\begin{array}{c|c} * & * \\ \hline 0 & * \end{array} \right) \right\}$.

Therefore:

$$M =: \left(\begin{array}{c|c} M_1 & M_3 \\ \hline 0 & M_2 \end{array} \right) \quad \begin{array}{l} \xrightarrow{\text{either}} \lambda \in \mathrm{Sp}(M_1) \longrightarrow \mathcal{A}u \subsetneq \mathbb{C}^{r\times 1} \\ \xrightarrow[\text{or}]{} \lambda \in \mathrm{Sp}(M_2) \longrightarrow v\mathcal{A} \subsetneq \mathbb{C}^{1\times r} \end{array}$$

∎

Link with exponential parts

[van Hoeij, 1996]

Assume that P is Fuchsian. Let σ be a singular point of P.
There is a basis of local solutions at σ of the form

$$f(x) = (x-\sigma)^{\boldsymbol{\alpha}} \sum_{0 \leqslant j \leqslant d} s_j(x-\sigma) \log(x-\sigma)^j \quad \text{where} \quad \begin{cases} \boldsymbol{\alpha} \in \overline{\mathbb{K}}, d \in \mathbb{Z}_{\geqslant 0}, \\ s_0, \ldots, s_d \in \overline{\mathbb{K}}[[x]]. \end{cases}$$

$$(x-\sigma)^\alpha = e^{\alpha \log(x-\sigma)} \xrightarrow{\text{monodromy after } \circlearrowleft \text{ around } \sigma} \boldsymbol{e^{2i\pi\alpha}}(x-\sigma)^\alpha$$

eigenvalues $e^{2i\pi\alpha}$ of the local monodromy matrix $\xleftrightarrow{\text{one-to-one correspondence}}$ exponential parts $:= \alpha + \mathbb{Z}$

Link with exponential parts [van Hoeij, 1996]

Assume that P is Fuchsian. Let σ be a singular point of P.
There is a basis of local solutions at σ of the form

$$f(x) = (x-\sigma)^{\boldsymbol{\alpha}} \sum_{0\leqslant j\leqslant d} s_j(x-\sigma)\log(x-\sigma)^j \quad \text{where} \quad \begin{cases} \boldsymbol{\alpha} \in \overline{\mathbb{K}}, d \in \mathbb{Z}_{\geqslant 0}, \\ s_0, \ldots, s_d \in \overline{\mathbb{K}}[[x]]. \end{cases}$$

$$(x-\sigma)^{\alpha} = e^{\alpha\log(x-\sigma)} \xrightarrow{\text{monodromy after } \circlearrowleft \text{ around } \sigma} \boldsymbol{e}^{2\boldsymbol{i\pi\alpha}}(x-\sigma)^{\alpha}$$

eigenvalues $e^{2i\pi\alpha}$ of the local monodromy matrix $\xleftrightarrow{\text{one-to-one correspondence}}$ exponential parts $:= \alpha + \mathbb{Z}$

Remark: in van Hoeij's local-to-global method:

e is an exponential part of multiplicity 1 of $P = LR$ $\xrightarrow{\text{either}}$ e is an exponential part of R

e is an exponential part of multiplicity 1 of $P = LR$ $\xrightarrow{\text{or}}$ e is an exponential part of L

> the local-to-global method $\simeq$
> Norton's criterion with M a **local** monodromy matrix

Certifying the irreducibility

Monodromy matrices

Rigorous arbitrary-precision computation in the SageMath `ore_algebra` package

Interval arithmetic

$[a; b]$ representing $t \in \mathbb{R} \Rightarrow t \in [a; b]$

Example: $[1; 2] \times [2; 3] \supset [2; 6]$

Difficulty: no exact zero-test

Example of interval Gaussian elimination:

$$\begin{pmatrix} [3.1;3.2] & [3.3;3.4] & [1.7;1.8] \\ [2.4;2.5] & [3.5;3.6] & [2.1;2.2] \\ [-0.7;-0.6] & [1.2;1.3] & [1.2;1.3] \end{pmatrix}$$

Certifying the irreducibility

Monodromy matrices

Rigorous arbitrary-precision computation in the SageMath `ore_algebra` package

Interval arithmetic

$[a; b]$ representing $t \in \mathbb{R} \Rightarrow t \in [a; b]$
Example: $[1; 2] \times [2; 3] \supset [2; 6]$
Difficulty: no exact zero-test

Example of interval Gaussian elimination:

$$\begin{pmatrix} \mathbf{1} & [1.0,1.1] & [0.5;0.6] \\ [2.4;2.5] & [3.5;3.6] & [2.1;2.2] \\ [-0.7;-0.6] & [1.2;1.3] & [1.2;1.3] \end{pmatrix}$$

$r_1 \leftarrow r_1/c_{11}$
with $c_{11} \in [3.1;3.2]$
the exact *unknown* value

Certifying the irreducibility

Monodromy matrices

Rigorous arbitrary-precision computation in the SageMath `ore_algebra` package

Interval arithmetic

$[a; b]$ representing $t \in \mathbb{R} \Rightarrow t \in [a; b]$

Example: $[1; 2] \times [2; 3] \supset [2; 6]$

Difficulty: no exact zero-test

Example of interval Gaussian elimination:

$$\begin{pmatrix} \mathbf{1} & [1.0,1.1] & [0.5;0.6] \\ \mathbf{0} & [0.7;1.2] & [0.6;1.0] \\ \mathbf{0} & [1.8;2.1] & [1.5;1.8] \end{pmatrix} \qquad \begin{array}{c} r_1 \leftarrow r_1/c_{11} \\ r_2 \leftarrow r_2 - c_{21} r_1 \\ r_3 \leftarrow r_3 - c_{31} r_1 \end{array}$$

Certifying the irreducibility

Monodromy matrices

Rigorous arbitrary-precision computation in the SageMath `ore_algebra` package

Interval arithmetic

$[a; b]$ representing $t \in \mathbb{R} \Rightarrow t \in [a; b]$
Example: $[1; 2] \times [2; 3] \supset [2; 6]$
Difficulty: no exact zero-test

Example of interval Gaussian elimination:

$$\begin{pmatrix} \mathbf{1} & [1.0,1.1] & [0.5;0.6] \\ \mathbf{0} & \mathbf{1} & [0.5;1.5] \\ \mathbf{0} & \mathbf{0} & [-1.7;0.9] \end{pmatrix} \qquad \begin{array}{c} r_2 \leftarrow r_2 / c_{22} \\ r_3 \leftarrow r_3 - c_{32} r_2 \end{array}$$

Certifying the irreducibility

Monodromy matrices

Rigorous arbitrary-precision computation in the SageMath `ore_algebra` package

Interval arithmetic

$[a;b]$ representing $t \in \mathbb{R} \Rightarrow t \in [a;b]$

Example: $[1;2] \times [2;3] \supset [2;6]$

Difficulty: no exact zero-test

Example of interval Gaussian elimination:

$$\begin{pmatrix} \mathbf{1} & [1.0,1.1] & [0.5;0.6] \\ \mathbf{0} & \mathbf{1} & [0.5;1.5] \\ \mathbf{0} & \mathbf{0} & \color{red}{[-1.7;0.9]} \end{pmatrix} \qquad \begin{array}{c} r_2 \leftarrow r_2/c_{22} \\ r_3 \leftarrow r_3 - c_{32} r_2 \end{array}$$

Remark 1: number of “interval pivots” $\leqslant$ rank of the exact matrix

Remark 2: monotonicity + convergence (as the working precision $\to \infty$)

Conclusion: if $\mathcal{A}v = \mathbb{C}^r$, this can be certified by interval computation

Reconstructing a right-hand factor from a seed vector v

Proposition: $\mathcal{A}v \simeq \mathrm{Sol}(P_f)$ where $P_f :=$ minimal annihilator of f

Hermite–Padé approximants

Given $f, f', \ldots, f^{(r-1)} \in \overline{\mathbb{K}}[[x]]$, find $p_0, \ldots, p_{r-1} \in \overline{\mathbb{K}}[x]$ of degree $\leqslant d$ such that $\sum_{i=0}^{r-1} p_i f^{(i)} = o(x^\sigma)$. $(\sigma < r(d+1) \Rightarrow$ nonzero solutions$)$

- if we find a candidate right-hand factor (using LLL algorithm to guess coefficients in $\overline{\mathbb{K}}$), we can check it by Euclidean division in $\overline{\mathbb{K}}(x)\langle\partial\rangle$

Reconstructing a right-hand factor from a seed vector v

Proposition: $\mathcal{A}v \simeq \mathrm{Sol}(P_f)$ where $P_f :=$ minimal annihilator of f

Hermite–Padé approximants

Given $f, f', \ldots, f^{(r-1)} \in \overline{\mathbb{K}}[[x]]$, find $p_0, \ldots, p_{r-1} \in \overline{\mathbb{K}}[x]$ of degree $\leqslant d$ such that $\sum_{i=0}^{r-1} p_i f^{(i)} = o(x^\sigma)$. $\quad (\sigma < r(d+1) \Rightarrow$ nonzero solutions$)$

- if we find a candidate right-hand factor (using LLL algorithm to guess coefficients in $\overline{\mathbb{K}}$), we can check it by Euclidean division in $\overline{\mathbb{K}}(x)\langle\partial\rangle$
- irreducible case? Two approaches:
  - **either** Hermite–Padé approximants to certify that $\mathrm{order}(P_f) = r$
    price: computations at a large order of truncation
    ($\sigma > rB$ where B is a degree bound for right-hand factors [van Hoeij, 1997] [Bostan, Rivoal, Salvy, 2022])
  - **or** orbit computation to certify that $\mathcal{A}v = \mathbb{C}^r$
    price: a complete set of generators of $\mathcal{A}$ is *a priori* needed

Algorithm RightFactor(P) (this work)

Input: $P \in \mathbb{K}(x)\langle\partial\rangle$ of order r
Output: `'Irreducible'` or a nontrivial right-hand factor of P

1. Loop over working precision p and order of truncation σ until success:
2.   **While** a complete set of generators of $\mathcal{A}$ has not been computed:
3.   Compute a new local monodromy matrix
4.   Choose randomly a combination M of the computed generators
5.   Try Norton's criterion or another method depending on the spectrum of M
6.   If the previous step succeeds then return the result

Lemma [Eberly, 1991]
The set $\{M \in \mathcal{A}$ without simple eigenvalue$\}$ is an algebraic subset of $\mathcal{A}$.

Remark: Local-to-global method $\simeq$ 1st iteration of **While** loop

operator	order, degree	classic(1)	new(2)	
rand(3)	4, 20	11s	35s	local-to-global method applies
	6, 18	"incomplete" (1200s)	**770s**	
rand×rand(3) (different factors)	4, 20	11s	17s	
	6, 18	280s	39s	
rand(3)	4, 20	780s	1100s	local-to-global method does not apply
	6, 18	"incomplete" (3200s)	**1700s**	
rand×rand(3) (different factors)	4, 20	950s	**16s**	
	6, 18	"incomplete" (650s)	**120s**	
fcc5(4)	6, 17	62s	**2s**	combinatorial examples
fcc6(4)	8, 43	>36000s	**30s**	

(1) command `DFactor(_,'one step')` of the Maple package `DEtools`

(2) command `rfactor(_)` of my branch of the SageMath `ore_algebra` package

(3) random Fuchsian irreducible operators, created following [Ince, 1926]

(4) irreducible operators at http://koutschan.de/data/fcc1/ (probabilistic walks)

3) Algebraicity of the solutions

work in progress and discussions with M. Barkatou, T. Cluzeau, M. Mezzarobba, S. Yurkevich, F. Chyzak and J.-A. Weil

How to use the approximate monodromy matrices to test the algebraicity of a solution of a linear ODE?

The problem statement

Denote by $\mathcal{D} = K\langle\partial\rangle$ the algebra of linear differential operators, where $K = \mathbb{C}(x)$ and $\partial = \frac{d}{dx}$.

Consider an irreducible Fuchsian operator $L \in \mathcal{D}$ of order $r \geqslant 1$ such that 0 is not singular.

> *Given a solution $f \in \mathbb{C}[[x]]$ of L (by initial conditions), how to decide whether f is algebraic?*

Definition: f is algebraic if there is a nonzero $P \in \mathbb{C}[X, Y]$ such that $P(x, f(x)) = 0$.

The running example

In his paper *The Art of Algorithmic Guessing in* gfun (2022), S. Yurkevich shows the algebraicity of the solution f of

$$L_a = 1800x(7x-62)(x^2+50x+20)\partial^2 + (30240x^3+124560x^2-10245600x-446400)\partial + 604$$

with initial conditions $f(0)=1$ and $f'(0)=-48300$.

This example comes from a sequence studied in *Arithmetic and Topology of Differential Equations* [Zagier, 2016] (relationship between the zeta function of a variety defined over a number field and some periods).

History about the algebraicity of a solution of a linear ODE

- **[Schwarz, 1872]** classification of the 2nd order hypergeometric equations
- **[Klein, 1878]** reducting 2nd order equations to Schwarz's cases
- **[Jordan, 1878]** bound for algebraicity degree --> turning into an algorithm by **[Painlevé, 1887]** and **[Boulanger, 1898]**
- **[Fuchs, 1881]** and **[Pépin, 1881]** minimal polynomial for 2nd order

- **[Risch, 1970]** solution to **Abel's problem**: given u algebraic, decide whether $\exp\left(\int u\right)$ is algebraic
- **[Baldassari, Dwork, 1979]** turning [Klein, 1878] into an algorithm
- **[Singer, 1979]** complete decision procedure + computation of minimal polynomial

- **[Bostan, Kauers, 2010]**, [Yurkevich, 2022] ``Guess&Prove approach'' for concrete examples

Numerical approach? Not directly studied yet.

Let's see how to rediscover the algebraicity of L_a with the approximate monodromy matrices.

The Guess&Prove approach

1. Compute the first coefficients of f
2. Guess the minimal polynomial P (Hermite–Padé approximation)
3. Validate that f is indeed solution of P by unicity

Main issues:

- complexity because of the size of P (which can be much larger than L)
- cannot conclude in the case of nonalgebraicity

Therefore, the new question is:

> *How to decide whether all the solutions of L are algebraic?*

A key result

For the remainder of this talk: let us fix a basis of power series solutions $(f_1, \dots, f_r)$ of L at 0.

In this way, one will identify the solution space $V = \mathrm{Span}_{\mathbb{C}}(f_1, \dots, f_r)$ of L at 0 with $\mathbb{C}^r$.

Let $\mathcal{M} \subset \mathrm{GL}_r(\mathbb{C})$ be the matrix representation of the monodromy group of L with respect to the f_i's.

THEOREM: All the solutions of L are algebraic $\Leftrightarrow$ $\mathcal{M}$ is finite.

Brief summary for the remainder of this talk:

- 1st approach: Finiteness of the monodromy group
- 2nd approach: nullity of the Lie algebra thanks to a maximal decomposition of $\mathrm{End}_{\mathbb{C}}(\mathrm{Sol}(P))$
- 3rd approach: nullity of the Lie algebra thanks to the invariants

```
In [1]: Dops, x, Dx = DifferentialOperators(QQ, 'x')
        La = 1800*x*(7*x - 62)*(x^2 + 50*x + 20)*Dx^2 + (30240*x^3 + 124560*x^2 - 10245600*x - 446400)*Dx + 6048*x^2 - 139453*x - 249550
```

Shift ($x \to x + 1$) so that 0 is not singular

```
In [2]: dop = La.annihilator_of_composition(x+1)
        pretty_print(dop)
```

$$\left(x^4 + \frac{316}{7}x^3 - \frac{2054}{7}x^2 - \frac{6268}{7}x - \frac{3905}{7}\right)Dx^2 + \left(\frac{12}{5}x^3 + \frac{598}{35}x^2 - \frac{27516}{35}x - \frac{5854}{7}\right)Dx + \frac{12}{25}x^2 - \frac{127357}{12600}x - \frac{76591}{2520}$$

Computation of a family of generators of the monodromy group (with 200 digits of precision).

```
In [3]: from ore_algebra.analytic.monodromy import monodromy_matrices
        %time gen = monodromy_matrices(dop, 0, eps=1e-200)
```

```
CPU times: user 582 ms, sys: 7.78 ms, total: 590 ms
Wall time: 604 ms
```

In [1]:
```
Dops, x, Dx = DifferentialOperators(QQ, 'x')
La = 1800*x*(7*x - 62)*(x^2 + 50*x + 20)*Dx^2 + (30240*x^3 + 124560*x^2 - 10245600*x - 446400)*Dx + 6048*x^2 - 139453*x - 249550
```

Shift ($x \to x + 1$) so that 0 is not singular

In [2]:
```
dop = La.annihilator_of_composition(x+1)
pretty_print(dop)
```

$$\left(x^4 + \frac{316}{7}x^3 - \frac{2054}{7}x^2 - \frac{6268}{7}x - \frac{3905}{7}\right)Dx^2 + \left(\frac{12}{5}x^3 + \frac{598}{35}x^2 - \frac{27516}{35}x - \frac{5854}{7}\right)Dx + \frac{12}{25}x^2 - \frac{127357}{12600}x - \frac{76591}{2520}$$

Computation of a family of generators of the monodromy group (with 200 digits of precision).

In [3]:
```
from ore_algebra.analytic.monodromy import monodromy_matrices
%time gen = monodromy_matrices(dop, 0, eps=1e-200)
```

```
CPU times: user 582 ms, sys: 7.78 ms, total: 590 ms
Wall time: 604 ms
```

1st approach: Finiteness of the monodromy group

Let's observe that the monodromy group is finite.

```
In [4]: %run utilities.ipynb
        #%time M = generated_group(gen) # 10min
```

```
In [51]: len(gen), len(M)
```

```
Out[51]: (3, 600)
```

```
In [43]: customized_accuracy(M), RR(max(max(c.abs() for c in mat.list()) for mat in M).mid())
```

```
Out[43]: (644, 9.63461481942224)
```

Conclusion of this method:

- simple
- quite long (but here, the finiteness of the orbit of f under G is sufficient)
- not certified (although very convincing)

The Lie algebra of the differential Galois group

Denote by $\mathcal{G}$ the Zariski closure of $\mathcal{M}$. Note that $\mathcal{G}$ is finite $\Leftrightarrow$ $\mathcal{M}$ is finite.

THEOREM: $\mathcal{G}$ is finite $\Leftrightarrow$ the Lie algebra of $\mathcal{G}$ is zero.

Let $\mathcal{I} \subset \mathbb{C}[X_{1,1}, \ldots, X_{r,r}]$ be an ideal such that

$$\mathcal{G} = \{M \in \mathrm{GL}_r(\mathbb{C}) \mid \forall p \in \mathcal{I}, p(M) = 0\}.$$

Let ϵ be such that $\epsilon^2 = 0$.

Definition: The Lie algebra of $\mathcal{G}$ is $\mathfrak{g} = \{N \in \mathbb{C}^{r\times r} \mid \forall p \in \mathcal{I}, p(I_r + \epsilon N) = 0\}.$

Action of $\mathcal{G}$ on $\mathbb{C}^{r\times r} \simeq \mathrm{End}_{\mathbb{C}}(V)$ where $V = \mathrm{Sol}(L) \simeq \mathbb{C}^r$

Let $M \in \mathcal{G}$ and $N \in \mathbb{C}^{r\times r}$. Note that $\mathcal{G} \subset \mathrm{GL}_r(\mathbb{C}) \simeq \mathrm{GL}_{\mathbb{C}}(\mathbb{C}^r)$ and $\mathbb{C}^{r\times r} \simeq \mathrm{End}_{\mathbb{C}}(\mathbb{C}^r)$.

We require that the following diagram commutes.

$$\begin{array}{ccccc} & \mathbb{C}^r & \xrightarrow{N} & \mathbb{C}^r & \\ M & \downarrow & \circlearrowright & \downarrow & M \\ & \mathbb{C}^r & \dashrightarrow & \mathbb{C}^r & \end{array}$$

$$M \cdot N = MNM^{-1}$$

Linearization: $\mathrm{vect}(MNM^{-1}) = \left((M^{-1})^T \otimes M\right) \mathrm{vect}(N)$ where

- $\mathrm{vect}(A) = \left(C_1^T \mid \cdots \mid C_r^T\right)^T$ for $A = (C_1 \mid \cdots \mid C_r)$,
- $A \otimes B = (a_{i,j}B)_{i,j}$ with $A = (a_{i,j})_{i,j}$ for matrices A, B in $\mathbb{C}^{r\times r}$.

2nd approach: nullity of $\mathfrak{g}$ thanks to a maximal decomposition of $\mathrm{End}_{\mathbb{C}}(V) \simeq \mathbb{C}^{r\times r}$

(based on the article *Computing the Lie Algebra of the Differential Galois Group of a Linear Differential System* [Barkatou et al., 2016])

Lemma: $\mathfrak{g}$ is a $\mathcal{G}$-submodule of $\mathbb{C}^{r\times r}$.

Assumption: The $\mathcal{G}$-module $\mathbb{C}^{r\times r}$ admits a maximal decomposition:

$$\exists W_1, \ldots, W_t \subset \mathbb{C}^{r\times r} \text{ irreducible } G\text{-submodules such that } \mathbb{C}^{r\times r} = W_1 \oplus \cdots \oplus W_t.$$

Corollary: $\exists S \subset \{1, \ldots, t\}$ such that $\mathfrak{g} = \bigoplus_{i\in S} W_i$.

Algorithm:

1. Compute the W_i's of the maximal decomposition
2. Use the p-curvature to show that $\mathfrak{g} \cap W_i = \{0\}$ for all i (for a convenient prime p)

In [5]:
```
GEN = [mat.tensor_product(~mat.transpose()) for mat in gen]
```

The method of the eigenring

Let $\mathcal{H} \subset \mathrm{GL}_s(\mathbb{C})$ be such that the $\mathcal{H}$-module $\mathbb{C}^s$ admits a maximal decomposition.

In our case: $\mathbb{C}^s = \mathrm{vect}(\mathbb{C}^{r \times r})$ and $\mathcal{H} = \left\{(M^{-1})^T \otimes M; M \in \mathcal{G}\right\}$.

Definition: the eigenring of $\mathcal{H}$ is $\mathcal{E}(\mathcal{H}) = \mathrm{Centralizer}(\mathcal{H}) = \{M \in \mathbb{C}^{s \times s} \mid \forall N \in \mathcal{H}, MN = NM\}$.

Lemma: Let $M \in \mathcal{E}(\mathcal{H})$ be with a maximal number of eigenvalues. Then the generalized eigenspaces of M provide the maximal decomposition of the $\mathcal{H}$-module $\mathbb{C}^s$.

Notes on complexity:

- a system of generators of $\mathcal{H}$ is sufficient
- computing $\mathcal{E}(\mathcal{H})$ is reduced to a linear algebra problem in $\mathbb{C}^{r^4 \times r^4}$ (here $s = r^2$) $\rightarrow O(r^{4\omega})$

In [6]:
```
E = centralizer(GEN)
pretty_print(guess_rational_numbers(E, p=50))
```

$$\left[\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{pmatrix}, \begin{pmatrix}0 & 0 & 0 & 1\\ 0 & -1 & 0 & 0\\ 0 & 0 & -1 & 0\\ 1 & 0 & 0 & 0\end{pmatrix}\right]$$

In [7]:
```
decomp = gen_eigenspaces(E[1])
print([space["multiplicity"] for space in decomp], "\n")

basis1 = [guess_rational_numbers(b, p=50) for b in decomp[0]["basis"]]
basis2 = [guess_rational_numbers(b, p=50) for b in decomp[1]["basis"]]

pretty_print(basis1, basis2)
pretty_print([matrix(2,2,b) for b in basis1], [matrix(2,2,b) for b in basis2])
```

```
[3, 1]
```

$$[(0, 1, 0, 0), (0, 0, 1, 0), (1, 0, 0, -1)]\ [(1, 0, 0, 1)]$$

$$\left[\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}, \begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}, \begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}\right]\ \left[\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\right]$$

We observe that the maximal decomposition of $\mathrm{End}_{\mathbb{C}}(V)$ is given by

$$\mathrm{End}_{\mathbb{C}}(V) = \mathbb{C}I_2 \oplus W$$

where $W = \mathrm{Span}_{\mathbb{C}}(u1, u2, u3)$

with $u_1 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$, $u_2 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$ and $u_3 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$.

Therefore $\mathfrak{g}$ is one of the following subspaces: $\{0\}, \mathbb{C}I_2, W$ or $\mathrm{End}_{\mathbb{C}}(V)$.

$\rightarrow$ ensure that the intersection of $\mathfrak{g}$ with each block is zero thanks to the p-curvature.

Conclusion:

- computation of the eigenring is costly
- not certified?

3rd approach: nullity of $\mathfrak{g}$ thanks to the invariants

(based on the work of Aparicio-Monforte, Compoint, Singer, Ulmer, van Hoeij, Weil... since the 90s)

THEOREM:

$$\mathfrak{g} = \{N \in \mathbb{C}^{r\times r} \mid \forall v \in \mathrm{Inv},\ N \cdot v = 0\}$$

where

$$\mathrm{Inv} = \bigcup_{h\in\mathbb{N}} \left\{v \in \mathrm{Sym}^h(\mathbb{C}^r) \mid \forall M \in \mathcal{G},\ M \cdot v = v\right\}$$

Definition: The h-th symmetric power of a $\mathbb{C}$-space U is $\mathrm{Sym}^h(U) = (U \otimes \cdots \otimes U)/S_h$ where S_h stands for the permutation group on h elements.

Remark: if U is a $\mathbb{C}$-algebra, then U^h is the quotient of $\mathrm{Sym}^h(U)$ by the homogeneous algebraic relations of degree h between the elements of U.

But, how to define the actions of $\mathcal{G}$ and $\mathfrak{g}$ on $\mathrm{Sym}^h(\mathbb{C}^r)$?

Action of $\mathcal{G}$ and $\mathfrak{g}$ on the constructions (dual, $\oplus$, $\otimes$, $\mathrm{Hom}_{\mathbb{C}}$, $\dots$)

Already discussed example: action on $\mathrm{End}_{\mathbb{C}}(\mathbb{C}^r)$

	$v \in \mathbb{C}^r$	$v \in \mathrm{End}_{\mathbb{C}}(\mathbb{C}^r)$	$\overset{\text{vect}}{\simeq}$	$v \in \mathbb{C}^{r^2}$
$M \in \mathcal{G}$	$M \cdot v = Mv$	$M \cdot v = MvM^{-1}$		$M \cdot v = \left((M^{-1})^T \otimes M\right) v$

On symmetric powers

	$v \in \mathbb{C}^r$	$v = v_1 \otimes \cdots \otimes v_h \in \mathrm{Sym}^h(\mathbb{C}^r)$	$\simeq$	$v \in \mathbb{C}^s$
$M \in \mathcal{G}$	$M \cdot v = Mv$	$M \cdot v = (Mv_1) \otimes \cdots \otimes (Mv_h)$		$M \cdot v = \mathrm{SymPowGroup}(M, h)\, v$
$N \in \mathfrak{g}$	$N \cdot v = Nv$	so that $(I + \epsilon N) \cdot v = v + \epsilon\,(N \cdot v)$		$N \cdot v = \mathrm{SymPowAlg}(N, h)\, v$

Remark: $\mathrm{SymPowGroup}(I_r + \epsilon N, h) = I_s + \epsilon\, \mathrm{SymPowAlg}(N, h)$

Example: $\mathrm{Sym}^2(\mathrm{Span}_{\mathbb{C}}(f_1, f_2)) = \mathrm{Span}_{\mathbb{C}}(f_1 \otimes f_1, f_1 \otimes f_2, f_2 \otimes f_2)$.

Let $M = \begin{pmatrix} m_{0,0} & m_{0,1} \\ m_{1,0} & m_{1,1} \end{pmatrix}$ and $N = \begin{pmatrix} n_{0,0} & n_{0,1} \\ n_{1,0} & n_{1,1} \end{pmatrix}$. Then:

$$\mathrm{SymPowGroup}(M, 2) = \begin{pmatrix} m_{0,0}^2 & m_{0,0}m_{0,1} & m_{0,1}^2 \\ 2m_{0,0}m_{1,0} & m_{0,1}m_{1,0} + m_{0,0}m_{1,1} & 2m_{0,1}m_{1,1} \\ m_{1,0}^2 & m_{1,0}m_{1,1} & m_{1,1}^2 \end{pmatrix}$$

$$\mathrm{SymPowAlg}(N, 2) = \begin{pmatrix} 2n_{0,0} & n_{0,1} & 0 \\ 2n_{1,0} & n_{0,0} + n_{1,1} & 2n_{0,1} \\ 0 & n_{1,0} & 2n_{1,1} \end{pmatrix}$$

such that $\mathrm{SymPowGroup}(I_2 + \epsilon N, 2) = I_3 + \epsilon\,\mathrm{SymPowAlg}(N, 2)$.

The algorithm for ensuring the algebraicity thanks to the approximate monodromy matrices

1. Approximate a family of generators for the monodromy group
2. Approximate some v in Inv
3. Guess exact $v \in \mathrm{Inv}$ and validate them
4. Compute $\mathfrak{g}_{\text{guess}} = \{N \in \mathbb{C}^{r \times r} \mid \text{for all validated } v,\, N \cdot v = 0\}$
5. If $\mathfrak{g}_{\text{guess}} = \{0\}$ then return OK, else go back to the beginning with better precision and more v

Step 2: Approximate an invariant

In [8]:

```
%run tools.ipynb

%time gen = monodromy_matrices(dop, 0, eps=1e-400)

#SPG = SymPowGroup(M, 20) # 5min (but D&C could solve this)
SPG = load("SymPowGroup(2×2,20).sobj")
C = gen[0].base_ring()
GEN = [SPG(m00=m[0,0], m01=m[0,1], m10=m[1,0], m11=m[1,1]).change_ring(C) for m in gen]
Id = GEN[0].parent().one()
inv = optimistic_intersection([ker(M-Id) for M in GEN]).basis_matrix()
inv.dimensions()
```

```
CPU times: user 936 ms, sys: 240 µs, total: 936 ms
Wall time: 936 ms
```

Out[8]:

```
(1, 21)
```

Step 3: Guess the exact invariant v then validate it

In [9]:
```
v = guess_rational_numbers(inv)[0]
v
```

Out[9]:
```
(1, 1218173190271906037987/59853669753670090231, -16000909106616073775487/40259468392205460692220, 30606065460742048671192107/
10719083459424703909303575, -927111471066477929916800494721/730612728594387818458131672000, 34014898175155115890757765640193/8
1052349578439898610198982362500, -281415960792583675458321224389070779/2486037666269908570172023187022600000, 6773534490470089
7920238688601121343467/2647630114577452627233204694179069000000, -431081993517033751032627633343251168242791/90231234304799585
5361076159776226715200000000, 309963288584603616422783752033591472316837929/432433190405752013681795749572756653259600000000, -
517524720244692049552025119199928826513244719933/6140551303761678594281499643933144476286320000000000, 15134145835566052541795
574417250591436135438588491/1961906141551856310872939136236639660173479240000000000, -3844353781889927598368691493806286567978
4812790654291/6686176130408726307454976576294467961871217249920000000000, 180739806706247860142095862355864518508386990297261
159/44504859868033084483997187835960052371205289819780000000000000, -18002247205602319460923272577591205422930108491432366853/5
617502312231731552646756153516735499298801026141120000000000000, 623415040061319078937207693863870400856146110402254990225239/
2422969184823351611945362097915605939235055352600318584000000000000000, -56177135155547857295113843526127155687110429004006801
0527740469/33029915927511929174039176118785540163652274566647542937088000000000000000, 145243625046003418009858209268015433172
11201797728602367291428711/1758843023140010228517586128325330013714483620673981661399936000000000000000000, -3769769061443617108
0630201746147872738915321729326795083765882844943/13486808301437598432272850431998630545162660403328091379614709248000000000000
000000, 171923398908917724964166185228943011983212340712546485142086267138059/284276631228739379455251175511971134459756701313
89992611094129336800000000000000000000, -757430020442684178543353386726025727400060692767243109528357452276497001/12237660116558
4480654757390249869173840696947967718435560347948774502400000000000000000000)
```

Definition: Let $h \geqslant 1$ and $P \in \mathcal{D}$ be an operator whose solution space is $\mathrm{Span}_{\mathbb{C}}(f_1, \dots, f_r)$.
The hth symmetric power of P is the (monic) operator whose solution space is
$\{p(f_1, \dots, f_r)\ ;\ p \in \mathbb{C}[X_1, \dots, X_r] \text{ homogeneous of degree } h\}$.

Let denote by DOP the 20th symmetric power of dop.

Our invariant must correspond to a rational solution DOP.

In [10]:
```
sol = dop.power_series_solutions(200); sol.reverse()
f0, f1 = sol
B = vector([ f0^(20 - i)*f1^i for i in range(21) ])
f = B*v

p, q = hp_approximants([f.parent().one(), -f], 190)
f = p/q
pretty_print(f)
```

$$\frac{-\frac{130936577510053912869639540403054362624 1}{6145538767665562785 69}x^{12} + \frac{116330920534728222694727728803313881993386}{2048512922555518759523}x^{11} + \frac{66725694153489546609825143438794051554 48164}{614553876766556278569}x^{10} + \frac{53224648445751279934703505399871543017268930}{614553876766556278569}x^{9} + \frac{80809952944442863906980431368727982303336385}{2048512922555518759523}x^{8} + \frac{70907331091551453036495876767311923775179642 8}{614553876766556278569}x^{7} + \frac{140560205972220827192794023327413383472780201 6}{614553876766556278569}x^{6} + \frac{21443254345863118749534683904798133317984845 2}{68283764085172919841}x^{5} + \frac{67876740670119973389249445621880451231388015}{227612546950576399 47}x^{4} + \frac{43478422355577713505620940932642340974293090}{22761254695057639947}x^{3} + \frac{19677074488216107639554911939026303018059 08}{2529028299450848883}x^{2} + \frac{14973378001490854254460761110436284149620 2}{843009433150282961}x + 16409682740640811134241}{x^{24} + 624x^{23} + 179316x^{22} + 31421104x^{21} + 3746292066x^{20} + 321061273104x^{19} + 20364339690596x^{18} + 969540805840464x^{17} + 34763279455387791x^{16} + 933444208818900064x^{15} + 18510838203706894056x^{14} + 265385932325240504544x^{13} + 2683320137949763243996x^{12} + 18842401195092075822624x^{11} + 93313135384886452936296x^{10} + 334089950222581340806304x^{9} + 883393368034168276186671x^{8} + 1749273978889557372258864x^{7} + 2608677696229036893706916x^{6} + 2920090857086868588215664x^{5} + 2419181330697477530432226x^{4} + 1440610509318461721750224x^{3} + 583717252592887857438516x^{2} + 144220310283941776729104x + 16409682740640811134241}$$

In [11]:
```
DOP = dop.symmetric_power(20)
DOP(f)
```

Out[11]: 0

Step 4: Compute $\mathfrak{g}_{\text{guess}}$

In [12]:
```
var('n00,n01,n10,n11')
%time N = SymPowAlg(matrix(2,2,[n00,n01,n10,n11]), 20)
eqs = (N*v).list()
solve(eqs, n00,n01,n10,n11)
```

```
CPU times: user 1.95 s, sys: 20 ms, total: 1.97 s
Wall time: 1.97 s
```

Out[12]:
```
[[n00 == 0, n01 == 0, n10 == 0, n11 == 0]]
```

Step 5: Conclusion

We get a certification that the Lie algebra of the Galois group of L_a is zero, *i.e.* all the solutions of L_a are algebraic.

?

Back to the validation of the invariant

DOP is something we don't want to compute (too big).

> *How to validate an invariant without computing DOP?*

THEOREM [Bertrand, Beukers, 1985]. Let $L \in \mathcal{D}$ with basis of power series solution $(f_1, \ldots, f_r)$. Given $f = p(f_1, \ldots, f_r)$ (with $p \in \mathbb{C}[X_1, \ldots, X_r]$ homogeneous of degree h) which seems to correspond with a rational fraction $\frac{n}{d} \in \mathbb{C}(x)$, we can compute a bound σ such that:

- either $f = \frac{n}{d}$,
- or $\mathrm{valuation}_x(df - n) \leqslant \sigma$.

Here, $\sigma = 1730$ is sufficient.

In [13]:

```
PS = PowerSeriesRing(QQ, 'x', 1750)
f = PS(f)

t0 = cputime()
sol = dop.power_series_solutions(1750); sol.reverse()
f0, f1 = sol[0], sol[1]
powers_of_f0, powers_of_f1 = [1], [1]
for i in range(20):
    powers_of_f0.append(powers_of_f0[-1]*f0)
    powers_of_f1.append(powers_of_f1[-1]*f1)
B = vector([ powers_of_f0[20 - i]*powers_of_f1[i] for i in range(21) ])
print(cputime(t0))
f - PS(B*v)
```

```
24.927034999999997
```

Out[13]:

```
O(x^1750)
```

Conclusion

Contributions:

- link between local-to-global method and Norton's criterion ($\rightarrow$ incremental hybrid algorithm)
- implementation of this algorithm in SageMath (GNU GPL) (https://github.com/a-goyer/ore_algebra/tree/facto)
- promising results on algebraicity with numerics (to be continued)

Future work:

- complexity analysis
- extension to the non-Fuchsian case

Thank you for listening.