Universidad de Valladolid

FACULTAD DE CIENCIAS

DEPARTAMENTO DE ÁLGEBRA,
ANÁLISIS MATEMÁTICO, GEOMETRÍA Y TOPOLOGÍA

TESIS DOCTORAL:

Métodos numérico-simbólicos para calcular soluciones liouvillianas de ecuaciones diferenciales lineales

Presentada por Alberto Llorente Mediavilla para optar al grado de doctor por la Universidad de Valladolid

Dirigida por:
Jorge Mozo Fernández

Numeric-symbolic methods for computing Liouvillian solutions of linear differential equations

Alberto Llorente

January 15, 2014

Contents

Resumen en castellano

Introducción

Uno de los problemas centrales de las matemáticas es la resolución de ecuaciones. Éstas pueden ser de muy diversos tipos: algebraicas, diferenciales, funcionales, etc. Asimismo, en cada uno de estos tipos podemos distinguir varias clases. Así, por ejemplo, una ecuación diferencial puede ser ordinaria, en derivadas parciales, lineal, etc.

La primera pregunta que cabe hacerse al enfrentarse con este problema es la siguiente: ¿qué entendemos por resolver una ecuación? La respuesta no es trivial ni es única; depende del contexto en que trabajemos. Así, por ejemplo, resolver una ecuación puede interpretarse de las siguientes formas:

1. Demostrar la existencia de una solución. Para ello es preciso definir con precisión qué entendemos por solución o, lo que viene a ser un problema equivalente, en qué espacio vamos a buscar las soluciones.

2. Describir de manera explícita una solución. De nuevo, es importante precisar lo que entendemos por solución explícita. Por el momento nos quedaremos con la idea intuitiva de que se trata de una solución que se puede obtener en un número finito de pasos a partir de objetos conocidos.

3. Aproximar una solución. En ocasiones puede ocurrir que la descripción explícita de una solución sea imposible o, al menos, muy laboriosa. Para efectos prácticos puede ser suficiente una aproximación numérica o en otros términos, por ejemplo, la truncación de una serie solución.

Según el contexto, y eventualmente el problema, cualquiera de las nociones de solución anteriores puede ser válida. Estas tres nociones están mencionadas en su orden lógico, pero históricamente surgieron en orden inverso. Ya en Mesopotamia

y Egipto había interés por resolver problemas que en términos modernos equivaldrían a la resolución de ecuaciones polinómicas de primer y segundo grado, pero carecían, según [BM91, p. 41], "de una distinción clara entre resultados exactos y aproximados". Los matemáticos griegos dedicaron importantes esfuerzos a la resolución de problemas geométricos, lo que para ellos significaba[1] *constructibilidad* por medio de regla y compás. En términos modernos, estos instrumentos son capaces de hacer construcciones equivalentes a resolver una cadena de ecuaciones de primer y segundo grado.

Introducido el lenguaje algebraico en el Renacimiento, el interés por resolver ecuaciones de grados superiores se incrementó, llegando a resolver las ecuaciones de tercer y cuarto grado. En los métodos de resolución, estas soluciones aparecen descritas en términos de radicales, involucrando eventualmente raíces de índice par de números negativos, las cuales fueron introducidas por Cardano en su *Ars Magna* [Car1545], no sin cierta controversia. Supone un avance conceptual destacado el hecho de considerar la existencia de soluciones (complejas) en espacios distintos del de los coeficientes de las ecuaciones (números reales), comparable sólo a la introducción de los irracionales tras la demostración de la irracionalidad de las raíces cuadradas de los números primos.

Dos hitos notables en el capítulo de la solución de las ecuaciones algebraicas son la demostración del Teorema Fundamental del Álgebra por Gauss[2] y la de la imposibilidad de construir soluciones de la quíntica por Abel.[3] El primer resultado es de índole existencial; el segundo es de naturaleza constructiva: establece la imposibilidad de la existencia de una construcción explícita finita, en términos de radicales, que permita resolver la ecuación general de quinto grado. Destaquemos aquí que, para generalizar este resultado, resulta básica la construcción de Galois[4] del grupo de simetrías asociado a una ecuación algebraica. Tales garantía de existencia e imposibilidad de construcción por radicales abren la puerta a los métodos aproximativos para las ecuaciones de grado 5 en adelante. En §2.1.4 se considera

[1] Ésta es la llamada geometría *plana* de los griegos; cfr. [PH1876, lib. III, prop. 4, p. 55].

[2] K. F. Gauss dio la primera demostración válida del Teorema Fundamental del Álgebra en su tesis doctoral [Gau1799], con el descriptivo título "nueva demostración del teorema [que afirma que] toda función algebraica racional entera [i.e., polinomio] de una variable se puede descomponer en factores reales de primer o segundo grado". Todavía publicaría otras tres demostraciones más de este teorema.

[3] N. H. Abel publicó la imposibilidad de resolver por radicales la ecuación general de grado 5 en la resumida memoria [Abe1824] "sobre las ecuaciones algebraicas donde se demuestra la imposibilidad de la resolución de la ecuación general de quinto grado". Posteriormente publicaría versiones más amplias.

[4] É. Galois remitió su memoria "sobre las condiciones de resolubilidad de las ecuaciones por radicales" a la Academia de Ciencias de París, pero fue rechazada. Varios años después de su prematura muerte, la publicó Liouville en [Gal1846].

varios tales métodos de aproximación de raíces de polinomios.

Nos ocupamos ahora del terreno de las ecuaciones diferenciales. De nuevo, las tres nociones de solución anteriormente citadas tienen sentido. Consideremos, por ejemplo, la ecuación diferencial más simple, que no es más que $y' = f$, donde f es una función expresable en términos conocidos. Como bien sabe todo estudiante de primer año, la continuidad de la función f es suficiente para garantizar la existencia de una solución de dicha ecuación, que será de la forma

$$y(x) = \int_{x_0}^{x} f(t)\,\mathrm{d}t,$$

tomada la integral anterior en el sentido de Riemann. (Extensiones de esta integral permiten abordar el problema para funciones f más generales.) Esto responde a la noción existencial de la solución, pero, incluso en el caso de funciones f sencillas (expresadas en términos elementales[5]), la búsqueda de una primitiva en términos finitos se convierte en un problema altamente no trivial.

En 1833, Liouville presenta la memoria sobre "la integración de una clase de funciones trascendentes", publicada en [Lio1835], en la que determina para qué funciones, compuestas de algebraicas y primitivas de algebraicas, se puede obtener una primitiva expresable en términos finitos: «*Si P es una función algebraica de x e $y_1, \ldots, y_m$, con y_i primitiva de algebraica de x e $y_1, \ldots, y_m$, y P admite una primitiva en términos finitos (composición de funciones algebraicas, exponenciales y logaritmos) de x e $y_1, \ldots, y_m$, entonces la primitiva de P (respecto a x) es combinación lineal de funciones algebraicas y logaritmos de algebraicas de x e $y_1, \ldots, y_m$.*» Como él mismo indica en [Lio1835, p. 94], este teorema "debe ser considerado como fundamental en la teoría de funciones de una variable". Aplica Liouville este resultado a probar que "la integral $\int \mathrm{e}^x/x\,\mathrm{d}x$, la cual ha interesado enormemente a los geómetras, no es expresable en términos finitos"; cf. [Lio1835, §VIII]. Asimismo, Liouville muestra que la ecuación diferencial lineal $y'' - y = -1/x$ no admite solución de este tipo. Los resultados anteriores resultan ser la base del algoritmo de Risch, presentado en 1968 [Ris68, Ris69, Ris70], para la determinación de si una función elemental admite una primitiva elemental y calcularla si la admite.

Con respecto a ecuaciones diferenciales lineales de forma más compleja que $y' = f$, Liouville mismo se plantea, entre otros problemas, la búsqueda de soluciones algebraicas. Para una ecuación de primer orden, el problema no es más complejo que el del cálculo de primitivas, expuesto anteriormente. Con respecto a la ecuación

[5] Una función elemental es, explicado informalmente, composición de algebraicas, exponenciales y logaritmos. Esta noción se puede formalizar de manera similar a como se hace con la noción de función liouvilliana en §1.1.3. La terminología de funciones *elementales* es de Ritt, cfr. [Rit48], quien traduce los resultados de Liouville a lenguaje algebraico; Liouville se refiere a ellas como "en términos finitos".

de segundo orden, H. Schwarz determina en 1871, publicado en [Sch1872], criterios para que la ecuación hipergeométrica de Gauss

$$x(x-1)y'' + (c-(a+b+1)x)y' - aby = 0$$

tenga soluciones algebraicas, en términos de a, b y c. Otros resultados de esta índole fueron obtenidos, y un estudio sistemático del problema llegó con la teoría de Galois diferencial, desarrollada por Ritt, Kolchin, Kaplansky y Ramis, entre otros autores. Imitando la teoría de Galois clásica de las ecuaciones algebraicas, se construye un grupo de Galois diferencial asociado a una ecuación diferencial lineal. Este grupo resulta ser un grupo algebraico de matrices. La existencia de un sistema fundamental de soluciones expresables en un número finito de extensiones algebraicas, cuadraturas y exponenciales de cuadraturas (i.e., soluciones liouvillianas) resulta ser equivalente a que la componente de la identidad (la componente conexa en la topología de Zariski donde está la matriz identidad) del grupo de Galois sea resoluble. En términos matriciales, esto equivale a que dicha componente sea conjugada a un subgrupo de matrices triangulares.

Empleando estas técnicas, J. Kovacic obtiene en 1979 (y publica en [Kov86]) el primer algoritmo completo para determinar la existencia de soluciones liouvillianas de una ecuación diferencial lineal de segundo orden sobre las funciones racionales. El algoritmo se basa en el estudio fino de los subgrupos algebraicos de $\mathrm{SL}(2,\mathbb{C})$ y en analizar todos los posibles casos en un orden particular. En 1996 [UW96] F. Ulmer y J.-A. Weil publican una alternativa al algoritmo de Kovacic mediante el uso de potencias simétricas de los operadores diferenciales. En 1981 M. Singer publica [Sin81] un algoritmo general para orden arbitrario, pero basado en una cota de Jordan que lo hace impracticable. Para las ecuaciones de orden 3, Singer y Ulmer desarrollaron algoritmos específicos [SU93a, SU93b], y O. Cormier [Cor01] abrió el camino para órdenes 4 y 5, pero también obtuvo que un algoritmo similar para orden 6 requeriría trabajar con una ecuación auxiliar de orden mayor que 10^{15} incluso cuando no hay solución liouvilliana. Estos algoritmos son de naturaleza simbólica y su complejidad aumenta terriblemente con el orden de la ecuación, resultando impracticables del orden sexto en adelante.

Otra vía para tratar de determinar la existencia de soluciones liouvillianas pasa por el cómputo del grupo de Galois diferencial de la ecuación. El único algoritmo en esta línea publicado hasta la fecha es el de E. Hrushovski [Hru02], si bien es oscuro y difícil de entender e implementar. Citemos que M. Singer y R. Feng tratan en la actualidad de revisar el trabajo de Hrushovski con el objetivo de lograr un algoritmo más comprensible.

Abandonando el cálculo simbólico, no es fácil obtener resultados de carácter numérico, debido a la sensibilidad del grupo de Galois frente a pequeñas variaciones. No obstante, J. van der Hoeven expone en [vdH07a] un cálculo híbrido

numérico-simbólico, que incorpora técnicas de aproximaciones numéricas pero con precisión que puede ser arbitraria, los números complejos efectivos, que desarrolla en trabajos anteriores. Como define en [vdH06a], un número $x \in \mathbb{R}$ es efectivo si está dotado de un algoritmo con entrada $\varepsilon \in \mathbb{Z}_{>0}2^{\mathbb{Z}}$ y salida $x_\varepsilon \in \mathbb{Z}2^{\mathbb{Z}}$ de manera que $|x_\varepsilon - x| < \varepsilon$. Un argumento de cardinalidad muestra que casi todo número real es no efectivo, pues los números efectivos son numerables, aunque por definición es difícil encontrar un número que no lo sea. El lector versado podrá relacionar la efectividad de un número real con la complejidad de Kolmogorov.

Se definen de manera análoga números complejos efectivos, funciones holomorfas efectivas, etc.; cfr. [vdH05]. En una serie de artículos, J. van der Hoeven demuestra la efectividad de la prolongación analítica de una solución holomorfa de una ecuación diferencial sobre las funciones racionales evitando las singularidades en [vdH99] y, extendiendo la noción de prolongación analítica, en singularidades regulares en [vdH01] y en singularidades irregulares en [vdH07b]; y, lo que resulta más relevante para nuestro estudio, realiza la construcción efectiva del grupo de Galois de una ecuación diferencial sobre las funciones racionales basada en el teorema de densidad de J.-P. Ramis, que da tres tipos de generadores del grupo de Galois diferencial como grupo algebraico: la monodromía formal, el toro exponencial y los automorfismos de Stokes. En [vdH07a], J. van der Hoeven construye los generadores de Ramis como matrices de números complejos efectivos y los utiliza para dar un algoritmo numérico-simbólico de factorización de operadores diferenciales sobre las funciones racionales, es decir, descomponer una ecuación diferencial lineal $L[y] = 0$ como $L_1[L_2[y]] = 0$.

Es en este marco en el que se encuadra la presente tesis. El objetivo es presentar una serie de técnicas de tipo algorítmico que permitan decidir si una ecuación diferencial lineal sobre $\mathbb{C}(x)$ admite o no una solución liouvilliana, encontrando una en caso afirmativo. Obsérvese que la noción general de función liouvilliana, construida a partir de una cadena de extensiones simples, no es sencilla de manejar. El propio Liouville, en su memoria sobre "la integración de una clase de ecuaciones diferenciales de segundo orden en cantidades finitas explícitas", publicada en [Lio1839], muestra que, si una ecuación diferencial del tipo $y'' = P(x)y$, con $P(x)$ un polinomio, admite "una integral expresable en función finita explícita de x", entoces habrá una solución de la forma $y = \mathrm{e}^{\int u}$, donde u es una función algebraica[6] determinada por la ecuación de Riccati $u' + u^2 = P(x)$. Aunque en principio Liouville sólo se ocupa de las integrales en términos de funciones elementales, en [Lio1839, §18] afirma que todo funciona igual si se añaden cuadraturas,

[6]Esto es válido en general para $P(x)$ racional. En el caso particular de $P(x)$ polinomio, Liouville obtiene que u es racional, e incluso suma de un polinomio y la derivada logarítmica de otro polinomio.

llegando a la noción actual de integral liouvilliana, llamada así[7] en su honor. En general, si una ecuación diferencial lineal admite una solución liouvilliana, de hecho admite una solución y con y'/y algebraica sobre el cuerpo de coeficientes de la ecuación. Este resultado es la base de los distintos algoritmos simbólicos antes mencionados. Vessiot dio una versión errónea en [Ves1892, p. 245], corregida por Kolchin en [Kol48]. Más aún, Singer prueba que existe una función aritmética I tal que, si una ecuación diferencial lineal de orden n admite una solución liouvilliana, entonces admite una solución y con y'/y algebraica sobre el cuerpo de coeficientes de la ecuación de grado a lo sumo $I(n)$. Llamaremos *singerianas* a estas soluciones. Las técnicas que desarrollamos persiguen encontrar una solución singeriana, si la hay, o bien decidir que no es el caso.

La idea general es, pues, la siguiente. En el marco del cálculo efectivo, se construyen los generadores del grupo de Galois de la ecuación diferencial, haciendo uso de alguna de las técnicas desarrollada por J. van der Hoeven. El grupo algebraico que generan es muy sensible a pequeñas variaciones de las entradas de los generadores. Por ejemplo, el grupo algebraico generado por $\lambda \in \mathbb{C}^*$ es finito si λ es una raíz de la unidad, pero es todo $\mathbb{C}^*$ si λ no lo es. Por esta razón construimos un grupo más grande que el de Galois, su *cierre eurimérico*, que resulta más fácil de computar en el marco del cálculo efectivo. Dicho grupo conserva algunas propiedades interesantes del grupo de Galois; en particular, si existe una recta invariante por la componente de la identidad del grupo de Galois, entonces esta misma recta es invariante por la componente de la identidad del cierre eurimérico. Por tanto, la existencia o no de soluciones liouvillianas puede leerse en este nuevo grupo que definimos.

Aunque esta ampliación al cierre eurimérico soluciona muchos problemas de sensibilidad numérica, deja sin resolver el problema de las raíces de la unidad antes citado. El problema equivalente de la decisión de si un número efectivo es racional o no se resuelve mediante el desarrollo en fracción continua: un número es racional si y sólo si su desarrollo en fracción continua es finito. Para quedarnos con un proceso finito, truncamos este desarrollo cuando el denominador del último convergente obtenido sobrepasa una cota Q dada, decidiendo que el número es irracional, aunque pueda ser racional con denominador grande. Si la ecuación a resolver tiene orden n, fijamos $\mathsf{Q} = I(n)$ con I la cota de Singer. De esta manera, decidimos correctamente que una raíz de la unidad de orden $I(n)$ a lo sumo es raíz de la unidad, pero decidimos incorrectamente que una raíz de la unidad de orden mayor que $I(n)$ no lo es, al precio de poder perder soluciones liouvillianas y con y'/y algebraico de grado mayor que $I(n)$, pero conservando aquéllas con y'/y

[7] Según [Kol48, p. 5, n. 4], esta terminología fue sugerida por Ritt, quien previamente utilizó otros términos como "funciones de Liouville" o "l-funciones".

algebraico de grado $I(n)$ a lo sumo, es decir, las soluciones singerianas.

A lo largo de la tesis se exponen los instrumentos de carácter teórico que permiten justificar la existencia de estos objetos, su computabilidad y, finalmente, se resume todo ello en un algoritmo. La idea general de dicho algoritmo es la siguiente:

1. Se computan, en principio, los generadores de Ramis del grupo de Galois, como matrices de números efectivos, y un subespacio V de soluciones que contiene las soluciones singerianas.
   (a) Para un generador unipotente $\mathtt{U}$, se computa $V := V \cap \ker(\mathtt{U} - \mathtt{I})$ a cierta precisión.
   (b) Para un toro algebraico, se computa su cierre eurimérico, ya que que es más sencillo y, de todos modos, se va a tener que calcular más adelante.
2. Se computa el cierre eurimérico del grupo de Galois a cierta precisión.
3. Se determina si el grupo tiene alguna recta invariante.
   (a) Si la respuesta es sí, se reconstruye un candidato a solución liouvilliana con técnicas numérico-simbólicas que incluyen el cálculo de sizigias y la aproximación de Padé.
   (b) Si la respuesta es no, el resultado es definitivo: no hay soluciones liouvillianas.
4. Se comprueba dicha solución.
   (a) Si la respuesta es sí, hemos terminado.
   (b) Si la respuesta es no, se reajusta la precisión y se comienza de nuevo.

Las técnicas que se desarrollan permiten determinar cuándo es posible dar por terminado el proceso. Aunque a priori no es factible saber el número de pasos que será necesario dar, demostramos que el caso 4b sólo se puede dar una cantidad finita de veces. Un estudio serio de la complejidad de las técnicas de cálculo numérico-simbólico está por ser realizado. Una implementación del algoritmo de factorización de van der Hoeven permitiría implementar fácilmente los algoritmos de esta tesis y conocer empíricamente su comportamiento. Ambos, estudio e implementación, permitirán sin duda avanzar seriamente en el desarrollo de las técnicas aquí expuestas.

Pasamos a continuación a detallar el contenido de esta memoria.

El capítulo 1 está dedicado a exponer los resultados propios del Álgebra Diferencial (§1.1) y de la Teoría de Galois Diferencial (§1.3), que serán objeto de uso a lo largo de la tesis. En este capítulo se introducen los conceptos básicos del problema a tratar: funciones liouvillianas (§1.1.3), sistemas explicitables de ecuaciones diferenciales (§1.2.1) y su equivalencia con ecuaciones diferenciales escalares (§1.1.4) por medio del Lema del Vector Cíclico. La presentación es esencialmente autocontenida, aunque no incluimos pruebas de la mayor parte de los resultados, bien por ser éstas de naturaleza muy técnica, y la prueba no ser relevante para el desarrollo de la tesis, o bien porque existen suficientes referencias para acceder a dichas pruebas. Entre los resultados más relevantes de este capítulo, destaquemos los siguientes.

- El teorema 10 (Fabry-Hukuhara-Turrittin) sobre la estructura formal de las soluciones de un sistema de ecuaciones diferenciales lineales.
- La sección 1.4 expone con detalle el teorema de densidad de Ramis, construyendo los generadores de grupo de Galois: la monodromía formal (§1.4.1) y sus componentes, el toro exponencial (§1.4.2) y los automorfismos de Stokes (§1.4.3).
- La sección 1.5 aborda la existencia soluciones liouvillianas y, en particular, un estudio detallado de la función I de Singer, que acota el grado de la extensión algebraica de las soluciones singerianas. Las cotas que allí se exponen mejoran apreciablemente las obtenidas por M. Singer, aplicando resultados recientes de Teoría de Grupos que no estaban a su disposición en 1981.
- En el teorema 43, dichas cotas son adaptadas para los sistemas de ecuaciones, que es el contexto en el que vamos a desarrollar el resto de la memoria.
- En §1.5.4, se refina el teorema de Fabry-Hukuhara-Turrittin para las soluciones singerianas.

El capítulo 2 está dedicado a exponer el numérico efectivo de van der Hoeven. En §2.1, se introducen los números complejos efectivos, tras unas consideraciones informáticas, así como la operaciones de cuerpo con tales números (§2.1.3) y el cálculo de raíces de polinomios (§2.1.4), de modo que los números complejo efectivos forman un cuerpo algebraicamente cerrado. La sección 2.2 aborda el cálculo de los generadores de Ramis del grupo de Galois, pero no son calculados completamente, como se describe en el paso 1 del esbozo del algoritmo, porque el cálculo completo no es necesario para nuestro propósito. (Véase [vdH99, vdH01, vdH07a, vdH07b] para el cálculo completo.) Concretamente, §2.2.1 trata de las soluciones en un punto no singular, §2.2.2 de la prolongación analítica, §2.2.3 de las soluciones en un

punto singular y §2.2.4 del cálculo parcial del grupo de Galois. El resto del capítulo está dedicado a los problemas que este numérico trae consigo y cómo evitarlos. Las secciones 2.3 y 2.4 tratan de los errores causados por el cálculo inexacto del rango sobre, respectivamente, $\mathbb{C}$ y $\mathbb{Q}$. Lidiamos con esas fuentes de error mediante unos parámetros globales que se discuten en §2.5. Este capítulo no tiene ningún enunciado formal, excepto el teorema 49, pero sí hay algunos enunciados informales que se demuestran en el texto o se remite a la fuente. Entre los métodos expuestos en este capítulo, destacaría los siguientes:

- La división de números complejos efectivos, en §2.1.3.
- El cálculo de raíces de polinomios por los métodos de cuadrisección y del círculo de escisión (*splitting circle*), en §2.1.4.
- El cálculo y sumación de las soluciones en un punto no singular, que es una generalización de van der Hoeven para sistemas de ecuaciones diferenciales, en §2.2.1.
- En §2.2.3 se cita de sus fuentes los métodos para tratar con las soluciones de los sistemas de ecuaciones diferenciales en un punto singular. También se cita un ejemplo de por qué no es buena idea convertir el sistema en una ecuación escalar.
- Las secciones 2.3 y 2.3 discuten, respectivamente, la eliminación gaussiana y el desarrollo en fracción continua con números complejos efectivos.

El capítulo 3 está dedicado a los grupos algebraicos lineales, que se introducen en §3.1. La sección 3.2 trata del algoritmo de Derksen–van der Hoeven para calcular el grupo algebraico lineal generado por varias matrices dadas, señalando el efecto de los errores descritos en §2.4. Para minimizar estos errores, introduzco en §3.3 los grupos eurimércos. Para ilustrar estos conceptos, en §3.4 repasamos los subgrupos algebraicos de $\mathrm{GL}(2,\mathbb{C})$ y en §3.5 calculamos su cierre eurimérico. Tras este intermedio, nos volvemos a centrar en el algoritmo de Derksen–van der Hoeven. En §3.6 se adapta este algoritmo para calcular el grupo eurimérico generado por los datos, que resulta ser más simple y más lineal. La sección 3.8 trata de cómo se comporta este algoritmo de Derksen–van der Hoeven linealizado, que es correcto bajo aritmética exacta, bajo numérico efectivo, §3.8.1 para los errores estudiados en §2.4 y §3.8.2 para los errores estudiados en §2.3, haciendo uso de algunos resultados demostrados en §3.7. De entre las definiciones y los resultados de este capítulo, permítaseme destacar las siguientes:

- La sección 3.1 introduce el concepto de los grupos algebraicos lineales y sus álgebras de Lie:

  - Un *grupo algebraico* lineal es un subgrupo de $\mathrm{GL}(V)$ ó $\mathrm{GL}(n, \mathbb{K})$ cerrado en la topología de Zariski.
  - Si un subgrupo algebraico de $\mathrm{GL}(V)$ ó $\mathrm{GL}(n, \mathbb{K})$ está dado por una familia $\mathcal{F}$ de ecuaciones algebraicas en n^2 variables, su *álgebra de Lie* asociada es el subespacio vectorial de $\mathfrak{gl}(V)$ or $\mathfrak{gl}(n, \mathbb{K})$ dado por los polinomios lineales homogéneos tangentes a la identidad a los polinomios de $\mathcal{F}$.
  - La componente de la identidad (tanto irreducible como conexa en la topología de Zariski) de un grupo algebraico G es un subgrupo algebraico normal G° de índice finito cuyas clases son las componentes de G.

- La sección 3.3 introduce los grupos anchos y eurimericos:
  - Una subálgebra $\mathfrak{g}$ de $\mathfrak{gl}(V)$ ó $\mathfrak{gl}(n, \mathbb{K})$ da lugar a su grupo multiplicativo G, llamado *grupo ancho*. También $\mathfrak{g}$ puede verse como un álgebra de Lie, llamada *álgebra de Lie ancha*.
  - El grupo ancho G es un grupo algebraico y su álgebra de Lie es $\mathfrak{g}$.
  - Podemos recuperar $\mathfrak{g}$ a partir de G tomando el espacio vectorial que genera. (Lema 57)
  - Un grupo algebraico dado por ecuaciones lineales es un grupo ancho. (Lema 58)
  - un grupo algebraico lineal cuya componente de la identidad es ancha se llama *grupo eurimérico*, también virtualmente ancho y ancho-por-finito.

- El teorema 64 dice que, llevando a cabo los cálculos de manera exacta, la modificación del algoritmo de Derksen–van der Hoeven introducida en §3.6 termina con un resultado exacto.

- El corolario 67 dice que, si H es el cierre eurimérico de un grupo algebraico G, entonces G° es diagonalizable/triangularizable/abeliano/resoluble si y sólo si H° lo es, lo cual puede extender el interés de los cierres eurimericos más allá del ámbito de esta memoria.

El capítulo 4 está dedicado al algoritmo principal de esta tesis. La sección 4.1 introduce los polinomios de Darboux, que será el formalismo para expresar las soluciones liouvillianas. La sección 4.2 trata de la reconstrucción de las funciones racionales, desde su desarrollo en serie de potencias a fracción de polinomios, necesaria para la reconstrucción simbólica de la solución. Para esta tarea usaremos la aproximación de Padé, introducida en §4.2.1. En §4.2.2 estudiamos los efectos

en la aproximación de Padé de la fuente de error descrita en §2.3. En §4.2.3 repasamos los modos de calcular los aproximantes de Padé. La sección 4.3 trata de la algebraicidad de los coeficientes numéricos de los aproximantes de Padé, que es necesaria para que funcionen los algoritmos de la sección 4.4, los cuales tratan de reconstruir simbólicamente el polinomio mínimo de estos números. El problema de la sección 4.4 se reduce al cálculo de sizigias, las cuales son combinaciones lineales con coeficientes enteros, de algunos números. Para esta tarea, esta sección describe diferentes algoritmos (LLL, HJLS y PSLQ) y su comportamiento bajo numérico efectivo. La sección 4.5 trata del algoritmo principal de esta tesis, cuya demostración se reduce a un estudio cuidadoso de los parámetros globales explicados en §2.5. La sección 4.6 está dedicada a los comentarios finales: §4.6.1 a las estrategias para acelerar el algoritmo y §4.6.2 a las cuestiones abiertas. Permítaseme destacar el teorema principal de esta tesis:

- El algoritmo principal descrito en §4.5 termina con una solución liouvilliana no nula, si es que la hay, o con la afirmación de que 0 es la única solución liouvilliana, si es que es el caso. (Teorema 103)

English summary

Introduction

One of the central problems of mathematics is solving equations. These can be of various types: algebraic, differential, functional, etc. Also, in each of these types we can distinguish several classes. Thus, for example, a differential equation may be ordinary, partial, linear, etc.

The first question to be asked in order to deal with this problem is the following: What do we mean by solving an equation? The answer is neither trivial nor unique; it depends on the context in which we work. Thus, for example, solving an equation can be interpreted in the following ways:

1. Proving the existence of a solution. This requires defining precisely what we mean by *solution* or, what comes to be an equivalent problem, in which space we will find the solutions.

2. Describing explicitly a solution. Again, it is important to clarify what we mean by *explicit solution*. For now we will stick with the intuitive idea that it is a solution which can be gotten in a finite number of steps from known objects.

3. Approximating a solution. Sometimes it may happen that the explicit description of a solution is impossible or at least very laborious. For practical purposes it may be enough an approximation numerical or in other terms, for example, the truncation of a series solution.

Depending on the context, and eventually on the problem, any notion of solution in the previous list may be valid. These three notions are introduced in their logical order, but historically they emerged in the reverse order. Already in Mesopotamia and Egypt there was interest in solving problems which in modern

terms would be equivalent to solving polynomial equations of first and second degree, but they lacked, according to [BM91, p. 41], "clear-cut distinctions between exact and approximate results." The Greek mathematicians devoted major efforts to solve geometric problems what meant[8] *constructibility* by straightedge and compass. In modern terms, these instruments are capable of constructions equivalent to solve a string equations of first and second degree.

Introduced the algebraic language in the Renaissance, the interest in solving equations of higher degrees increased, coming to solve the equations of third and fourth grade. In the solving methods, these solutions are described in terms of radicals, possibly involving even-index roots of negative numbers, which were introduced by Cardan in his *Ars Magna* [Car1545], not without some controversy. It was an important conceptual breakthrough to consider the existence of (complex) solutions in spaces different from the space of the coefficients of the equations (real numbers), comparable only to the introduction of irrational numbers after the demonstration of the irrationality of the square roots of prime numbers.

Two notable milestones in the chapter of the solution of algebraic equations are the proof of the Fundamental Theorem of Algebra by Gauss[9] and the impossibility of constructing solutions of the quintic by Abel.[10] The former result is of an existential nature; el latter is of a constructive kind: it proves the impossibility of the existence of a finite explicit construction, in terms of radicals, that allows to solve the fifth-degree general equation. Remark here that, in order to generalize this result, is basic the construction of Galois[11] of the group of symmetries associated to an algebraic equation. Such warranty of existence and impossibility of construction by radicals give way to the approximative methods for the equations of degree 5 and higher. In §2.1.4 several such approximative methods of roots of polynomials are considered.

We turn now to the terrain of differential equations. Again, the three aforementioned solution notions make sense. Consider, for example, the simplest differential equation, that is just $y' = f$, where f is a function expressible in known terms. As any first-year student knows, the continuity of the function f is enough to

[8] This is the so-called *plane* geometry of the Greeks; cf. [PH1876, book III, prop. 4, p. 55].

[9] Gauss gave the first valid proof of the Fundamental Theorem of Algebra in his PhD thesis [Gau1799], with the descriptive title "new proof of the theorem [which states that] all integral rational algebraic function [i.e., polynomial] of one variable can be split into real factors of the first or second degree." Yet he would publish three proofs more of this theorem.

[10] N. H. Abel published the impossibility of solving the quintic by radicals in his abridged memoir [Abe1824] "on the algebraic equations where it is proved the impossibility of solving the general equation of the fifth degree." He would later publish extended versions.

[11] É. Galois submitted his memoir "on the conditions of resolubility of equations by radicals" to the Academy of Sciences in Paris, but it was rejected. Several years after his early death, it was published by Liouville in [Gal1846].

guarantee the existence of a solution of this equation, which will be of the form

$$y(x) = \int_{x_0}^{x} f(t)\,\mathrm{d}t,$$

taking the integral in the sense of Riemann. (Extensions of this integral allow to address the problem for f in wider families of functions.) This meets the existential notion of solution, but, even in the case of simple functions f (expressed in elementary[12] terms), the search of a primitive in finite terms becomes highly non-trivial problem.

In 1833, Liouville presents a memoir "on the integration of a class of transcendental functions", published in [Lio1835], where he determines for which functions, composed of algebraic functions and their primitives, one can get a primitive expressible in finite terms: «*If P is an algebraic function of x and $y_1, \ldots, y_m$, with y_i primitive of an algebraic function of x and $y_1, \ldots, y_m$, and P admits a primitive in finite terms (composition of algebraic functions, exponentials and logarithms) of x and $y_1, \ldots, y_m$, then the primitive of P (w.r.t. x) is a linear combination of algebraic functions and logarithms of algebraic functions of x and $y_1, \ldots, y_m$.*» As he states in [Lio1835, p. 94], this theorem "should be considered as fundamental in the theory of functions of one variable." He applies this result to proving that "the integral $\int \mathrm{e}^x/x\,\mathrm{d}x$, which has greatly interested the geometers, is not expressible in finite terms;" cf. [Lio1835, §VIII]. Liouville also shows that the linear differential equation $y'' - y = -1/x$ admits no such solution. The results above turn out to be the basis of the Risch algorithm, presented in 1968 [Ris68, Ris69, Ris70], for determining whether an elementary function admits an elementary and primitive compute it in the positive case.

With respect to linear differential equations of a form more complex than $y' = f$, Liouville himself considers, among other problems, la search of algebraic solutions. For a first-order equation, the problem is not more complex than the computation of primitives, explained above. With respect to the second-order equation, H. Schwarz determines in 1871, published in [Sch1872], criteria for the Gauss hypergeometric equation

$$x(x-1)y'' + (c - (a+b+1)x)y' - aby = 0$$

to have algebraic solutions, in terms of a, b and c. Other results of this kind were obtained, and a systematic study of the problem came with the differential Galois

[12] An elementary function is, informally speaking, a composition of algebraic functions, exponentials and logarithms. This notion can be formalized in a similar way as in §1.1.3 with the notion of Liouvillian functions. The terminology of *elementary* functions is due to Ritt, cf. [Rit48], who translated Liouville's results to an algebraic language; Liouville uses the expression "in finite terms" instead.

theory, developed by Ritt, Kolchin, Kaplansky and Ramis, among other authors. Imitating the classic Galois theory of algebraic equations, one constructs a differential Galois group associated to a linear differential equation. This group turns out to be an algebraic group of matrices. The existence of a fundamental system of solutions expressible in a finite number of algebraic extensions, quadratures and exponentials of quadratures (i.e., Liouvillian solutions) turns out to be equivalent to the component of the identity (the connected component in the Zariski topology containing the identity matrix) of the Galois group being solvable. In matrix terms, it is equivalent to this component being conjugate to a subgroup of triangular matrices.

Using these techniques, J. Kovacic obtains in 1979 (and publishes in [Kov86]) the first complete algorithm for determining the existence of Liouvillian solutions of a second-order linear differential equation over the rational functions. The algorithm is based on the fine study of algebraic subgroups of $\mathrm{SL}(2,\mathbb{C})$ and analyzing all the possible cases in a particular order. In 1996 [UW96] F. Ulmer and J.-A. Weil publish an alternative to the Kovacic algorithm by means of the use of the symmetric powers differential operators. In 1981 M. Singer publishes [Sin81] a general algorithm for arbitrary order but based on a bound of Jordan that makes it impracticable. For third-order equations, Singer and Ulmer develop specific algorithms [SU93a, SU93b], and O. Cormier [Cor01] opens the way for orders 4 and 5, but he also obtains that a similar algorithm for order 6 would require to work with an auxiliary equation of order grater than 10^{15} even when there is no Liouvillian solution. These algorithms are of symbolic nature and their complexity increases terribly with the order of the equation, becoming impracticable from the sixth order onwards.

Another way to try to determine the existence of Liouvillian solutions is through the computation of the differential Galois group of the equation. The only line in this algorithm published to date is that of E. Hrushovski [Hru02], though it is obscure and difficult to understand and implement. Let us mention M. Singer and R. Feng are trying nowadays to review Hrushovski's work in order to get a more comprehensive algorithm.

Leaving the symbolic computation is not easy to get results of numerical character, due to the sensitivity of the Galois group to small variations. Nevertheless, J. van der Hoeven expounds in [vdH07a] a hybrid kind of numeric-symbolic computation, which incorporates numerical approximation techniques but with arbitrary precision, the effective complex numbers, which he develops in previous works. As he defines in [vdH06a], a number $x \in \mathbb{R}$ is effective if it is endowed with an algorithm with input $\varepsilon \in \mathbb{Z}_{>0}2^{\mathbb{Z}}$ and output $x_\varepsilon \in \mathbb{Z}2^{\mathbb{Z}}$ so that $|x_\varepsilon - x| < \varepsilon$. A cardinality argument shows that almost all real numbers are not effective, as effective

numbers are countable, although by definition is hard to find a number that is not. The knowledgeable reader may relate the effectiveness of a real number with the Kolmogorov complexity.

Similarly one defines effective complex numbers, effective holomorphic numbers, etc.; cf. [vdH05]. In a series of articles, J. van der Hoeven proves the effectiveness of the analytical continuation of a holomorphic solution of a differential equation over the rational functions avoiding the singularities in [vdH99] and, extending the notion of analytic continuation, in regular singularities in [vdH01] and in irregular singularities in [vdH07b]; and, what is most relevant for our study, he performs the effective construction of the Galois group of a differential equation over the rational functions based on the density theorem of J.-P. Ramis, which gives three kinds of generators of the differential Galois group as an algebraic group: the formal monodromy, the exponential torus and the Stokes automorphisms. In [vdH07a], J. van der Hoeven constructs the Ramis generators as matrices of effective complex numbers and uses them for giving numeric-symbolic algorithm for factoring differential operators over the rational functions, i.e., decomposing a linear differential equation $L[y] = 0$ as $L_1[L_2[y]] = 0$.

This is the context wherein this thesis is framed. The objective is to present a set of algorithmic techniques for deciding if a linear differential equation over $\mathbb{C}(x)$ admits or not a Liouvillian solution finding one in the affirmative case. Notice that the general notion of Liouvillian function, constructed from a chain of simple extensions, is not easy to handle. Liouville himself, in his memoir "on the integration of a class of second-order differential equations in explicit finite quantities," published in [Lio1839], shows that, if a differential equation of the kind $y'' = P(x)y$, with $P(x)$ a polynomial, admits "an integral expressible as an explicit finite function of x", then there will be a solution of the form $y = \mathrm{e}^{\int u}$, where u is an algebraic function[13] determined by the Riccati equation $u' + u^2 = P(x)$. Although in principle Liouville only deals with the integrals in terms of elementary functions, in [Lio1839, §18] claims that everything works the same if quadratures are added, reaching the current notion of Liouvillian integral, named[14] in his honor. in general, if a linear differential equation admits a Liouvillian solution, it actually admits a solution y with y'/y algebraic over the field of coefficients of the equation. This result is the basis of the aforementioned different symbolic algorithms. Vessiot gave an erroneous version in [Ves1892, p. 245], corrected by Kolchin in [Kol48]. Moreover, Singer proves that there exists an arithmetic function I such that, if a

[13] This is true in general for $P(x)$ rational. In the particular case of $P(x)$ polynomial, Liouville gets that u is rational, and even the sum of a polynomial and the logarithmic derivative of another polynomial.

[14] According to [Kol48, p. 5, n. 4], this terminology was suggested by Ritt, who had previously used other terms like "functions of Liouville", "Liouville functions" or "l-functions".

linear differential equation of order n admits a Liouvillian solution, then it admits a solution y with y'/y algebraic over the field of coefficients of the equation of degree $I(n)$ at most. We will call these solutions *Singerian*. The techniques we develop pursue finding a Singerian solution, if any, or deciding that this is not the case.

The general idea is, thus, the following. In the framework of effective computations, we construct the generators of the Galois group of the differential equation, using some of the techniques developed by J. van der Hoeven. The algebraic group they generate is very sensitive to small variations in the entries of the generators. For instance, the algebraic group generated by $\lambda \in \mathbb{C}^*$ is finite if λ is a root of unity, but it is the whole $\mathbb{C}^*$ if λ is not. For this reason we construct a bigger group than the Galois group, its *eurymeric closure*, which is easier to compute in the context of effective computation. The group preserves some interesting properties of the Galois group; particularly, if there exists an invariant line by the identity component of the Galois group, then the same line is invariant by the identity component of the eurymeric closure. Therefore, the existence or not of Liouvillian solutions can be read in this new group we define.

Although this extension to the eurymeric closure solves many problems of numerical sensitivity, it leaves the aforementioned problem of roots of unity unsolved. The equivalent problem of deciding whether an effective number is rational or not is solved by the continued fraction expansion: a number is rational if and only if its continued fraction expansion is finite. In order to keep the process finite, we truncate this expansion when the denominator of the last convergent obtained exceeds a given bound $\mathtt{Q}$, deciding that the number is irrational, though it may be rational with a large denominator. If the equation to solve has order n, we make $\mathtt{Q} = I(n)$ with I the bound of Singer. This way, we decide correctly that a root of unity of order $I(n)$ at most is a root of the unity, but we decide incorrectly that a root of unity of order greater than $I(n)$ is not a root of the unity, at the price of the possibility of losing Liouvillian solutions y with y'/y algebraic of degree greater than $I(n)$, but keeping those with y'/y algebraic of degree $I(n)$ at most, i.e., the Singerian solutions.

Throughout the thesis we expound theoretical tools to justify the existence of these objects, their computability and finally all is summarized in an algorithm. The general idea of this algorithm is the following:

1. We compute, in principle, the Ramis generators of the Galois group, as matrices of effective numbers, and a subspace V of solutions that contains the Singerian solutions.

   (a) For a unipotent generator $\mathtt{U}$, we compute $V := V \cap \ker(\mathtt{U} - \mathtt{I})$ up to

certain precision.

   (b) For an algebraic torus, we compute its eurymeric closure, as it is easier and, anyway, it will be computed later.

2. We compute the eurymeric closure of the Galois group up to certain precision.

3. It is determined whether the group has any invariant line.

   (a) If the answer is yes, a candidate for Liouvillian solution is reconstructed with numeric-symbolic techniques, including the computation of syzygies and the Padé approximation.

   (b) If the answer is no, the result is definitive: there are no Liouvillian solutions.

4. This solution is verified.

   (a) If the answer is yes, we are done.

   (b) If the answer is no, we start again with a finer precision.

The techniques we develop allow us to recognize when we can terminate the process. Although a priori it is not feasible to know the number of steps that will be necessary to take, we show that case 4b can occur only finitely many times. A serious study of the complexity of the techniques of numeric-symbolic computation is to be made. An implementation of van der Hoeven's algorithm of factorization would allow to easily implement the algorithms of this thesis and to empirically know its behavior. Both, the study and the implementation, will certainly allow to make serious progress in the development of the techniques presented here.

We turn next to detail the contents of this memoir.

Chapter 1 is devoted to expound the results belonging to Differential Algebra (§1.1) and to Differential Galois Theory (§1.3), that will be used throughout the thesis. In this chapter we introduce the basic concepts for the problem we deal with: Liouvillian functions (§1.1.3), explicitable systems of differential equations (§1.2.1) and their equivalence with scalar differential equations (§1.1.4) by means of the Cyclic Vector Lemma. The presentation is essentially self-contained, though we do not include proofs of most of the results, either because they are highly technical, and the proof is not relevant for the development of the thesis, or because there are enough references to access these proofs. Among the most relevant results in this chapter, let me highlight the following:

- Theorem 10 (Fabry-Hukuhara-Turrittin) on the formal structure of the solutions of a system of linear differential equations.

- Section 1.4 expounds the Ramis density theorem, constructing with detail the generators of the Galois group: the formal monodromy (§1.4.1) and its components, the exponential torus (§1.4.2) and the Stokes automorphisms (§1.4.3).
- Section 1.5 approaches the existence of Liouvillian solutions and, particularly, a detailed study of the function I of Singer, which bounds the degree of the algebraic extension of the Singerian solutions. The bounds there shown improve appreciably those obtained by M. Singer, applying recent results in Group Theory that were not available to him in 1981.
- In Theorem 43, those bounds are adapted for systems of equations, which is the context within which we will develop the rest of the memoir.
- In §1.5.4, Fabry-Hukuhara-Turrittin is refined for Singerian solutions.

Chapter 2 is devoted to expound van der Hoeven's effective numerics. In §2.1, effective complex numbers are introduced, after some computer-science considerations, as well as the field operations with such numbers (§2.1.3) and the computation of roots of polynomials (§2.1.4), so the effective complex numbers form a algebraically closed field. Section 2.2 approaches computation of the Ramis generators of the Galois group, but they are not completely computed, as described in the step 1 of the sketch of the algorithm, because the full computation is not needed for our purpose. (See [vdH99, vdH01, vdH07a, vdH07b] for the complete computation.) Concretely, §2.2.1 deals with the solutions at a non-singular point, §2.2.2 with the analytic continuation, §2.2.3 with the solutions at a singular point and §2.2.4 with the partial computation of the Galois group. The rest of the chapter is devoted to the problems that this numerics brings and how to avoid them. Sections 2.3 and 2.4 deal with the errors caused by the inexact computation of the rank over, respectively, $\mathbb{C}$ and $\mathbb{Q}$. We cope with those sources of error by means of some global parameters discussed in §2.5. This chapter has no formal claim, except Theorem 49, but some informal claims are proved in the text or referred to the source. Among the methods expounded in the chapter, I would highlight the following:

- The division of effective complex numbers, in §2.1.3.
- The computation of roots of polynomials by the methods of quadrisection and splitting circle, in §2.1.4.
- The computation and summation of the solutions at a non-singular point, which is generalized from van der Hoeven to systems of differential equations, in §2.2.1.

- In §2.2.3 the methods for dealing with the solutions of differential systems at a singular point are cited from their sources. An example of why converting the system to an scalar equation is not a good idea is also cited.

- Sections 2.3 and 2.3 discuss, respectively, Gaussian elimination and continuous fraction expansion with effective complex numbers.

Chapter 3 is devoted to linear algebraic groups, which are introduced in §3.1. Section 3.2 deals with the Derksen–van der Hoeven algorithm for computing the lineal algebraic group generated by some given matrices, pointing out the effect of the errors described in §2.4. In order to minimize these errors, I introduce in §3.3 the eurymeric groups. In order to illustrate these concepts, in §3.4 we survey the algebraic subgroups of $\mathrm{GL}(2,\mathbb{C})$ and in §3.5 we compute their eurymeric closure. After this intermezzo, we focus again on Derksen–van der Hoeven algorithm. In §3.6 this algorithm is adapted for computing the eurymeric group generated by the data, and it becomes simpler and more linear. Section 3.8 deals with how this linearized Derksen–van der Hoeven algorithm, which is correct under exact arithmetic, behaves under effective numerics, §3.8.1 for the errors studied in §2.4 and §3.8.2 for the errors studied in §2.3, using some results proved in §3.7. Among the definitions and results of this chapter, let me highlight the following:

- Section 3.1 introduces the concept of linear algebraic groups and their Lie algebras:
  - A linear *algebraic group* is a subgroup of $\mathrm{GL}(V)$ or $\mathrm{GL}(n,\mathbb{K})$ closed under the Zariski topology.
  - If an algebraic subgroup of $\mathrm{GL}(V)$ or $\mathrm{GL}(n,\mathbb{K})$ is given by a family $\mathcal{F}$ of algebraic equations in n^2 variables, its associated *Lie algebra* is the vector subspace of $\mathfrak{gl}(V)$ or $\mathfrak{gl}(n,\mathbb{K})$ given by the linear homogeneous polynomials tangent at the identity to the polynomials in $\mathcal{F}$.
  - The component of the identity (both irreducible and connected in Zariski topology) of an algebraic group G is a finite-index normal algebraic subgroup G° whose cosets are the components of G.

- Section 3.3 introduces the broad and eurymeric groups:
  - A subalgebra $\mathfrak{g}$ of $\mathfrak{gl}(V)$ or $\mathfrak{gl}(n,\mathbb{K})$ yields its multiplicative group G, called a *broad group*. Also $\mathfrak{g}$ can be seen as a Lie algebra, called a *broad Lie algebra*.
  - The broad group G is an algebraic group and its Lie algebra is $\mathfrak{g}$.
  - We can recover $\mathfrak{g}$ from G by taking the linear span. (Lemma 57)

  - An algebraic group given by linear equations is a broad group. (Lemma 58)
  - A linear algebraic group whose identity component is broad is called a *eurymeric group*, also called a virtually broad and broad-by-finite.

- Theorem 64 says that, performing the computations exactly, the modification of Derksen–van der Hoeven algorithm introduced in §3.6 terminates with exact output.

- Corollary 67 says that, if H is the eurymeric closure of an algebraic group G, then G° is diagonalizable/triangularizable/abelian/solvable if and only if H° is so, which may extend the interest of eurymeric closures outside the scope of this memoir.

Chapter 4 is devoted to the main algorithm of this thesis. Section 4.1 introduces Darboux polynomials, which is the formalism for expressing the Liouvillian solutions. Section 4.2 deals with the reconstruction of rational functions, from their power series expansion to fraction of polynomials, necessary for the symbolic reconstruction of the solution. For this task we use Padé approximation, which is introduced in §4.2.1. In §4.2.2 we study the effect in the Padé approximation of the source of error described in §2.3. In §4.2.3 we review the ways to compute the Padé approximants. Section 4.3 deals with the algebraicity of the numeric coefficients of the Padé approximants, which is necessary in order that the algorithms of Section 4.4, which try to reconstruct the symbolic minimal polynomial of these numbers, work. The problem of Section 4.4 reduces to the computation of syzygies, which are linear combinations with integer coefficients, of some numbers. For this task, this section describes different algorithms (LLL, HJLS and PSLQ) and their behavior under effective numerics. Section 4.5 deals with the main algorithm of the thesis, whose proof reduces to a careful study of the global parameters explained in §2.5. Section 4.6 is devoted to final remarks: §4.6.1 to devices to speed up the algorithm and §4.6.2 to open questions. Let me highlight the main theorem of the thesis:

- The main algorithm described in §4.5 terminates with a nonzero Liouvillian solution, if such a solution exists, or with the statement that zero is the only Liouvillian solution if this is the case. (Theorem 103)

Chapter 1

Foundations

This chapter is devoted to the mathematical foundations of this work, excluding linear algebraic groups, reserved for chapter 3. It introduces the language of differential algebra for dealing with differential equations and the Galois theory of differential equations. We finally review a theorem of Singer that is key for finding Liouvillian solutions and the classic methods that use it.

1.1 Concepts from Differential Algebra

In this section I shall introduce several concepts from differential algebra that will be necessary for formalizing differential equations. The reference for differential algebra will be [vdPS03, §1.1], and the reference for algebra will be [Lan02].

1.1.1 Differential rings, fields and polynomials

In the same way we define rings and polynomials for dealing with algebraic equations, we define differential rings and differential polynomials for dealing with differential equations. In the same way a ring $(A, +, \cdot)$ is defined as a set A with two binary operations $+$ and $\cdot$ such that $(A, +)$ is an abelian group, $(A \setminus \{0\}, \cdot)$ is a monoid[1] and $\cdot$ is distributive over $+$, the definition of differential ring includes the

[1] Some authors ask $(A \setminus \{0\}, \cdot)$ only for being a semigroup, calling $(A, +, \cdot)$ a unitary ring if there is a 1 in A, but all our rings will be unitary.

derivative as a third constituent operation. We define a *differential ring* $(A, +, \cdot, ')$ as a set A with two binary operations $+$ and $\cdot$ and a unary operation $'$ such that $(A, +)$ is an abelian group, $(A \setminus \{0\}, \cdot)$ is a monoid, $\cdot$ and $'$ are distributive over $+$ and the Leibniz rule $(a \cdot b)' = a' \cdot b + a \cdot b'$ holds for any $a, b \in A$. A homomorphism of differential rings is an application that respects the three operations.

The usual subcategories of rings (commutative, integral domain, field) extend trivially to the differential case, since these conditions affect only to the multiplicative monoid and the 0. An element whose derivative is zero is called a *constant*. The constants of a differential ring/field form a subring/subfield.

Example 1. Any ring can be made a differential ring with the zero derivation. In this case, the ring of constants is the whole ring. This is the usual practice for $\mathbb{C}$ and all its subfields, and also for the rings of matrices over them. The polynomials, rational fractions or formal series over a ring of constants are also made constant when the variables are interpretered as parameters.

Example 2. Let p be a prime and $\mathbb{F}_p$ the field with p elements. Consider the ring $\mathbb{F}_p[x]$ with the usual derivation w.r.t. x. Its ring of constants is $\mathbb{F}_p[x^p]$. As we expected $\mathbb{F}_p$ as ring of constants, positive characteristic is pathological.

Remark 3. **From now on, all the rings will contain $\mathbb{Q}$.** This will be no practical restriction, and we will avoid pathologies of the kind explained in the example above. All the rings will be commutative except the rings of matrices and the rings of differential operators (to be defined).

Example 4. If A is a ring, the rings $A[x]$ and $A[[x]]$ are differential rings with the usual derivation w.r.t. x, and A is their ring of constants. If K is a field, the fields $K(x)$ and $K((x))$ are differential fields with the usual derivation w.r.t. x, and K is their field of constants.

In the same way a polynomial is defined as a pattern of operations in a ring, we define differential polynomials for representing operations in a differential ring. Given a differential ring R, we define $R\langle y \rangle = R[y, y', \dots, y^{(n)}, \dots]$ with the usual polysemy; if y is taken from an extension $A \supset R$, $R\langle y \rangle$ is a differential subring of A; if y is an independent variable, $R\langle y \rangle$ is the ring of polynomials in infinitely many independent variables, with the derivation that respects the derivation of R and the formal derivatives of y (i.e. $(y^{(n)})' = y^{(n+1)}$). In the latter case, we call the elements of $R\langle y \rangle$ *differential polynomials* over R. In the same way as with polynomials, the substitution of y by $a \in A$, being $R \subset A$ and extension of differential rings, is a homomorphism $R\langle y \rangle \to A$ of differential rings.

Remark 5. Recall the difference between the differential ring $R\langle y \rangle$ of differential polynomials and the ring $R[y]$ of polynomials over a differential ring, which can

become a differential ring with a suitable derivation. It is easy to prove that, the derivations on $R[y]$ extending the derivation of R correspond bijectively to the (unrestricted) choices of y' in $R[y]$. We may denote these differential rings like $R[y; y' = P(y)]$. Contrary, the elements of $R\langle y\rangle$ are denoted like $Q[y]$, being $Q[y] = P(y, y', \dots, y^{(r)})$ as a polynomial. The substitution would be like $Q[a] = P(a, a', \dots, a^{(r)})$.

1.1.2 Algebraic formalization of differential equations

The differential equations modeled by differential polynomials are called *algebraic differential equations*.[2] Algebraic differential equations are a class wide enough for many purposes, and it includes linear differential equations as differential polynomials of first degree.

If R is a differential commutative ring/field, the homogeneous linear differential polynomials over R form a module/vector-space over R, and it is easy to check that it is an algebra with the composition, defined as $P[y] \circ Q[y] = P[Q[y]]$. Notice that its identity element is $1_\circ = y$, and that $y^{(m)} \circ y^{(n)} = y^{(m+n)}$. This R-algebra is isomorphic to the Ore algebra $R[\partial; ']$, which is the R-module of polynomials $R[\partial]$ with the only product that satisfies $\partial \cdot a = a\partial + a'$ for any $a \in R$, with the correspondence given by $y^{(n)} \mapsto \partial^n$. This algebra is commutative only if R is a constant (commutative) ring. The elements of $R[\partial; ']$ are called the *differential operators* over R, and they act on any extension $A \supset R$ giving the same result as the evaluation of the corresponding differential polynomial. The action of $L = \sum_{i=0}^{r} L_i \partial^i$ on $a \in A$ yields $L[a] = \sum_{i=0}^{r} L_i a^{(i)}$.

Both formalisms, differential polynomials and differential operators, work for differential equations and systems. Differential polynomials in many variables, like $R\langle y_1, y_2, \dots, y_n\rangle$, are a usual formalism for systems, but for the vectorial formalism we need to be able to derive in vector spaces. Formally, if R is a differential ring, a *differential module* is $(M, +, \nabla, \cdot\,)$ such that $(M, +, \cdot\,)$ is a module[3] over R, ∇ is a unary operation distributive over $+$ and the Leibniz rule $\nabla(a \cdot v) = a' \cdot v + a \cdot \nabla v$ holds for any $a \in R$ and $v \in M$. The unary operation ∇ is called derivation or *connection* depending on the background. When it is called derivation, it is usually denoted the same way as the derivation of R, and the elements whose derivative is zero are called constants, but they are called *horizontal* elements when we speak of a connection.

[2] Distinct from *differential-algebraic* equations, to be introduced in §1.2.1.

[3] All our modules will be left and unitary.

Example 6. Let K be a differential field, and let us consider the element-wise derivation on the matrices over K. Then, $K^{n\times n}$ is a differential module over K, and a differential ring, and $K^{n\times 1} \simeq K^n$ is a differential module over $K^{n\times n}$. This provides the natural formalism for differential systems of the form

$$\mathtt{A}_0\mathbf{y} + \mathtt{A}_1\mathbf{y}' + \cdots + \mathtt{A}_r\mathbf{y}^{(r)} = \mathbf{0}, \tag{1.1}$$

with $\mathtt{A}_0, \mathtt{A}_1, \ldots, \mathtt{A}_r \in K^{n\times n}$.

Example 7. Let K be a differential field. A connection on K^n is determined by its action on the standard basis. For each $\mathtt{A} \in K^{n\times n}$, we denote by $\nabla_{\mathtt{A}}$ the connection on K^n whose matrix in the standard basis is $\mathtt{A}$. Notice that $\nabla_{\mathtt{0}}$ is the element-wise derivation, and can be noted with a prime. The formula $\nabla_{\mathtt{A}}\,\mathbf{y} = \mathbf{y}' + \mathtt{A}\mathbf{y}$ is immediate, and its consequence is that the horizontal elements of $(K^n, \nabla_{\mathtt{A}})$ are precisely the solutions of the system $\mathbf{y}' = -\mathtt{A}\mathbf{y}$. Thus, there is a bijective correspondence between the connections on K^n and the explicit differential systems of first order over K.

Remark 8. Example 6 provides the natural formalization of differential systems when the coordinates are fixed, whereas Example 7 provides the natural coordinate-free formalization. The restriction to first-order systems in Example 7 is avoidable by using companion systems.

In order to apply the formalism of differential operators to vectors, it suffices to observe that, if R is a differential ring, a module M over $R[\partial;']$ and a differential module (M, ∇) over $(R, ')$ are essentially the same thing, with the identification $\partial \cdot v = \nabla v$. This extends the action of $R[\partial;']$ on differential modules over $(R, ')$. The differential operator associated to (1.1) is

$$\mathtt{A}_0 + \mathtt{A}_1\partial + \cdots + \mathtt{A}_r\partial^r.$$

1.1.3 Liouvillian extensions

Formal exponentials, logarithms and primitives can be added to any differential field K. For $a \in K$, the behavior in differential algebra of $b_1 = \exp a$, $b_2 = \int a$ and $b_3 = \log a$ is given by $b_1' = a'b_1$, $b_2' = a$ and $ab_3' = a'$, thus they can be taken from $K[b_1; b_1' = a'b_1]$, $K[b_2; b_2' = a]$ and $K[b_3; b_3' = a'/a]$ respectively. The field of fractions F of any of these differential rings is a differential field with the only derivative that satisfies

$$\left(\frac{a}{b}\right)' = \frac{a'b - ab'}{b^2}$$

for any $a, b \in F$; cf. [vdPS03, Ex. 1.5.1.d]. An algebraic extension F/K is also a differential field, as there is only one derivation on F extending the derivation of K; cf. [vdPS03, Ex. 1.5.3]. Iterating these kinds of extension we define Liouvillian extensions.

We say that an extension F/K of differential fields is *Liouvillian* if

- the field of constants of K and F is the same
- and there exist a chain $K = K_0 \subset K_1 \subset \cdots \subset K_m = F$ of differential fields such that $K_{i+1} = K_i(a_i)$ where
  - either $a_i' \in K_i$,
  - or $a_i'/a_i \in K_i$,
  - or a_i is algebraic over K_i.

We say that an element is *Liouvillian* over K if it belongs to a Liouvillian extension of K. Liouvillian over the rational functions is plainly said *Liouvillian.*

Example 9. Let us consider the differential field $\mathbb{C}(x)$ and its extension F (as differential field) by $f = \exp \int \sqrt{x}$. It is easy to check that $F = \mathbb{C}(x, \sqrt{x}, f)$, so we have the chain $\mathbb{C}(x) \subset \mathbb{C}(x, \sqrt{x}) \subset F$. The extension $F/\mathbb{C}(x)$ is Liouvillian because $\sqrt{x}$ is algebraic over $\mathbb{C}(x)$ and $f'/f = \sqrt{x}$ belongs to $\mathbb{C}(x, \sqrt{x})$. Moreover, F is the field of fractions of $\mathbb{C}(x, \sqrt{x})[f; f' = f\sqrt{x}]$.

1.1.4 Differential equations, systems and modules

Linear differential equations, explicit differential systems and finite-dimensional differential modules are equivalent in the sense explained below. An explicit differential system

$$\mathbf{y}^{(r)} = \mathtt{A}_0\mathbf{y} + \mathtt{A}_1\mathbf{y}' + \cdots + \mathtt{A}_{r-1}\mathbf{y}^{(r-1)},$$

with $\mathtt{A}_0, \mathtt{A}_1, \ldots, \mathtt{A}_{r-1}$ matrices $n \times n$ over a differential field K, can be rewritten as

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{y}' \\ \vdots \\ \mathbf{y}^{(r-1)} \end{pmatrix}' = \begin{pmatrix} \mathtt{0} & \mathtt{I} & & \\ & & \ddots & \\ & & & \mathtt{I} \\ \mathtt{A}_0 & \mathtt{A}_1 & \ldots & \mathtt{A}_{r-1} \end{pmatrix} \begin{pmatrix} \mathbf{y} \\ \mathbf{y}' \\ \vdots \\ \mathbf{y}^{(r-1)} \end{pmatrix},$$

which is called its *companion system*. The particular case $n = 1$ is the classic companion system of a differential equation. This device reduces any explicit differential equation or system to an explicit first-order system $rn \times rn$.

Example 7 provides an explicit equivalence between explicit first-order systems $n \times n$ and connections on K^n. Let M be an n-dimensional differential module over K. A vector $v \in M$ is called *cyclic* if $\{\nabla^k v\}_{k=0}^{\infty}$ spans M. Not all the vectors are cyclic, since M may have proper differential submodules, but Cyclic Vector Lemma[4] grants the existence of cyclic vectors if K is not constant, a case we may avoid by taking a larger K. Let f be a cyclic vector of the dual module $(\check{M}, \check{\nabla})$, given by the formula $\check{\nabla} g(v) = g(v)' - g(\nabla v)$. Then $\{\check{\nabla}^k f\}_{k=0}^{n-1}$ is a basis of $\check{M}$. Its dual basis fixes an isomorphism $(M, \nabla) \simeq (K^n, \nabla_{\mathtt{A}})$, with $\nabla_{\mathtt{A}}$ as explained in Example 7. If $\check{\nabla}^n f = a_0 f + a_1 \check{\nabla} f + \cdots + a_{n-1} \check{\nabla}^{n-1} f$, one gets

$$\mathtt{A} = \begin{pmatrix} 0 & -1 & & \\ & & \ddots & \\ & & & -1 \\ -a_0 & -a_1 & \dots & -a_{n-1} \end{pmatrix}.$$

The corresponding system $\mathbf{y}' = -\mathtt{A}\mathbf{y}$ is the companion system of

$$u^{(n)} = a_0 u + a_1 u' + \cdots + a_{n-1} u^{(n-1)}.$$

The remaining step is reducing a differential equation

$$y^{(rn)} = a_0 y + a_1 y' + \cdots + a_{rn-1} y^{(rn-1)}$$

to an r-order system. Let B be the result of applying to the right hand side the substitution $y^{(k)} \mapsto u_i^{(j)}$, with i the quotient and j the remainder of $k \div r$; B is a homogeneous linear differential polynomial of order $r - 1$ at most, thus

$$\{u_i^{(r)} = u_{i+1}\}_{i=0}^{n-2} \cup \{u_{n-1}^{(r)} = B\}$$

is explicit of order r.

[4] See [CK00] for proofs and references on Cyclic Vector Lemma. The cyclic vectors are generic in a sense explained in [CK00, thm. 7.3] and, though a random vector is almost surely cyclic, a cyclic vector can be found algorithmically in $\mathcal{O}(n^2)$ tries.

1.2 Solutions of systems of higher order differential equations

In this section I shall introduce explicitable differential systems and their solutions, discussing also how pathological non-explicitable systems are and how they are not necessary in this thesis.

1.2.1 Explicitable differential equations

The most general kind of linear systems of differential equations would be a homogeneous $m \times n$ linear differential system

$$\mathtt{A}_0\mathbf{y} + \mathtt{A}_1\mathbf{y}' + \cdots + \mathtt{A}_r\mathbf{y}^{(r)} = \mathbf{0}, \tag{1.2}$$

with $\mathtt{A}_0, \mathtt{A}_1, \ldots, \mathtt{A}_r$ are $m \times n$ matrices over a differential field K. Such a system will be called *explicitable* if $m = n$ and $\det \mathtt{A}_r \neq 0$. In this case we have the equivalent explicit system

$$\mathbf{y}^{(r)} = -\mathtt{A}_r^{-1}\mathtt{A}_0\mathbf{y} - \mathtt{A}_r^{-1}\mathtt{A}_1\mathbf{y}' - \cdots - \mathtt{A}_r^{-1}\mathtt{A}_{r-1}\mathbf{y}^{(r-1)}. \tag{1.3}$$

If K is the field of meromorphic functions over a connected Riemann surface, we define the *singularities* of (1.2) as the singularities of (1.3), i.e., the points where some of the entries of the matrix coefficients have poles.

Non-explicitable linear homogeneous differential systems may behave very pathologically. Notice that, if (1.2) is not explicitable, the implicit mapping theorem is not applicable either. For example, the second-order system $\{y_1' = y_1, y_2'' = y_2\}$ consists of independent first-order and second-order subsystems, and its solution space has dimension 3, instead of 4, the expected for a second-order 2×2 system. Another pathology appears in the first-order equation $y_1' = y_2$, where any choice of y_1 determines a solution. So, the dimension of the solution space may be lower or greater than expected, even infinite.

Another interesting example is the first-order linear system

$$\begin{cases} \mathbf{y}' = \mathtt{A}(x)\mathbf{y} + \mathtt{B}(x)\mathbf{z}, \\ \mathbf{0} = \mathtt{C}(x)\mathbf{y} + \mathtt{D}(x)\mathbf{z}, \end{cases}$$

called *differential-algebraic*[5] because of its two subsystems. Any homogeneous linear system can be reduced to this form, as explained in [HSW68], with $\mathtt{A}$, $\mathtt{B}$, $\mathtt{C}$

[5] Distinct from *algebraic differential equations*, introduced in §1.1.2.

and $\mathtt{D}$ matrices of meromorphic functions in a common domain. This article gives a complete algorithm for reducing the differential-algebraic system to the form

$$\begin{cases} \mathbf{u}' = \mathtt{E}(x)\,\mathbf{u}, \\ \mathbf{v} \;= \mathtt{F}(x)\,\mathbf{u} + \mathtt{G}(x)\,\mathbf{w} + \mathtt{H}(x)\,\mathbf{w}', \end{cases}$$

where $\mathbf{u}$, $\mathbf{v}$ or $\mathbf{w}$ may be missing, with

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{z} \end{pmatrix} = \mathtt{P}(x) \begin{pmatrix} \mathbf{u} \\ \mathbf{v} \\ \mathbf{w} \end{pmatrix}, \tag{1.4}$$

$\mathtt{E}$, $\mathtt{F}$, $\mathtt{G}$, $\mathtt{H}$ and $\mathtt{P}$ matrices of meromorphic functions in a common domain and $\mathtt{P}$ invertible. This reduction algorithm is explained for meromorphic functions, but it also holds for any differential field. Notice that the space of solutions is finite-dimensional[6] if and only if $\mathbf{w}$ is missing.

In this work we will deal only with explicitable systems. The application of the formal Borel transform to an explicitable system, as done in [vdH07b, §3.3], produces in general non-explicitable systems, whose analysis is complicated. This analysis would be nevertheless necessary if we pretend to apply the acceleration operators to higher order systems. As we will see in §2.2.4, this will not be necessary, so we exclude non-explicitable systems from our study. A detailed study of non-explicitable systems can be found in the recent works of Barkatou, Cluzeau and El Bacha [BCEB11, BCEB09] among others.

1.2.2 Formal structure of the solutions

Let us consider an explicitable system of higher order differential equations

$$\mathtt{A}_r(x)\,\mathbf{y}^{(r)} + \mathtt{A}_{r-1}(x)\,\mathbf{y}^{(r-1)} + \cdots + \mathtt{A}_0(x)\,\mathbf{y} = \mathbf{0}, \tag{1.5}$$

where $\mathtt{A}_0(x), \mathtt{A}_1(x), \ldots, \mathtt{A}_r(x)$ are $n \times n$ matrices of formal power series. For $n = 1$, Fabry proved in his thesis[7] [Fab1885] the following result.

Theorem 10 (Fabry)**.** *A scalar differential equation*

$$a_r(x)\,y^{(r)} + a_{r-1}(x)\,y^{(r-1)} + \cdots + a_0(x)\,y = 0,$$

[6] It holds, in general, for any differential field of infinite dimension over its constants.

[7] His claim was for $a_0, a_1, \ldots, a_r \in \mathbb{C}(x)$.

with $a_0, a_1, \ldots, a_r \in \mathbb{K}((x))$ *and* $\mathbb{K}$ *an algebraically closed field of constants, has a complete system of solutions of the type*

$$\exp\left(q(x^{-1/p})\right) x^{\alpha} \sum_{k=0}^{\infty} \sum_{i=0}^{N} y_{ki} x^{k/p} \log^i x \tag{1.6}$$

with $p \in \mathbb{Z}_{>0}$, $q(x) \in x\,\mathbb{K}[x]$ *and* $\alpha, y_{ki} \in \mathbb{K}$.

The set of the $q(x^{-1/p})$, counted with their multiplicity, is an invariant of the equation. The minimal p is called the *ramification index* of the equation. The bound N for the degree in $\log x$ can be chosen $N = r-1$ because the formal derivative w.r.t. the symbol $\log x$ belongs to the Lie algebra of the Galois group, as will be explained in §1.4.1. If a solution y has degree N in $\log x$, then $y, \partial_{\log x} y, \ldots, \partial^N_{\log x} y$ are $N+1$ linearly independent solutions, as Fabry says in [Fab1888, p. 136].

For $r = 1$, (1.5) reduces to the system

$$\mathbf{y}' = -\mathtt{A}_1(x)^{-1}\mathtt{A}_0(x)\,\mathbf{y},$$

which has a formal fundamental matrix solution

$$\mathtt{F} = \mathtt{H}(x^{1/p})\, x^{\mathtt{L}} \exp\left(\mathtt{Q}(x^{-1/p})\right) \tag{1.7}$$

where p is a positive integer and $\mathtt{H}$, $\mathtt{L}$ and $\mathtt{Q}$ are $n \times n$ matrices: $\mathtt{H}$ of formal series, $\mathtt{L}$ of scalars and $\mathtt{Q}$ diagonal of polynomials with $\mathtt{Q}(0) = \mathtt{O}$. This result (together with that $\mathtt{L}$ and $\mathtt{Q}$ commute) is the classic Hukuhara-Turrittin theorem, which is equivalent to Fabry's modulo Cyclic Vector Lemma and $\partial_{\log x}$ belonging to the Lie algebra of the Galois group. The diagonal of $\mathtt{Q}$ is, up to permutation, an invariant of the equation. Putting in Jordan form

$$\mathtt{J}\,\mathtt{L}\,\mathtt{J}^{-1} = \operatorname{diag}(\alpha_1\mathtt{I} + \mathtt{N}_1, \alpha_2\mathtt{I} + \mathtt{N}_2, \ldots, \alpha_p\mathtt{I} + \mathtt{N}_p),$$

with α_i eigenvalues and $\mathtt{N}_i$ nilpotent, we have

$$x^{\mathtt{L}} = \mathtt{J}^{-1} \operatorname{diag}(x^{\alpha_1}\mathtt{I} + \mathtt{M}_1, x^{\alpha_2}\mathtt{I} + \mathtt{M}_2, \ldots, x^{\alpha_p}\mathtt{I} + \mathtt{M}_p)\,\mathtt{J},$$

with $\mathtt{M}_i = \sum_{k=0}^{n-1} \frac{1}{k!}\mathtt{N}_i^k \log^k x$. Notice that the entries of $x^{\mathtt{L}}$ are polynomials in $\log x$ of degree less than n, and thus the entries of $\mathtt{F}$ are linear combinations of Fabry type; see (1.6), considering n for r, as usual.

In the general case, Hukuhara-Turrittin is applicable to the companion system of (1.5) and thus (1.5) has a formal fundamental matrix solution

$$\mathtt{F} = \mathtt{H}(x^{1/p})\, x^{\mathtt{L}} \exp\left(\mathtt{Q}(x^{-1/p})\right)$$

where p is a positive integer, $\mathtt{H}$ is an $n \times nr$ matrix of formal series, $\mathtt{L}$ and $\mathtt{Q}$ are $nr \times nr$ matrices: $\mathtt{L}$ of scalars and $\mathtt{Q}$ diagonal of polynomials with $\mathtt{Q}(0) = \mathtt{0}$. As in the previous case, the entries of $\mathtt{F}$ are linear combinations of Fabry type; see (1.6), considering nr for r, as expected.

1.2.3 Universal field extension

Fabry's and Hukuhara-Turrittin theorems give a complete system of solutions for homogeneous linear differential equations and systems over the differential field $\mathbb{C}((x))$ of formal power series. These formal solutions are built by adding to $\mathbb{C}((x))$ symbols like $\log x$, x^α, for $\alpha \in \mathbb{C}$, and $\exp\left(q(x^{-1/p})\right)$, for $q(x) \in x\,\mathbb{C}[x]$ and $p \in \mathbb{Z}_{>0}$. The extensions defined by these symbols following §1.1.3 glue together with the algebraic relations of $x^{\mathbb{C}}$ and $\exp(y_1)\exp(y_2) = \exp(y_1 + y_2)$. The detailed construction is found in [Hen96, §2.2]=[HvdP95, §2], which uses an algebraically closed subfield $\mathbb{K} \subseteq \mathbb{C}$ and proves that the differential ring

$$R = \mathbb{K}((x))\left\langle \log x, x^{\mathbb{K}}, \exp\left(x^{-1/p}\,\mathbb{K}[^{-1/p}]\right)\right\rangle$$

contains a complete system of solutions for any homogeneous linear differential equation over $\mathbb{K}((x))$. This property grants the field of fractions Ω of R the denomination "universal field extension of $\mathbb{K}((x))$."

1.3 Differential Galois theory

In this section I shall introduce differential Galois theory, ending with Schlesinger's theorem, which gives generators of the Galois group when all the singularities of the equation are *regular*, i.e., the growth of the solutions does not correspond to essential singularities. The next section will deal with the generalization of Schlesinger's theorem to irregular singularities.

1.3.1 Picard-Vessiot extensions

Let Ω be the universal field defined in §1.2.3. Each particular differential equation or system Δ over $\mathbb{K}\{x\}$ (where $\mathbb{K}$ is an algebraically closed subfield of $\mathbb{C}$) has a minimal intermediate differential field $\mathbb{K}(\{x\}) \subseteq \hat{K} \subseteq \Omega$ where a complete system of solutions of Δ is defined. If Δ is defined over the rational functions, we have

another minimal intermediate differential field $\mathbb{K}(x) \subseteq \hat{F} \subseteq \Omega$ where a complete system of solutions of Δ is defined. At each regular point z of Δ there is a minimal intermediate differential field $\mathbb{K}(x) \subseteq K_z \subseteq \mathcal{O}_z$ where a complete system of solutions of Δ is defined. The extensions $\hat{K}/\mathbb{K}(\{x\})$, $\hat{F}/\mathbb{K}(x)$ and $K_z/\mathbb{K}(x)$ are examples of Picard-Vessiot extensions, defined below, which play the same role in Galois theory as the splitting fields of polynomials.

Example 11. The differential field $\mathbb{C}(\{x\})$ contains a solution $f = \sum_{k=0}^{\infty} x^k/k!$ of $y' = y$, but we may add another E in $\mathbb{C}(\{x\})[E; E' = E]$. This ring contains two solutions of a first-order equation that are linearly independent over $\mathbb{C}$. The field of fractions F contains a new constant $f^{-1}E$, thus these solutions are linearly dependent over the constants of F.

The informal definition of Picard-Vessiot extension F/K of a differential equation or system Δ over K is that F is generated by a complete system of solutions of Δ, but Example 11 shows the problems that may appear adding new solutions without restriction. We say that an extension of differential fields F/K is the *Picard-Vessiot extension* of Δ if the following conditions are satisfied:

- K and F have the same field of constants C;
- the C-vector space $V \subset F$ of solutions of Δ has the right dimension; if Δ has order r and size $n \times n$, the right dimension is nr;
- F is the field of fractions of $K\langle V\rangle$.

So Example 11 is not a Picard-Vessiot extension because the field of constants is augmented.

Picard-Vessiot extensions have the following properties.

Theorem 12 (Existence)**.** *If K is a differential field, its field of constants is algebraically closed, and Δ defined over K, then there exists a Picard-Vessiot extension F/K for Δ.* [Mag97, thm. 3.4]

Theorem 13 (Uniqueness)**.** *If F_1/K and F_2/K are Picard-Vessiot extensions for Δ, then there exists an isomorphism $F_1 \simeq F_2$ that keeps K fixed.* [Mag97, thm. 3.5]

Theorem 14 (Minimality)**.** *If F/K and E/K are Picard-Vessiot extensions for Δ with $E \subseteq F$, then $E = F$.* [Mag97, lemma, p. 24]

The proof of the existence of a Picard-Vessiot extension is rather technical in the general case, but in many usual cases, such a Picard-Vessiot extension can be easily

understood. If K is the field of meromorphic functions on a connected Riemann surface X, take a simply connected open set U on X free from singularities of Δ; a Picard-Vessiot extension is included in the field $\mathcal{M}(U)$ of meromorphic functions on U.

1.3.2 Differential Galois group

We associate to an extension of differential fields F/K the group $\mathrm{Gal}(F/K)$ of the automorphisms of F that leave K fixed; this is the *differential Galois group* of the extension. Here "fixed" means that all the elements are invariant. If we drop the word "differential," we get the classic definition of Galois group. Obviously, the differential Galois group is a subgroup of the classic one, but they coincide for an algebraic extension. In the same way the Galois group of an algebraic equation is defined up to isomorphism, we may define $\mathrm{Gal}(\Delta)$ as the differential Galois group of any Picard-Vessiot extension for Δ, by virtue of Theorem 13.

If F/K is a Picard-Vessiot extension for Δ with field of constants C algebraically closed and space of solutions V, $\mathrm{Gal}(F/K)$ leaves V invariant. The representation $\mathrm{Gal}(F/K) \to \mathrm{GL}(V)$ is faithful and its image, which may also be denoted by $\mathrm{Gal}(\Delta)$, is a linear algebraic group; see [Kap76, thm. 5.5]. Chapter 3 deals with linear algebraic groups, but briefly speaking we recall that a linear algebraic group is a group of linear transformations that is closed in the Zariski topology. Let us consider the lattice $\mathcal{G}$ of algebraic subgroups of $\mathrm{Gal}(F/K)$. The application that sends each $G \in \mathcal{G}$ to its fixed field

$$\mathrm{Fix}(G) = \{a \in F : \forall \sigma \in G, \sigma(a) = a\}$$

is an order-reversing homomorphism $\mathcal{G} \to \mathcal{K}$, where $\mathcal{K}$ is the lattice of intermediate differential fields of F/K. The application $E \mapsto \mathrm{Gal}(F/E)$ is another order-reversing homomorphism $\mathcal{K} \to \mathcal{G}$. As in the classic case, we have a fundamental theorem.

Theorem 15 (Galois correspondence)**.** *With the notation of the paragraph above, the homomorphisms $\mathcal{G} \to \mathcal{K}$ and $\mathcal{K} \to \mathcal{G}$ are inverses of each other, hence they give an order-reversing isomorphism of lattices $\mathcal{G} \simeq \mathcal{K}$.* [vdPS03, Prop. 1.34.1] [Mag97, thm. 6.5]

This implies a useful result.

Proposition 16. *With the notation above, if the fixed field of $G < \mathrm{Gal}(F/K)$ is K, then G is Zariski-dense in $\mathrm{Gal}(F/K)$.*

Proof. The fixed field of $\overline{G}$, the Zariski closure of G, is also K. By the Galois correspondence $\overline{G} = \mathrm{Gal}(F/K)$. □

As we shall review in more detail in §3.1, the component[8] of the identity of an algebraic group G is another linear algebraic group denoted by G°. The component of the identity $\mathrm{Gal}(F/K)^\circ$ of $\mathrm{Gal}(F/K)$ is a distinguished element of $\mathcal{G}$ with a useful property.

Theorem 17. *With the notation and hypotheses above, the fixed field of* $\mathrm{Gal}(F/K)^\circ$ *is the relative algebraic closure of* K *in* F. [vdPS03, prop. 1.34.3] [Mag97, thm. 6.5]

As reviewed in §3.1, a Lie algebra is associated to any linear algebraic group. There are several constructions of this Lie algebra but, in the case of a differential Galois group, this Lie algebra can be seen as consisting of derivations over the a differential field. We denote $\mathfrak{gal}(F/K)$ the Lie algebra of the derivations on F that vanish on K and commute with the derivation of F. These properties grant that $\mathfrak{gal}(F/K)$ leaves V invariant, thus we have a faithful representation $\mathfrak{gal}(F/K) \to \mathfrak{gl}(V)$. Moreover, the image of this representation corresponds to the Lie algebra usually associated to the image of $\mathrm{Gal}(F/K) \to \mathrm{GL}(V)$; see [vdPS03, Prop. 1.27.2].

Remark 18. As a final note, from now on, we will only work with differential rings that have an algebraically closed field of constants (of characteristic 0, according to Remark 3). This is needed for differential Galois theory. In the practical cases, this is granted by taking as constants an algebraically closed subfield $\mathbb{K}$ of $\mathbb{C}$.

What happens if we consider the same differential equation over two different (algebraically closed) fields of constants, K a subfield of F? For example, a differential equation defined over $\mathbb{Q}(x)$ can be considered over $\overline{\mathbb{Q}}(x)$ and over $\mathbb{C}(x)$. We have that the Galois group over the field of rational functions commutes with the extension of the field of constants.

Theorem 19. *Let* K/k *be an extension of algebraically closed fields,* Δ *be a linear differential equation of order* n *over* $k(x)$ *with the usual derivation,* F_k *be a Picard-Vessiot extension for* Δ *over* $k(x)$*, and* $\mathcal{B}$ *a fundamental system of solutions of* Δ *in* F*. Extending the constants to* K*,* $F_K = K \otimes_k F_k$ *is a Picard-Vessiot extension for* Δ *over* $K(x)$*. The system* $\mathcal{B}$ *yields representations* $\mathrm{Gal}(F_k/k(x)) \to \mathrm{GL}(n,k)$ *and* $\mathrm{Gal}(F_K/K(x)) \to \mathrm{GL}(n,K)$*, with respective images* G_k *and* G_K*. The algebraic group* G_K *is given by the same equations as* G_k*, and* $G_k = G_K \cap \mathrm{GL}(n,k)$. [vdH07a, §2.2¶4]

[8] I say just "component" because it is irreducible component of the algebraic variety, connected component in the Zariski topology and, if $C = \mathbb{C}$, connected component of the Lie group.

1.3.3 Schlesinger's theorem

If X is a Riemann surface, γ a path on X and f a germ of analytic function at the starting point of γ, we denote by $\mathrm{cont}_\gamma f$ the analytic continuation of f along γ, which is a germ at the end of γ. If γ_1 is a path $z_1 \rightsquigarrow z_2$, and $\gamma_2 = \gamma_1^{\blacktriangleleft}$ its inverse $z_2 \rightsquigarrow z_1$, then cont_γ defines an isomorphism $A_{\gamma_1} \to A_{\gamma_2}$ of differential rings, where $A_{\gamma_i} \subset \mathcal{O}_{z_i}$ is the subring of the germs continuable along γ_i. If $V_z \subset \mathcal{O}_z$ is the space of solutions of a homogeneous linear differential equation Δ over a connected Riemann surface X, then cont_γ defines an automorphism of the vector space V_z for any $\gamma \in \pi_1(X \setminus S, z)$, where π_1 means the fundamental group and S is the set of singularities of Δ, and hence we have a homomorphism of groups $\pi_1(X \setminus S, z) \to \mathrm{GL}(V_z)$. If $V_w \subset \mathcal{O}_w$ is another space of solutions of Δ, then cont_γ defines an isomorphism of the differential vector spaces $V_z \to V_w$ for any $\gamma \in \Pi_1(X \setminus S)(z, w)$, where Π_1 means the fundamental groupoid,[9] and hence we have a mapping $\Phi_{zw} : \Pi_1(X \setminus S)(z, w) \to \mathcal{L}(V_z, V_w)$ that defines, together with the map $z \mapsto V_z$, a functor $\Pi_1(X \setminus S) \to \mathcal{L}$ to the category of vector spaces. The same holds for Δ an explicit differential system and $V_z \subset \mathcal{O}_z^n$.

Let us consider the groupoid $\mathcal{G}$ over $X \setminus S$ with $\mathcal{G}(z, w) = \mathcal{L}(V_z, V_w)$. This groupoid is finer than the corresponding subgroupoid of $\mathcal{L}$ because it does not confuse $V_z = V_w$ for $z \neq w$. The mappings Φ_{zw} define, together with the identity on $X \setminus S$, a functor $\Pi_1(X \setminus S) \to \mathcal{G}$, whose image $\textsc{Mon}(\Delta)$ is called the *monodromy groupoid* of Δ. The image of $\pi_1(X \setminus S, z)$ is called the *monodromy group* of Δ at z, and denoted $\mathrm{Mon}(\Delta, z)$. Let F_z be the differential field generated by $K = \mathcal{M}(X)$ and (the entries of) V_z for each $z \in X \setminus S$. With these definitions, F_z/K is the Picard-Vessiot extension for Δ at z, and we may write $\mathrm{Gal}(\Delta, z) = \mathrm{Gal}(F_z/K)$ for the Galois group. The isomorphisms $F_z/K \simeq F_w/K$ form a groupoid $\textsc{Gal}(\Delta)$ over $X \setminus S$. These isomorphisms keep K fix and V_z invariant, and are determined by their action on V_z, so this determines naturally a homomorphism of groupoids $\textsc{Gal}(\Delta) \to \mathcal{G}$ and $\textsc{Gal}(\Delta)$ is naturally isomorphic to its image in $\mathcal{G}$. From the properties of analytic continuation, $\textsc{Mon}(\Delta)$ can be seen as a subgroupoid of $\textsc{Gal}(\Delta)$, and thus $\mathrm{Mon}(\Delta, z)$ as a subgroup of $\mathrm{Gal}(\Delta, z)$.

Theorem 20 (Schlesinger)**.** *With the notation of the previous paragraph,* $\mathrm{Mon}(\Delta, z)$ *is Zariski-dense in* $\mathrm{Gal}(\Delta, z)$ *if all the singularities of* Δ *are regular.*

Proof. This theorem is classic, back to [Sch1897], but here we follow the proof

[9] The fundamental groupoid of a topological space Y is the groupoid with Y as set of objects, with arrows $\Pi_1(Y)(p, q)$ from $p \in Y$ to $q \in Y$ the classes of continuous paths $p \rightsquigarrow q$ on Y modulo the homotopy with fixed ends and with composition the concatenation of paths. The fundamental groupoid contains the fundamental group $\pi_1(Y, p)$ as $\Pi_1(Y)(p, p)$.

of [Żoł06, thm. 11.21]. According to Proposition 16, it suffices to prove that any $y \in F_z$ invariant by the monodromy lies in K.

Let $y \in F_z$ be invariant by the monodromy, so y defines a holomorphic function f on $X \setminus S$. Each $w \in S$ must be an isolated regular singularity of Δ, so the growth of any solution of Δ, and thus of f, corresponds to a pole or a removable singularity. Therefore, f is extended to a meromorphic function on X. □

For X the Riemann sphere, assuming $\infty \in S$, Theorem 20 can be rephrased as the following.

Theorem 21 (Schlesinger)**.** *The Galois group of a homogeneous linear differential equation over the rational functions is the smallest algebraic group containing the monodromy group if all the singularities are regular.*

Next section will deal with the generalization of Schlesinger's theorem when there are *irregular singularities*, which are the singularities that are not regular, i.e., when there are solutions whose growth correspond to an essential singularity.

1.4 Ramis density theorem

In the case of irregular singularities, the monodromy may be not enough. The monodromy group is generated by the loops around each singularity, but each one splits in the *formal monodromy* and the *Stokes automorphisms* (to be defined) so that all of them are necessary for generating the Galois group. Apart of these, the exponentials in the Fabry solutions add the *exponential torus* (to be defined) to the generators of the Galois group. For a regular singularity, the monodromy is the same as the formal monodromy and there is no Stokes automorphism or exponential torus. In this section, we shall review all these notions and state the fundamental density theorem of J.-P. Ramis.

1.4.1 Formal monodromy

The formal monodromy, as its name suggests, is the automorphism of differential fields defined by the formal substitutions in the Fabry solutions that act like the

monodromy in each atom. These formal substitutions are

$$x^\alpha \mapsto \mathrm{e}^{2\pi\mathrm{i}\alpha}\, x^\alpha, \tag{1.8}$$

$$\log x \mapsto \log x + 2\pi\mathrm{i}, \tag{1.9}$$

where (1.8) is also applied inside the symbol exp. As the symbol $\log x$ is a primitive of x^{-1} and this determines all its behavior in differential algebra, the substitution (1.9) is Galoisian, i.e., it defines a Galois automorphism. Moreover, the formal substitution

$$\log x \mapsto \log x + \beta \tag{1.10}$$

is Galoisian for any constant β. The symbols x^α are subject to the restrictions $x^\alpha x^\beta = x^{\alpha+\beta}$ and $(x^\alpha)' = \alpha x^{\alpha-1}$, and they determine all their behavior in differential algebra. As the substitution (1.8) respects both and keeps $x^{\mathbb{Z}}$ fixed, it is Galoisian. The *formal monodromy* is the automorphism of the formal Picard-Vessiot extension given by the substitutions (1.8) and (1.9).

The substitution (1.10) defines an algebraic group $G_{\log}$ isomorphic to the additive group of constants, whose Lie algebra is spanned by the derivation $\partial_{\log}$ w.r.t. the symbol $\log x$. The set of all the α lies in a free $\mathbb{Z}$-module of finite dimension. If $\{1/p, \alpha_1, \alpha_2, \ldots, \alpha_m\}$ is a basis of the module, any choice of non-zero constants $\beta_1, \beta_2, \ldots, \beta_m$ makes the substitutions $x^{\alpha_i} \mapsto \beta_i x^{\alpha_i}$ compatible, so the substitution

$$x^\alpha \mapsto \beta_1^{\lambda_1}\beta_2^{\lambda_2}\cdots\beta_m^{\lambda_m} x^\alpha, \quad \text{for } \alpha = \lambda_0/p + \lambda_1\alpha_1 + \lambda_2\alpha_2 + \cdots + \lambda_m\alpha_m$$

with $\lambda_0, \lambda_1, \ldots, \lambda_m \in \mathbb{Z}$, is Galoisian. This defines an algebraic group G_{pow} isomorphic to an algebraic torus of rank m. The change of variables

$$x^\alpha \mapsto x^{\lambda_0/p} x_1^{\lambda_1}\cdots x_m^{\lambda_m}, \quad \text{for } \alpha = \lambda_0/p + \lambda_1\alpha_1 + \lambda_2\alpha_2 + \cdots + \lambda_m\alpha_m$$

with $\lambda_0, \lambda_1, \ldots, \lambda_m \in \mathbb{Z}$, is the inverse of $x_i \mapsto x^{\alpha_i}$. With this identification, the Euler derivation $\delta_i = x_i\partial_{x_i}$ w.r.t. x_i belongs to the Lie algebra of the Galois group. The Lie algebra of G_{pow} is spanned by $\delta_1, \delta_2, \ldots, \delta_m$. The substitution

$$x^\alpha \mapsto \mathrm{e}^{2\pi\mathrm{i}\lambda_0/p}\, x^\alpha, \quad \text{for } \alpha = \lambda_0/p + \lambda_1\alpha_1 + \lambda_2\alpha_2 + \cdots + \lambda_m\alpha_m$$

with $\lambda_0, \lambda_1, \ldots, \lambda_m \in \mathbb{Z}$, defines an automorphism that generates a finite group G_{ram} of order equal to the ramification index p. All the automorphisms and derivations defined in this paragraph commute. The algebraic group generated by the formal monodromy is the product $G_{\mathrm{ram}}G_{\mathrm{pow}}G_{\log}$ and its Lie algebra is spanned by $\delta_1, \delta_2, \ldots, \delta_m$ and $\partial_{\log}$.

1.4.2 Exponential torus

The exponential symbols are subject to the restrictions $\exp(y_1)\exp(y_2) = \exp(y_1 + y_2)$ and $(\exp y)' = y' \exp y$, and they determine all their behavior in differential algebra, so the substitution $\exp y \mapsto \beta \exp y$ is Galoisian for any non-zero constant β. The exponential symbols $\exp y$ involved in the Fabry solutions define a free $\mathbb{Z}$-module of finite dimension generated by the exponents y. If $\{y_1, y_2, \ldots, y_m\}$ is a basis of the module, any choice of non-zero constants $\beta_1, \beta_2, \ldots, \beta_m$ makes the substitutions $\exp y_i \mapsto \beta_i \exp y_i$ compatible, so the substitution

$$\exp y \mapsto \beta_1^{\lambda_1} \beta_2^{\lambda_2} \cdots \beta_m^{\lambda_m} \exp y, \quad \text{for } y = \lambda_1 y_1 + \lambda_2 y_2 + \cdots + \lambda_m y_m$$

with $\lambda_1, \lambda_2, \ldots, \lambda_m \in \mathbb{Z}$, is Galoisian. This defines an algebraic group isomorphic to an algebraic torus of rank m, called the *exponential torus*. The change of variables

$$\exp y \mapsto E_1^{\lambda_1} E_2^{\lambda_2} \cdots E_m^{\lambda_m}, \quad \text{for } y = \lambda_1 y_1 + \lambda_2 y_2 + \cdots + \lambda_m y_m$$

with $\lambda_0, \lambda_1, \ldots, \lambda_m \in \mathbb{Z}$, is the inverse of $E_i \mapsto \exp(y_i)$. With this identification, the Euler derivation $\delta_i = E_i \partial_{E_i}$ w.r.t. E_i belongs to the Lie algebra of the Galois group. The Lie algebra of the exponential torus is spanned by $\delta_1, \delta_2, \ldots, \delta_m$.

Remark 22. The exponential torus commutes with the torus of the monodromy. Moreover, the product torus admits a description in terms of the so-called exponential part $\alpha + xy'$ associated to $x^\alpha \exp y$. The set of all the exponential parts lies in a $\mathbb{Z}$-module of finite dimension. If $\{1/p, q_1, q_2, \ldots, q_m\}$ is a basis of the module, with p the ramification index, any choice of non-zero constants $\beta_1, \beta_2, \ldots, \beta_m$ makes the substitution

$$x^\alpha \exp y \mapsto \beta_1^{\lambda_1} \beta_2^{\lambda_2} \cdots \beta_m^{\lambda_m} x^\alpha \exp y, \quad \text{for } \alpha + xy' = \lambda_0/p + \lambda_1 q_1 + \lambda_2 q_2 + \cdots + \lambda_m q_m$$

with $\lambda_0, \lambda_1, \ldots, \lambda_m \in \mathbb{Z}$, Galoisian. Thus they form an algebraic torus.

1.4.3 Extended analytic continuation

The formal monodromy and its factors act on the formal Picard-Vessiot extension, so we need a way to connect it to the groupoid $\textsc{Gal}(\Delta)$. For a regular singularity, it suffices to choose a branch of the logarithm. Let us define an *extended path* as a pair (θ, γ) where $\theta \in \mathbb{R}$ represents a direction and γ is a path that starts at a singularity $\gamma(0)$ with $\gamma'(0) \in \mathrm{e}^{\mathrm{i}\theta}\mathbb{R}_+$. With this definition, an *extended analytic continuation* $\mathrm{cont}_{(\theta,\gamma)}$ that begins by choosing the determination $(\theta - \pi, \theta + \pi)$ of

the logarithm gives the desired "coming out" of the singularity. For an irregular singularity, the power series that appear in the formal solutions may be divergent, hence choosing a branch of the logarithm is not enough. These power series are *multisummable*, as proved in [BBRS91], whose meaning is briefly explained below without the precise definitions, which can be found in [Bal94].

Let us assume, for simplicity, the singularity at the origin, and denote the sectors by $S(A,R) = \{z \in \mathbb{C} : 0 < |z| < R, \arg z \in A\}$, with $0 < R \leqslant \infty$ and A a segment representing an arc. If the sector covers the circle, we assume it is defined in the Riemann surface of the logarithm. In the same way $\mathcal{O}_0$ is the injective limit of $\mathcal{O}(B_R)$, where B_R is the disk of radius R, we define the algebra $\mathcal{O}_{0,\theta}$ as the injective limit of $\mathcal{O}(S(A,R))$, for A open, $\theta \in A$ and $R > 0$.

The simplest case of multisummability is Borel summability, which I shall describe in Example 23. Following [Bal94], the Borel transform of the formal power series $\hat{f} = \sum_{n=0}^{\infty} a_n x^n$ is $\mathcal{B}[\hat{f}] = \sum_{n=0}^{\infty} a_n \xi^n / n!$ and the Laplace transform will be given by

$$\mathcal{L}[f](z) = z^{-1} \int_0^{+\infty} f(t) \exp(-t/z)\, \mathrm{d}t.$$

They are not the usual definitions of Borel and Laplace transforms, but these definitions are tuned so that, restricted to monomials, they are inverses of each other and keep the degree.

Example 23. Let us explain Borel summation through an example due to Euler [Eul1760, §19]. The formal series to sum is $\hat{f} = \sum_{n=0}^{\infty} (-1)^n n!\, x^{n+1}$. The Borel transform of $\hat{f}$ is $\hat{g} = \sum_{n=0}^{\infty} (-1)^n \xi^{n+1}/(n+1)$, whose convergence radius is 1. The sum of $\hat{g}$ is $\log(\xi+1)$, which is ramified at $\xi = -1$. Let g be the principal branch of $\log(\xi+1)$, whose Laplace transform is

$$f(z) = z^{-1} \int_0^{+\infty} \log(t+1) \exp(-t/z)\, \mathrm{d}t = \int_0^{+\infty} \frac{\exp(-t/z)}{t+1}\, \mathrm{d}t,$$

which converges in the half-plane $\mathrm{Re}(z) > 0$. Both $\hat{f}$ and f satisfy the differential equation $x^2 y' + y = x$.

Example 23 shows the Borel sum in the direction of the positive semi-axis. We may move the integration path $0 \rightsquigarrow \infty$ in the Laplace transform to the ray $\arg = \theta$, giving the Laplace transform in the direction θ by

$$\mathcal{L}_\theta[f](z) = z^{-1} \int_0^{\mathrm{e}^{\mathrm{i}\theta}\infty} f(\zeta) \exp(-\zeta/z)\, \mathrm{d}\zeta,$$

which does not depend on the representant of $\theta + 2\pi\mathbb{Z}$. We say that a formal power series $\hat{f}$ is *Borel summable* in the direction θ if its Borel transform $\hat{g} = \mathcal{B}[\hat{f}]$ is convergent and can be analytically continued in an open sector $S(A_\theta, \infty)$ with $\theta \in A_\theta$ and a growth at infinity of the form $\mathcal{O}(\exp(B_\theta|\zeta|))$ for certain $B_\theta > 0$, and we say that θ is a *direction of Borel summability* of $\hat{f}$. The maximal star-shaped set where the analytic continuation of $\hat{g}$ along rays is defined is called its *Mittag-Leffler star*. Let g be the analytic continuation of $\hat{g}$ defined in its Mittag-Leffler star. Thus $\mathcal{L}_\theta[g]$ is defined in all the directions of Borel summability, so we define the *Borel sum* of $\hat{f}$ in the direction θ as $\mathcal{S}_{1,\theta}(\hat{f}) = \mathcal{L}_\theta[g]$, which is defined inside the disk of center $\mathrm{e}^{\mathrm{i}\theta}/2B_\theta$ and radius $1/2B_\theta$. All the directions $\theta' \in A_\theta$ are also directions of Borel summability, with Borel sum defined inside the disk of center $\mathrm{e}^{\mathrm{i}\theta'}/2B_\theta$ and radius $1/2B_\theta$, all of them glue together into a sum f_θ defined in a kidney-shaped domain.

A formal power series $\hat{f}$ is *Borel summable* if it is Borel summable in all the directions but finitely many modulo 2π. These exceptional directions are called the *singular directions* of $\hat{f}$. If the singular directions are represented by $\theta_1 < \theta_2 < \cdots < \theta_m < \theta_{m+1} = \theta_1 + 2\pi$, they define the open arcs where the Borel sums are compatible and glue together into sectors $S((\theta_i - \varepsilon, \theta_{i+1} + \varepsilon), R)$, with $\varepsilon > 0$ and $R > 0$ small enough. If an arc is wider than π, this is defined on the Riemann surface of the logarithm. If there is no singular direction, $\hat{f}$ is convergent. In the sector $S((\theta_i - \varepsilon, \theta_i + \varepsilon), R)$ there are defined two Borel sums, $\mathcal{S}_{1,\theta_i^-}(\hat{f})$ and $\mathcal{S}_{1,\theta_i^+}(\hat{f})$, that may differ. This feature is called *Stokes phenomenon*.

Let us consider the following example with Euler-like formal series.

Example 24. Let us consider the formal series $\hat{f}_s = \sum_{n=0}^{\infty}(n!)^s x^n$ for $s > 0$. The Borel transform $\hat{g}_s = \sum_{n=0}^{\infty}(n!)^{s-1}\xi^n$ is convergent for $s \leqslant 1$, and divergent for $s > 1$. For $s = 1$ we have the same behavior as in Example 23. For $s < 1$, $\hat{g}_s$ defines an entire function g_s with order $1/(1-s)$. Hence, Borel summability fails for $s > 1$ because Borel transform is "not strong enough" to make $\hat{g}_s$ convergent, and it fails for $s < 1$ because Borel transform is "too strong" that it makes $\hat{g}_s$ "too convergent" at the expense of making its order too large for convergence of Laplace transform. Thus, Borel summation is tuned to divergence as rapid as the Euler series of Example 23.

In order to generalize Borel summation for series diverging slower or more rapid than the Euler series of Example 23, where ordinary Borel summability fails according to Example 24, we define, for $k > 0$, the k-Borel transform of a formal power series $\hat{f} = \sum_{n=0}^{\infty} a_n x^n$ as $\mathcal{B}_k[\hat{f}] = \sum_{n=0}^{\infty} a_n \xi^n/\Gamma(n/k+1)$. As $\Gamma(n+1) = n!$, the ordinary Borel transform is now the 1-Borel transform. We define the k-Laplace

transform in the direction θ as given by

$$\mathcal{L}_{k,\theta}[f](z) = z^{-k} \int_0^{\mathrm{e}^{\mathrm{i}\theta}\infty} f(\zeta) \exp(-\zeta^k/z^k)\, k\zeta^{k-1} \mathrm{d}\zeta,$$

with the determination $(\theta - \pi, \theta + \pi)$ of the logarithm. For k integer, $\mathcal{L}_{k,\theta}$ does not depend on the representant of $\theta + 2\pi\mathbb{Z}$, but the branch of the logarithm does matter for k not integer.

We say that a formal power series $\hat{f}$ is *k-summable* in the direction θ if its k-Borel transform $\hat{g} = \mathcal{B}_k[\hat{f}]$ is convergent and can be analytically continued in an open sector $S(A_\theta, \infty)$ with $\theta \in A_\theta$ and a growth at infinity of the form $\mathcal{O}(\exp(B_\theta|\zeta|^k))$ for certain $B_\theta > 0$, and we say that θ is a *direction of k-summability* of $\hat{f}$. Let g be the analytic continuation of $\hat{g}$ defined in its Mittag-Leffler star. Thus $\mathcal{L}_{k,\theta}[g]$ is defined in all the directions of k-summability, so we define the *k-sum* of $\hat{f}$ in the direction θ as $\mathcal{S}_{k,\theta}(\hat{f}) = \mathcal{L}_{k,\theta}[g]$, which is defined inside the inverse image of the disk of center $\mathrm{e}^{\mathrm{i}k\theta}/2B_\theta$ and radius $1/2B_\theta$ by the k-th power in the determination $(\theta - \pi, \theta + \pi)$ of the logarithm. Among the connected components of this region of convergence, we choose the component bisected by the direction θ as domain of $\mathcal{L}_{k,\theta}[g]$. This domain has the shape of a petal for $k > 1$ and a shape of a heart for $k < 1$, inscribed in the sector $S((\theta - \pi/2k, \theta + \pi/2k), B_\theta^{1/k})$. All the directions $\theta' \in A_\theta$ are also directions of k-summability, with k-sum defined in another domain of the same shape as $\mathcal{L}_{k,\theta}[g]$, all of them glue together into a sum f_θ defined in a petal-shaped or heart-shaped domain.

A formal power series $\hat{f}$ is *k-summable* if it is k-summable in all the directions but finitely many modulo 2π. These exceptional directions are called the *singular directions* of $\hat{f}$. As in the case of Borel summability, if the singular directions are represented by $\theta_1 < \theta_2 < \cdots < \theta_m < \theta_{m+1} = \theta_1 + 2\pi$, they define the open arcs where the k-sums are compatible and glue together into sectors $S((\theta_i - \varepsilon, \theta_{i+1} + \varepsilon), R)$, with $\varepsilon > 0$ and $R > 0$ small enough. If an arc is wider than $(2 - 1/k)\pi$, this is defined on the Riemann surface of the logarithm. If there is no singular direction, $\hat{f}$ is convergent. In the sector $S((\theta_i - \varepsilon, \theta_i + \varepsilon), R)$ there are defined two k-sums, $\mathcal{S}_{k,\theta_i^-}(\hat{f})$ and $\mathcal{S}_{k,\theta_i^+}(\hat{f})$, that may differ, showing the Stokes phenomenon.

Let us revisit Example 24 for k-summability.

Example 25. Let us consider the series $\hat{f}_s = \sum_{n=0}^\infty (n!)^s x^n$ of Example 24. The k-Borel transform $\hat{g}_s = \sum_{n=0}^\infty (n!)^s \xi^n / \Gamma(n/k+1)$, using Stirling formula, is convergent for $s \leqslant 1/k$, and divergent for $s > 1/k$. For $s = 1/k$ we have the same behavior as in Example 23. For $s < 1/k$, $\hat{g}_s$ defines an entire function g_s with order $k/(1-sk)$. Hence, k-summability fails for $s > 1/k$ because k-Borel transform is "not strong

enough" to make $\hat{g}_s$ convergent, and it fails for $s < 1/k$ because Borel transform is "too strong" that it makes $\hat{g}_s$ "too convergent" at the expense of making its order too large for the convergence of Laplace transform. Thus, k-summation is tuned to divergence as rapid as $\hat{g}_{1/k}$.

In [Har49, §4.12] Hardy says "that, usually, the delicacy of a method decreases as its power increases, and that very powerful methods, adapted to the summation of rapidly divergent series, are apt to fail with divergent series of a less violent kind," what is illustrated in Example 25. In the formal solutions of a linear differential equation may appear not only k-summable series for any $k \in \mathbb{Q}_+$, but even combinations of k-summable series of different indices $k \in \mathbb{Q}_+$. For instance, $\hat{f}_1$ and $\hat{f}_2$ of Example 24 are solutions of a linear differential equation with polynomial coefficients, and $\hat{f}_1 + \hat{f}_2$ cannot be summed by k-summation for any $k > 0$, because $k < 2$ is not "powerful enough" to sum $\hat{f}_2$, and $k > 1$ is not "delicate enough" to sum $\hat{f}_1$. What we need is, in words of [MR91, p. 337], a "blend" of k-summations for different indices $k > 0$.

We say that a formal power series $\hat{f}$ is *integer-leveled multisummable* in the direction θ if it is the sum of finitely many k-summable power series in the direction θ of different integer levels $k > 0$. If $\hat{f} = \hat{f}_1 + \hat{f}_2 + \cdots + \hat{f}_r$ with each $\hat{f}_i$ k_i-summable in the direction θ and $k_1 > k_2 > \cdots > k_r > 0$, then we say that $\hat{f}$ is $(k_1, k_2, \ldots, k_r)$-summable in the direction θ and that its sum is $\mathcal{S}_\theta[\hat{f}] = \mathcal{S}_{k_1,\theta}[\hat{f}_1] + \mathcal{S}_{k_2,\theta}[\hat{f}_2] + \cdots + \mathcal{S}_{k_r,\theta}[\hat{f}_r]$. The decomposition is not unique, but the operator $\mathcal{S}_\theta$ is well defined and it is a homomorphism of differential algebras. The decomposition $\hat{f} = \hat{f}_1 + \hat{f}_2 + \cdots + \hat{f}_r$ is not effective, and for effective multisummation there are different processes explained in [Bal94]. Écalle's method applies the k_r-Borel transform, then successively applies certain integral transforms called *accelerations* that transition from k_i to k_{i-1}, and finally applies the k_1-Laplace transform. Balser's method applies the k_1-Borel transform, obtaining a $(k_2', k_3', \ldots, k_r')$-summable power series, with $k_i' = k_1 k_i/(k_1 - k_i)$, which is summed by induction, and ends with the k_1-Laplace transform. Écalle's and Balser's methods are equivalent and yield a definition of *multisummability* for general levels, but we only need integer levels for summing the series that appear in the Fabry solutions.

We say that $\hat{f}$ is *integer-leveled multisummable* if there exist integers $k_1 > k_2 > \cdots > k_r > 0$ such that $\hat{f}$ is $(k_1, k_2, \ldots, k_r)$-summable in all the directions but finitely many modulo 2π. It does not implies that $\hat{f} = \hat{f}_1 + \hat{f}_2 + \cdots + \hat{f}_r$ with each $\hat{f}_i$ k_i-summable; see [Bal94, p. 74] for a counterexample. For $r = 1$, (k_1)-summability is the usual (one-level) k_1-summability. For $r = 0$, it means that $\hat{f}$ is convergent. As in the case of one-level summability, if the singular directions are represented

by $\theta_1 < \theta_2 < \cdots < \theta_m < \theta_{m+1} = \theta_1 + 2\pi$, they define the open arcs where the k-sums are compatible and glue together into sectors $S((\theta_i - \varepsilon, \theta_{i+1} + \varepsilon), R)$, with $\varepsilon > 0$ and $R > 0$ small enough. If there is no singular direction, $\hat{f}$ is convergent. In the sector $S((\theta_i - \varepsilon, \theta_i + \varepsilon), R)$ there are defined two multisums, $\mathcal{S}_{\theta_i^-}(\hat{f})$ and $\mathcal{S}_{\theta_i^+}(\hat{f})$, that may differ, showing the Stokes phenomenon.

For any direction θ, $\mathcal{S}_\theta$ glues with the determination $(\theta - \pi, \theta + \pi)$ of the logarithm and gives a homomorphism of differential fields $\hat{K} \to \mathcal{O}_{0,\theta}$, for the $\hat{K}$ of §1.3.1, and a definition of *extended analytic continuation* in the irregular singular case. We may define *Stokes automorphism* at θ_i as $\mathrm{cont}_{(\beta,\delta)}^{-1} \circ \mathrm{cont}_{(\alpha,\gamma)}$ with $\theta_{i-1} < \alpha < \theta_i < \beta < \theta_{i+1}$ (understanding $\theta_0 = \theta_m - 2\pi$), γ the path formed by $0 \to \mathrm{e}^{\mathrm{i}\alpha}R$ followed by the arc $\mathrm{e}^{\mathrm{i}\alpha}R \rightsquigarrow \mathrm{e}^{\mathrm{i}\beta}R$ counterclockwise, δ the path $0 \to \mathrm{e}^{\mathrm{i}\beta}R$ and $R > 0$ smaller than the distance to any other singularity. This definition is independent of α, β and R. If we take other sequence of representatives of the singular directions of Δ, we would define other sequence of Stokes automorphism, but the Stokes automorphisms corresponding to the same direction modulo 2π are equal up to conjugation by the formal monodromy.

The multisummability of the formal series that appear in the Fabry solutions does not only have to do with the divergence of the series to sum, but also with the exponential parts. Let $\mathcal{Q}$ be the set of the polynomials $q(t)$ that appear in the exponential parts of the Fabry solutions in the form $\exp\left(q(x^{-1/p})\right)$, where p is the ramification index. We take $q = 0$ for solutions without exponential part. The set $\mathcal{Q}' = \{q_i - q_j : q_i, q_j \in \mathcal{Q}, q_i \neq q_j\}$ determines the singular directions and the levels of multisummability. The set of the degrees of $\mathcal{Q}'$ is the set of levels of multisummability. Each $q(t) \in \mathcal{Q}'$, of degree k, defines a pattern of sectors of opening $\pi p/k$, alternatively of growth and decay of $\exp\left(q(x^{-1/p})\right)$ at the origin. If α is an argument of the leading coefficient of $q(t)$, then the sectors of exponential growth or decay are respectively those of positive or negative sign of $\cos(\arg(z)k/p + \alpha)$, defined in the Riemann surface of the logarithm. In the rays $\cos(\arg(z)k/p + \alpha) = 0$, $\exp\left(q(x^{-1/p})\right)$ has an oscillatory behavior near the origin. These rays are called *Stokes lines*,[10] and their directions *Stokes directions*. The bisectrices of these sectors are called *anti-Stokes lines* and *anti-Stokes directions*, although other people use the opposite terminology.[11] The singular directions θ

[10] H. Żołądek calls them *rays of division*, which is descriptive and avoids the confusion explained below, but unfortunately it seems than only Żołądek uses this terminology.

[11] I follow this convention because it is the terminology used in most of my references (Loday-Richaud [LR94, def. I.4.5], Malgrange-Ramis [MR92, p. 363], van der Put–Singer [vdPS03, def. 7.18], Martinet-Ramis [MR91, p. 357, n. 38], Ramis-Martinet [RM90, p. 180]) with the exception of van der Hoeven [vdH07a, §2.4]. Ramis thinks that this usage is improper, but uses it anyway; cf. [Ram04, p. 73]. Reading Stokes [Sto1902], I agree with Ramis because the only

associated with $q(t)$ are those satisfying $\cos(\theta k/p+\alpha) = -1$, showing the maximal decay, so they are anti-Stokes directions. The rest of the anti-Stokes directions associated with $q(t)$ are the singular directions associated with $-q(t)$, which also belongs to $\mathcal{Q}'$, so I shall speak of singular directions instead of speaking of anti-Stokes directions, in order to avoid confusion. Not all the singular directions of Δ are singular directions of a solution, as shown in Example 26, but all the singular directions of the solutions are singular directions of the equation.

Example 26. Revisiting Example 23, the Euler series $\hat{f} = \sum_{n=0}^{\infty}(-1)^n n!\, x^{n+1}$ is a solution of the inhomogeneous differential equation $x^2y' + y = x$, and thus of the homogeneous equation $x^3y'' + (x^2 + x)y' - y = 0$. A complete system of solutions of the latter is $\{\hat{f}, \exp(1/x)\}$. There are two exponential parts, 1 and $\exp(1/x)$, so $\mathcal{Q} = \{0, x\}$ and $\mathcal{Q}' = \{x, -x\}$. The singular lines of the equation are both real semi-axes, but the only singular line of the solutions is the negative semi-axis.

Let θ be a singular direction and σ_θ the Stokes automorphism in this direction. Let $\mathcal{Q}'_\theta = \{q \in \mathcal{Q}' : \theta \text{ singular direction associated with } q\}$ and $\mathcal{Q}^2_\theta = \{(q_i, q_j) \in \mathcal{Q}^2 : q_i - q_j \in \mathcal{Q}'_\theta\}$. According to [MR91, p. 361f], $\mathcal{Q}^2_\theta$ can be completed to a total order $\preceq$ on $\mathcal{Q}$, with associated strict order $\prec$, so that $\mathcal{Q}$ can be listed as $q_1 \prec q_2 \prec \cdots \prec q_s$. If V_i is the subspace of solutions corresponding to the exponential part $\exp(q_i)$, the decomposition in direct sum $V_1+V_2+\cdots+V_s$ induces a block structure in the matrix $\mathtt{S}_\theta$ of σ_θ such that the block corresponding to $V_i \to V_j$ is the identity if $i = j$ and zero if $i < j$, so that $\mathtt{S}_\theta$ is the identity plus a strictly triangular (and hence nilpotent) matrix, thus it is unipotent. As told in §3.2, $\mathtt{S}_\theta$ generates a connected algebraic group whose Lie algebra is generated by $\log \mathtt{S}_\theta = \sum_{k=0}^{\infty}(-1)^k(\mathtt{S}_\theta - \mathtt{I})^{k+1}/(k+1)$, where the sum is finite. Hence $\log \sigma_\theta = \sum_{k=0}^{\infty}(-1)^k(\sigma_\theta - \mathrm{id})^{k+1}/(k+1)$ is a finite sum and defines a Galoisian derivation $\dot{\Delta}_\theta$, called the *alien derivation* in the direction θ. Such a derivation admits a "Fourier decomposition" as a sum of the contribution of each $q \in \mathcal{Q}'_\theta$, having $\dot{\Delta}_\theta = \sum_{q\in\mathcal{Q}'_\theta} \dot{\Delta}_{q,\theta}$, according to [MR91, p. 385]. These components are also alien derivations and are Galoisian, but we do not need such a refinement for Ramis density theorem.

For picturing it geometrically, we may "blow up" the origin, substituting a disk for it, as shown in Figure 1.1. We may understand the circle as the support of the $\mathcal{O}_{0,d}$, and its center as the support of $\hat{K}$. Figure 1.2 shows the ways of "coming out" of the singularity and connect to the Galois groupoid. We may consider the ends of singular directions after blowing up as singularities in the sense of the monodromy. Indeed, as Figure 1.3 shows, the monodromy around the singularity splits and its factors are the monodromies around the singularities in the circle

directions Stokes mentions are the so-called anti-Stokes directions.

and around the center. The monodromy around the center represents the formal monodromy, and the monodromy around a singularity in the circle represents the Stokes automorphism at this direction. These geometric interpretations are explained in [RM90, p. 181].

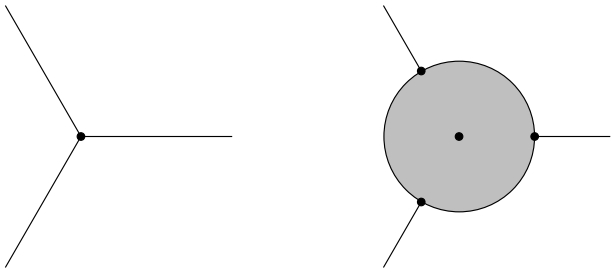


Figure 1.1: Blowing up the singularity.

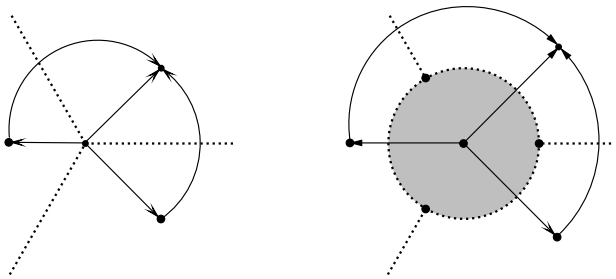


Figure 1.2: The different ways of coming out of the singularity.

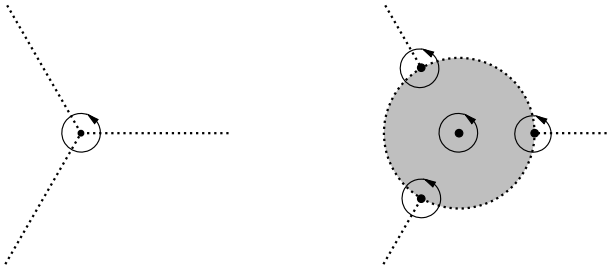


Figure 1.3: The monodromy and the extended monodromy.

1.4.4 Density theorems

Let us study the fixed fields of the automorphisms defined above. Fixed by the exponential torus means free of exponentials. Fixed by (1.9) means free of logarithms. Fixed by (1.8) means that the only x^α allowed are for $\alpha \in \mathbb{Z}$. Fixed by all the Stokes automorphisms means all the power series are convergent. So, fixed by all of them means convergent Laurent series, thus the algebraic group generated by them is the Galois group of $\hat{K}/\mathbb{C}(\{x\})$. These ideas prove Ramis's local theorem, cf. [MR91, thm. 20].

Theorem 27 (Ramis)**.** *The Galois group of a homogeneous linear differential equation over the germs of meromorphic functions is the smallest algebraic group containing the formal monodromy, the exponential torus and the Stokes automorphisms.*

If we "bring" the local groups to a common point, assuming X a connected Riemann surface, the fixed field is $\mathcal{M}(X)$. This is the Ramis's global theorem, cf. [MR91, thm. 21].

Theorem 28 (Ramis)**.** *The Galois group of a homogeneous linear differential equation over a connected Riemann surface is generated by the local groups at all the singularities.*

The algebraic tori are connected, hence the exponential torus and the monodromy one. The algebraic group $G_{\log}$, isomorphic to the additive group of constants, is also connected. Each Stokes automorphism is unipotent, see [MR91, §4], and generates another algebraic group isomorphic to the additive group of constants, and thus connected. Therefore, the monodromy group meets any component of the Galois group.

1.5 Finding Liouvillian solutions

Liouvillian integrability is the differential counterpart of solvability by radicals, as the following theorem states.

Theorem 29 (Kolchin)**.** *Let K be a differential field with algebraically closed field of constants $\mathbb{K}$. Let Δ be a homogeneous linear differential equation with coefficients in K. All the solutions of Δ are Liouvillian over K if and only if* $\mathrm{Gal}(\Delta)^\circ$ *is solvable.* [vdPS03, thm. 1.43]

According to Lie-Kolchin Theorem, stated below, the solvability of $\mathrm{Gal}(\Delta)^\circ$ is equivalent to its triangularizability.

Theorem 30 (Lie-Kolchin)**.** *A solvable connected linear algebraic group is triangularizable. In particular, it admits an invariant line.* [Kap76, thm. 4.11]

So, if all the solutions of Δ are Liouvillian over K, then $\mathrm{Gal}(\Delta)^\circ$ has an invariant line of solutions $\mathbb{K}y$, for certain solution $y \neq 0$ of Δ. In this case, y'/y is algebraic over K, so y is a Liouvillian solution of a very special kind. What happens if Δ has some non-zero Liouvillian solutions, but not necessarily all of them? After some technicalities, this case can be reduced to the previous one, proving the following result.

Theorem 31. *Let K be a differential field with algebraically closed field of constants. Let Δ be a homogeneous linear differential equation with coefficients in K. Let F be a Picard-Vessiot extension of K for Δ. If Δ admits a non-zero solution Liouvillian over K, then it has a non-zero solution $y \in F$ such that y'/y is algebraic over K.*

This result is assumed as known by M. F. Singer in his proof of [Sin81, thm. 2.4], which is a stronger theorem that bounds the degree of y'/y over K in terms of the order of Δ alone. Next subsection is devoted to this stronger result.

1.5.1 A theorem of Singer

The following theorem about Liouvillian solutions of differential equations is the most important for our purpose.

Theorem 32 (Singer)**.** *Let K be a differential field with algebraically closed field of constants. Let Δ be a homogeneous linear differential equation of order r with coefficients in K. Let F be a Picard-Vessiot extension of K for Δ. If Δ admits a non-zero solution Liouvillian over K, then it has a non-zero solution $y \in F$ such that y'/y is algebraic over K of degree $I(r)$ at most, for the function I of Theorem 33.* [Sin81, thm. 2.4]

Let $P \in K[x]$ be the minimal polynomial of such a quotient y'/y. The group $\mathrm{Gal}(\Delta)$ permutes the roots of P. The stabilizer H of y'/y has index equal to the degree of P. We have that y is a common eigenvector of H. Conversely, if $u \neq 0$ is a solution of Δ that is a common eigenvector u of $H' < \mathrm{Gal}(\Delta)$ with

$[\mathrm{Gal}(\Delta) : H'] = k$, then u'/u is algebraic over K of degree k. This correspondence explains the following group-theoretical result that defines the function I.

Theorem 33 (Singer)**.** *There exists a function $I : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n and any field $\mathbb{K}$ algebraically closed, every subgroup G of $\mathrm{GL}(n, \mathbb{K})$ with a 1-reducible subgroup of finite index admits a 1-reducible subgroup of index $I(n)$ at most.* [Sin81, prop. 2.2]

A linear group is called *m-reducible* if it has an invariant subspace of dimension m. In particular, a linear group is 1-reducible if it has a common eigenvector. The proof of Theorem 33 uses a theorem of Jordan that he proved in [Jor1877] for a weaker version of Theorem 32.

Theorem 34 (Jordan)**.** *There exists a function $J : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n, every finite subgroup G of $\mathrm{GL}(n, \mathbb{C})$ admits an abelian normal subgroup of index $J(n)$ at most.*

Jordan's proof does not control the growth of J. A further result of Schur's gives an explicit bound

$$J_{\mathrm{Schur}}(n) = \left(\sqrt{8n} + 1\right)^{2n^2} - \left(\sqrt{8n} - 1\right)^{2n^2},$$

which satisfies $\log J_{\mathrm{Schur}}(n) = \mathcal{O}(n^2 \log n)$ asymptotically. The proof can be found in [CR62, §36]. Blichfeldt refined the bound; in [Dor71, §30] we find

$$J_{\mathrm{Blich}}(n) = 6^{(n-1)(\pi(n+1)+1)} n!,$$

where π is the prime-counting function, which satisfies

$$\log J_{\mathrm{Blich}}(n) = \mathcal{O}(n^2 / \log n).$$

This growth order was improved after the classification of finite simple groups. Weisfeiler announced in [Wei84] a bound that satisfies

$$\log J_{\mathrm{Weis}}(n) = \mathcal{O}(n \log^3 n),$$

but unfortunately he disappeared in the Andes and his work kept unfinished and unpublished.

Collins proved in [Col07] that $J_0(n) = (n+1)!$ is the optimal bound for $n \geqslant 71$. It satisfies $\log J_0(n) = \mathcal{O}(n \log n)$ asymptotically. This bound is achieved by the symmetric group in $n+1$ elements contained in $\mathrm{GL}(n, \mathbb{C})$. The permuted elements are $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n$ of the standard basis and $-\mathbf{e}_1 - \mathbf{e}_2 - \cdots - \mathbf{e}_n$.

Remark 35. Although Jordan's original statement is in the complex field, the only property of $\mathbb{C}$ he uses is, according to [Bre11, §2], that every finite-order matrix is diagonalizable. In particular, it is valid for any algebraically closed field $\mathbb{K}$ (of characteristic 0). Other proofs and bounds are specific to the complex field; for instance, the proof in [CR62, §36] reduces to the unitary case and the bound computes volumes. Even the proofs and bounds that uses non-algebraic properties of $\mathbb{C}$ are valid for any algebraically closed field $\mathbb{K}$ thanks to the following trick that reduces to the complex field. Any given finite subgroup G of $\mathrm{GL}(n, \mathbb{K})$ is defined over the field generated by the entries of its members. Let $\mathbb{K}_0$ be the algebraic closure of this field. As $\mathbb{K}_0$ has a finite degree of transcendence d, it is isomorphic to $\overline{\mathbb{Q}(X_1, X_2, \ldots, X_d)}$, so it can be embedded in $\mathbb{C} \simeq \overline{\mathbb{Q}(X_t : t \in \mathbb{R})}$ and thus G is defined over $\mathbb{C}$.

1.5.2 Detailed proof of Theorem 33

I shall present a proof of Theorem 33 using the same technique as Singer did, but keeping a finer track of the bounds than he needed for J_{Schur}. I shall start with a weaker form of Theorem 34:

Theorem 36. *There exists a function $J_{\mathrm{prim}} : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n, every **primitive finite** subgroup G of $\mathrm{GL}(n, \mathbb{C})$ admits an abelian normal subgroup of index $J_{\mathrm{prim}}(n)$ at most.*

A *system of imprimitivity* of a subgroup G of $\mathrm{GL}(V)$ is a decomposition $V = V_1 \oplus V_2 \oplus \cdots \oplus V_m$ such that no V_i is zero and the action of G permutes the family $\{V_1, V_2, \ldots, V_m\}$. If G admits a system of imprimitivity with $m > 1$, G is called *imprimitive*. If G is irreducible (its only invariant subspaces are V and 0) and not imprimitive, G is called *primitive*.

The optimal value of J_{prim} is given in [Col08, thm. A]; it is $J_{\mathrm{prim}}(n) = (n+1)!$ with the following exceptions: $J_{\mathrm{prim}}(2) = 60$, $J_{\mathrm{prim}}(3) = 360$, $J_{\mathrm{prim}}(4) = J_{\mathrm{prim}}(5) = 5!\,6^3$, $J_{\mathrm{prim}}(6) = 7!\,6^4$, $J_{\mathrm{prim}}(7) = 8!\,6^2$, $J_{\mathrm{prim}}(8) = 10!\,96$, $J_{\mathrm{prim}}(9) = 6^7 15$ and $J_{\mathrm{prim}}(12) = 13!\,72$. Collins's statement is for $\mathbb{C}$, but the trick in Remark 35 allows to extend both the result and the bounds to any algebraically closed field:

Theorem 37. *There exists a function $J_{\mathrm{prim}} : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n and any field $\mathbb{K}$ algebraically closed, every primitive finite subgroup G of $\mathrm{GL}(n, \mathbb{K})$ admits an abelian normal subgroup of index $J_{\mathrm{prim}}(n)$ at most.*

With $I_{\mathrm{prim}} = J_{\mathrm{prim}}$, we have a weaker version of Theorem 33:

Theorem 38. *There exists a function $I_{\mathrm{prim}} : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n and any field $\mathbb{K}$ algebraically closed, every primitive finite subgroup G of $\mathrm{GL}(n, \mathbb{K})$ admits a **1-reducible** subgroup of index $I_{\mathrm{prim}}(n)$ at most.*

The bound $I_{\mathrm{prim}} = J_{\mathrm{prim}}$ is too rough. According to [Cor01, Chap. 4], we have the bounds $I_{\mathrm{prim}}(2) = 12$, $I_{\mathrm{prim}}(3) = 36$, $I_{\mathrm{prim}}(4) = 120$ and $I_{\mathrm{prim}}(5) = 55$, but $I_{\mathrm{prim}}(6) \geqslant 3780$. Let us assume $I_{\mathrm{prim}}(n) = J_{\mathrm{prim}}(n)$ for $n \geqslant 6$. I shall proceed with the proof of Theorem 33 by proving a chain of weaker versions starting with Theorem 38.

Proposition 39. *There exists a function $I_{\mathrm{fin}} : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n and any field $\mathbb{K}$ algebraically closed, every **finite** subgroup G of $\mathrm{GL}(n, \mathbb{K})$ admits a 1-reducible subgroup of index $I_{\mathrm{fin}}(n)$ at most.*

Proof. Let $\mathcal{V}$ be a system of imprimitivity of G of maximal length. Pick $V_0 \in \mathcal{V}$. Let $\mathcal{V}'$ be the orbit of V_0 by the action of G. The stabilizer H of V_0 has a natural representation $H \to \mathrm{GL}(V_0)$ given by the restriction; let K be the image of this representation.

Let $\mathcal{W}$ be a system of imprimitivity of K. Pick representatives $\mathcal{A} \subset G$ of G/H, the left cosets of H. The family $\mathcal{W}' = \{\mathtt{A}W : \mathtt{A} \in \mathcal{A}, W \in \mathcal{W}\}$ yields the direct sum $\bigoplus_{W \in \mathcal{W}'} W = \bigoplus_{V \in \mathcal{V}'} V$ and has $\#\mathcal{W}' = (\#\mathcal{V}')(\#\mathcal{W})$. Thus $\mathcal{W}' \cup (\mathcal{V} \setminus \mathcal{V}')$ is a system of imprimitivity of G of length $\#\mathcal{V} + (\#\mathcal{V}')(\#\mathcal{W} - 1)$. As $\mathcal{V}$ has maximal length and $\mathcal{V}' \neq \varnothing$, necessarily $\#\mathcal{W} = 1$. Therefore, the only system of imprimitivity of K is $\mathcal{W} = \{V_0\}$.

Suppose that K has an invariant subspace W different from 0 and V_0. As K is finite, according to Maschke's Theorem [FH91, prop. 1.5], there is another invariant subspace W' of K complementary to W. Hence we have a system of imprimitivity $\{W, W'\}$ of K, in contradiction with the previous paragraph. Therefore K is irreducible.

From the two last paragraphs we conclude that K is primitive. According to Theorem 38, K admits a 1-reducible subgroup K' of index $I_{\mathrm{prim}}(\dim V_0)$ at most. The preimage H' of K' is 1-reducible and satisfies

$$[G : H'] = [G : H][H : H'] = \#(\mathcal{V}')[K : K'] \leqslant \#(\mathcal{V}')I_{\mathrm{prim}}(\dim V_0).$$

The vector space spanned by $\mathcal{V}'$ has dimension $\#(\mathcal{V}') \dim V_0 \leqslant n$, thus

$$I_{\mathrm{fin}}(n) = \max\{rI_{\mathrm{prim}}(s) : rs \leqslant n\}$$

makes the claim true. □

Proposition 40. *There exists a function $I_{\mathrm{sca}} : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n and any field $\mathbb{K}$ algebraically closed, every subgroup G of $\mathrm{GL}(n, \mathbb{K})$ with a **scalar** subgroup of finite index admits a 1-reducible subgroup of index $I_{\mathrm{sca}}(n)$ at most.*

Proof. Let N be such a scalar subgroup of G with $[G : N]$ finite. Thus $G' = (\mathbb{C}^* G) \cap \mathrm{SL}(n, \mathbb{C})$ is finite. According to Proposition 39, G' admits a 1-reducible subgroup H' of index $I_{\mathrm{fin}}(n)$ at most. The group $H = (\mathbb{C}^* H') \cap G$ is 1-reducible, and a subgroup of G with $[G : H] \leqslant [G' : H'] \leqslant I_{\mathrm{fin}}(n)$, therefore $I_{\mathrm{sca}} = I_{\mathrm{fin}}$ makes the claim true. □

Proposition 41. *There exists a function $I_{\mathrm{norm}} : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n and any field $\mathbb{K}$ algebraically closed, every subgroup G of $\mathrm{GL}(n, \mathbb{K})$ with a 1-reducible **normal** subgroup of finite index admits a 1-reducible subgroup of index $I_{\mathrm{norm}}(n)$ at most.*

Proof. Let N be such a 1-reducible normal subgroup of G with $[G : N]$ finite. Let $\mathcal{V} = \{V_1, V_2, \ldots, V_m\}$ be the family of maximal eigenspaces of N, following the terminology of [Sin81, prop. 2.2], whose sum is direct and hence $\sum_{i=0}^{m} \dim V_i \leqslant n$. The left action of G permutes $\mathcal{V}$ because $N \lhd G$. The stabilizer K of V_1 contains N and has a natural representation $K \to \mathrm{GL}(V_1)$. Let K' and N' be the respective images of K and N. As N' is scalar and $[K' : N'] = [K : N]$ if finite, according to Proposition 40, K' admits a 1-reducible subgroup H' of index $I_{\mathrm{sca}}(\dim V_1)$ at most. The preimage H of H' is 1-reducible and satisfies

$$[G : H] = [G : K][K : H] = \#(\mathcal{V}')[K' : H'],$$

where $\mathcal{V}'$ is the orbit of V_1, hence

$$[G : H] \leqslant m I_{\mathrm{sca}}(\dim V_1).$$

As $m \dim V_1 \leqslant n$,

$$I_{\mathrm{norm}}(n) = \max\{r I_{\mathrm{sca}}(s) : rs \leqslant n\}$$

makes the claim true. □

Proposition 42 (=Theorem 33)**.** *There exists a function $I : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ such that, for each n and any field $\mathbb{K}$ algebraically closed, every subgroup G of $\mathrm{GL}(n, \mathbb{C})$ with a 1-reducible subgroup of finite index admits a 1-reducible subgroup of index $I(n)$ at most.*

Proof. Let H be such a 1-reducible subgroup of G with $[G : H]$ finite. The left action of G permutes G/H, the left cosets of H, and gives a natural representation

$G \to \mathrm{Sym}(G/H)$. Its kernel K is contained in H and is thus 1-reducible. As $K \lhd G$ and $[G : K] \leqslant \#\,\mathrm{Sym}(G/H) = [G : H]!$, according to Proposition 41, G admits a 1-reducible subgroup H' of index $I_{\mathrm{norm}}(n)$ at most. Thus $I = I_{\mathrm{norm}}$ makes the claim true. □

Tracking the bounds of the previous theorems, for any bound I_{prim} of Theorem 38,

$$I(n) = \max\{rI_{\mathrm{prim}}(s) : rs \leqslant n\} \tag{1.11}$$

makes the claim of Proposition 42 true. Computing according to (1.11), we get the following values: $I(2) = 12$, $I(3) = 36$, $I(4) = I(5) = 120$, $I(6) = I(7) = 7!\,6^4$, $I(8) = I(9) = I(10) = 10!\,96$, $I(11) = 12!$, $I(12) = I(13) = 13!\,72$ and $I(14) = 15!$. If $s \leqslant 14$ and $n \geqslant 15$ in (1.11), $rI_{\mathrm{prim}}(s) \leqslant 15!\,r \leqslant (n+1)!$. If $s \geqslant 14$, $rI_{\mathrm{prim}}(s) = r(s+1)! \leqslant (n+1)!$. This proves that $I(n) = (n+1)!$ for $n \geqslant 14$.

1.5.3 A variant of Singer's theorem

Theorem 32 is stated for differential operators, but it is valid for differential systems in the following form, though the classic algorithms are stated for scalar equations. As we will need in this work a precise form of this result for differential systems, we present here a complete proof.

Theorem 43. *Let K be a differential field with algebraically closed field of constants. Let Δ be an $n \times n$ explicit differential system of order r with coefficients in K. Let F be a Picard-Vessiot extension of K for Δ. If Δ has a non-zero solution Liouvillian over K, then there exist an intermediate differential field F_0 and a non-zero solution $(y_1, y_2, \dots, y_n)^\intercal \in F^n$ of Δ such that F_0/K is an algebraic extension of degree $I(rn)$ at most and, for each i and j with $y_i \neq 0$, $y_i'/y_i \in F_0$ and $y_j/y_i \in F_0$.*

Proof. Write

$$\Delta : \quad \mathbf{y}^{(r)} = \mathtt{A}_0\mathbf{y} + \mathtt{A}_1\mathbf{y}' + \cdots + \mathtt{A}_{r-1}\mathbf{y}^{(r-1)},$$

with $\mathtt{A}_0, \mathtt{A}_1, \dots, \mathtt{A}_{r-1} \in K^{n\times n}$. Let $\mathbf{u}' = \mathtt{B}\mathbf{u}$ be the companion system of Δ, with[12] $\mathbf{u} = \left(\underline{\mathbf{y}}, \underline{\mathbf{y}'}, \dots, \underline{\mathbf{y}^{(r-1)}}\right)^\intercal$. By virtue of Cyclic Vector Lemma,[13] $\mathbf{u}' = \mathtt{B}\mathbf{u}$ is equivalent to a scalar equation

$$\Delta_0 : \quad v^{(rn)} = a_0 v + a_1 v' + \cdots + a_{rn-1} v^{(rn-1)},$$

[12] This notation is introduced on page 86 and explained in footnote 1. Here, an underlined column vector is the (horizontal) list of its entries.

[13] See §1.1.4.

with $a_0, a_1, \dots, a_{r-1} \in K$ and $\mathtt{P}\mathbf{u} = (v, v', \dots, v^{(rn-1)})^\intercal$.

If Δ has a non-zero solution Liouvillian over K, the associated solution of Δ_0 is also non-zero and Liouvillian over K. According to Theorem 32, Δ_0 has a non-zero solution $v_0 \in F$ such that v_0'/v_0 is algebraic over K of degree $I(rn)$ at most. The differential field $F_0 = K(v_0'/v_0)$ is an intermediate field of F/K. Moreover F_0/K is an algebraic extension of degree $I(rn)$ at most. By induction one may prove that $v_0^{(k)} \in v_0 F_0$, hence the associated solution of $\mathbf{u}' = \mathtt{B}\mathbf{u}$ is

$$\mathbf{u}_0 = \mathtt{P}^{-1}(v_0, v_0', \dots, v_0^{(rn-1)})^\intercal \in v_0 F_0^{rn}$$

and the associated solution $\mathbf{y}_0 = (y_1, y_2, \dots, y_n)^\intercal$ of Δ belongs to $v_0 F_0^n$.

Clearly $\mathbf{y}_0 \neq \mathbf{0}$. It is easy to prove that $\mathbf{y}_0'$ belongs to $v_0 F_0^n$. Therefore, for each i and j with $y_i \neq 0$, y_i'/y_i and y_j/y_i belong to F_0. □

1.5.4 Singerian solutions

I will call the solutions given by Theorem 43, which generalizes Theorem 32, *Singerian* solutions. These are the solutions we will look for in this thesis. In the analytic case, the Singerian solutions have a special Fabry form

$$\exp\left(q(x^{-1/p})\right) x^\alpha \left(f_1(x^{1/p}), f_2(x^{1/p}), \dots, f_n(x^{1/p})\right)^\intercal \tag{1.12}$$

with $p > 0$ integer, q a polynomial with $q(0) = 0$ and $f_1, f_2, \dots, f_n$ convergent series. These solutions are free of logarithms and divergent series, and have common exponential and potential parts. The following theorem formalizes this claim.

Theorem 44. *Let Δ be an $n \times n$ explicit differential system of order r over a connected Riemann surface X. Let us fix $z \in X$ and a local chart at z, so that $\mathcal{M}_z \simeq \mathbb{C}(\{x\})$. Let K be the image of $\mathcal{M}(X)$ in $\mathbb{C}(\{x\})$, and F/K the Picard-Vessiot extension for Δ taken in the universal field extension at z defined in §1.3.1. A Singerian solution of Δ in F is of the form (1.12).*

Proof. Let $(y_1, y_2, \dots, y_n)^\intercal \in F^n$ be the Singerian solution, and i an index with $y_i \neq 0$. In particular y_i'/y_i is algebraic over K, and thus over $\mathbb{C}(\{x\})$, therefore it belongs to $\mathbb{C}(\{x^{1/p_i}\})$ for certain integer $p_i > 0$. Decompose $y_i'/y_i = g_{i-} + \alpha_i x^{-1} + g_{i+}$ putting the terms of valuation less than -1 in g_{i-} and the terms of valuation greater than -1 in g_{i+}. There exists a polynomial q_i such that $q_i(0) = 0$ and $q_i(x^{-1/p_i})' = g_{i-}$. There exists $h_i \in \mathbb{C}\{t\}$ such that $h_i(x^{1/p_i})' = g_{i+}$. The composition $H_i =$

$\exp \circ h_i$ lies in $\mathbb{C}\{t\}$, and $H_i(x^{1/p_i})' = H_i(x^{1/p_i})\, g_{i+}$. It is straightforward that $Y_i = \exp\left(q_i(x^{-1/p_i})\right) x^{\alpha_i} H_i(x^{1/p_i})$ is a solution of $y_i Y' = y_i' Y$, thus $Y_i = \beta_i y_i$ for certain $\beta_i \in \mathbb{C}$. As $Y_i \neq 0$, $\beta_i \neq 0$ and $y_i = \exp\left(q_i(x^{-1/p_i})\right) x^{\alpha_i} H_i(x^{1/p_i})/\beta_i$.

For each i and j such that $y_i \neq 0$ and $y_j \neq 0$, y_i/y_j is algebraic over K, and thus over $\mathbb{C}(\{x\})$, therefore

$$y_i/y_j = \exp\left(q_i(x^{-1/p_i}) - q_j(x^{-1/p_j})\right) x^{\alpha_i - \alpha_j} H_i(x^{1/p_i})/H_j(x^{1/p_j})(\beta_j/\beta_i),$$

and necessarily $q_i(x^{-1/p_i}) = q_j(x^{-1/p_j})$ and $\alpha_i - \alpha_j \in \mathbb{Q}$. Let p be the common denominator of the $1/p_i$ and the $\alpha_i - \alpha_j$, and q the polynomial such that $q(x^{-1/p}) = q_i(x^{-1/p_i})$ for any i. Let α be the minimum of the α_i. For each i, we may write $y_i = \exp\left(q(x^{-1/p})\right) x^{\alpha} f_i(x^{1/p})$ for certain $f_i \in \mathbb{C}\{t\}$. □

It suffices to find a single non-zero Liouvillian solution because the classic d'Alembert reduction method reduces the problem to lower order. For a solution $f \neq 0$ of a scalar equation

$$a_r y^{(r)} + a_{r-1} y^{(r-1)} + \cdots + a_0 y = 0,$$

we apply the change of variable $y = f \int u$ and get

$$b_{r-1} u^{(r-1)} + b_{r-2} u^{(r-2)} + \cdots + b_0 u = 0.$$

Notice that, if $a_0, a_1, \ldots, a_r$ lie in a differential field K, then $b_0, b_1, \ldots, b_{r-1}$ lie in $F = K(f'/f)$. If f is a solution given by Theorem 32, then F/K is an algebraic extension of degree $I(r)$ at most. A generalization for differential systems is found in [CL72, pp. 71–73]. For a system $\mathbf{y}' = \mathtt{A}\mathbf{y}$, with $\mathtt{A} \in K^{n\times n}$, and a particular solution $\mathbf{f} = (f_1, f_2, \ldots, f_n)^{\intercal}$, with $f_1 \neq 0$,[14] the reduced system $\mathbf{u}' = \mathtt{B}\mathbf{u}$ has $\mathtt{B} \in F^{(n-1)\times(n-1)}$ with

$$F = K(f_2/f_1, f_3/f_1, \ldots, f_n/f_1).$$

If $\mathbf{f}$ is a Singerian solution, then F/K is an algebraic extension of degree $I(n)$ at most.

1.5.5 Review of classic methods on Liouvillian solutions

The first complete algorithm for computing the Liouvillian solutions of a differential equation is Kovacic's, published in [Kov86], valid for second-order equations

[14] If $\mathbf{f} \neq \mathbf{0}$ but $f_1 = 0$, we reorder the variables, so we proceed without loss of generality.

over the Riemann sphere. Kovacic algorithm relies on the classification of the subgroups of $\mathrm{SL}(2,\mathbb{C})$, which is finer than the value $I(2) = 12$. If a second-order equation have non-zero Liouvillian solutions, it has a solution $y \neq 0$ with y'/y an algebraic function of degree 1, 2, 4, 6 or 12. The Kovacic algorithm tries sequentially these options. It computes all the possible principal parts of the coefficients of the minimal polynomial of y'/y at all the singularities of the equation and tries to glue them into a minimal polynomial.

An alternative to Kovacic algorithm is Ulmer-Weil's, published in [UW96], valid for the same kind of equations. Ulmer-Weil algorithm uses the symmetric powers of the differential operator. The m-th *symmetric power* of a differential operator L is another operator $L^{\circledS m}$ whose solution space is spanned by the products $y_1 y_2 \cdots y_m$ of solutions y_i of L. If a second-order operator L have non-zero Liouvillian solutions, either it has a solution $y \neq 0$ with y'/y a rational function or $L^{\circledS m}$ has a rational solution for $m \in \{1, 2, 4, 6, 8, 12\}$. A rational solution of $L^{\circledS m}$ gives a solution of L algebraic of degree m. The advantage of this method is that it reduces the problem of finding Liouvillian solutions almost to the problem of finding rational solutions of linear differential equations.

Theorem 32 is proved in [Sin81] as an auxiliary result for an algorithm for computing the Liouvillian solutions of a differential operator L of any order r over rational functions. Singer algorithm uses the symmetric powers $L^{\circledS m}$ up to $m = I(r)$. There are algorithms fit to third-order equations, and [Cor01] makes feasible algorithms for order 4 and 5, but the bound $I_{\mathrm{prim}}(6) \geqslant 3780$ leads to a symmetric power of order greater than 10^{15}.

All these algorithms are completely algebraic and implementable in symbolic computation. The aim of this thesis is to give an algorithm for computing the Liouvillian solutions of an explicitable differential system over the Riemann sphere of any order and size. This algorithm will not use symbolic computation alone, but a symbiosis of symbolic computation and exact numerics as explained in the following chapters.

HERE ENDETH THE FIRSTE CHAPTER ✠

Chapter 2

Effective numerics

The framework of this thesis is hybrid numeric-symbolic computation, which combines symbolic computation with "exact" numerical computation in a way that will be described in the following chapters. This chapter is devoted to introduce the second ingredient, effective numerics, including some details of its implementation in C++. Likewise, we shall study certain problems that are source of difficulties we will have to deal with in the following chapters. These problems come from the computation of the rank of a complex matrix and from the computation of the rank over $\mathbb{Q}$ of a subset of $\mathbb{C}$, including a test of rationality. We shall study the direction of the errors (from above or from below) we will have to deal with in the rest of this work.

In the first section I introduce the effective complex numbers, their implementation and their operations. The second section is devoted to the computation of the Ramis generators of the Galois groups of a differential equation or system, including the effective numeric methods necessary for such a computation. Then I study the errors of the computation of the rank over $\mathbb{C}$ (§2.3) and over $\mathbb{Q}$ (§2.4). In the last section I discuss the global parameter introduced in this chapter and give a framework for the global parameters to be introduced in the following chapters. These considerations will be necessary in §4.5, where we will study the main algorithm of this thesis.

2.1 Effective complex numbers

In this section, I shall introduce effective complex numbers after some considerations on arbitrary precision and object-oriented programming. Then I continue with the operations with such numbers, including the computation of roots of polynomials.

2.1.1 Arbitrary precision

The usual computational implementation of arithmetic is in hardware. This implementation is efficient but fixed-precision. The implementation of natural numbers is straightforward in a binary computer. With a fixed precision of n bits, one represent the numbers from 0 through $2^n - 1$. The computations are exact if the result is less than 2^n; otherwise there will be an error. The device of [Luc10, cap. 7] and [Wel01] allows us to implement arbitrary-precision numbers by software. The idea is using a pair $(n, \mathbf{v})$ with $\mathbf{v} = (v_0, v_1, \ldots, v_{n-1})$. Both n and each v_i are numbers implemented by hardware. In the programming language C,[1] n would be of type `size_t` and $\mathbf{v}$ an array of `unsigned int`. Mathematically speaking, this device limits the size of the numbers, but this size bound is the restriction for allocatable memory, so we would get a "run out of memory" message before reaching it. The arithmetic of unbounded natural numbers may be implemented in a Turing machine.

Remark 45. A historical note:
This device of extending fixed-precision integer arithmetic is found in Archimedes's Sand-Reckoner.[2] In this case, the fixed precision of the Greeks was eight decimal digits, which Archimedes called the *first order*. He defined the nth order whose unit is 10^8 units of the $(n-1)$th order. Each order ranged from 1 to 10^8 of its units. This way he developed a system of numeration in base 10^8. Archimedes admitted up to 10^8 digits in base 10^8, which he called the *first period*. He went further, defining the nth period with unit 10^8 units of the 10^8th order of the $(n-1)$th period, up to 10^8 periods. As one may observe, the number 10^8 is a limitation everywhere, requiring a new implementation for each new extension.

This idea is also behind our systems of naming the numbers. The short scale (billion=10^9) uses a base 10^3, the thousand being the unit of the second order and so on. The long scale (billion=10^{12}) uses a base 10^6, the million being the unit of the second order and so on.

In our case, the limitation is the number of bits the computer may compute with. The number n in $(n, \mathbf{v})$ plays the role of Archimedes's order, so we are limited to the analog of his first period. With unrestricted allocatable memory, we could extend with a suitable implementation

[1] The programming language C is standardized by ISO/IEC 9899, latest version [ISO11b].
[2] See [AH1897] for an English translation and [AOG05] for a Spanish one.

to Archimedes's periods and higher levels. Knuth explores in [Knu81b] the binary representation of arbitrarily long numbers by codifying the level (order, period or higher) in a prefix.

Representing integer numbers reduces to representing natural numbers reserving a bit for the sign. Rational numbers reduces to numerator and denominator. Their arithmetic and zero-testing are exact. Real numbers are approximated with floating-point numbers, the numbers of finite binary expression. The usual implementation in hardware of floating point numbers is the standard IEEE 754, see [IEE08]. A bit field (S, E, M) with 1 bit for S, A bits for E and B bits for M represents the number $(-1)^S(1 + M2^{-B})2^{E-2^{A-1}+1}$ generically. There are exceptions for representing zero, infinity, NaN ("not a number") and too small numbers (subnormal numbers). The constants A and B depend on the precision. The standard IEEE 754 defines single-precision (`float` in C) with $A = 8$ and $B = 23$, and double-precision (`double` in C) with $A = 11$ and $B = 52$. The data types defined in the standard IEEE 754 are usually implemented by hardware, so their arithmetic is efficient. There are generalizations of these types implemented in software; see [vdH06a] and [vdH06b]. Example 46 shows how the arithmetic of fixed-precision floating-point numbers is not associative. So, for exact arithmetic, we need a data type with A and B variable in the same way we did for natural numbers.

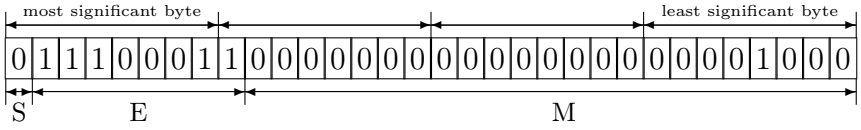


Figure 2.1: The number $2^{100} + 2^{80}$ represented in single precision.

Example 46. Figure 2.1 shows the number $a = 2^{100} + 2^{80}$ in single precision. The number 2^{60} is negligible compared to a, despite being greater than a 10^{18}, so that $a + 2^{60}$ yields a. Thus, the sum $2^{60} + a - 2^{100}$ yields 2^{80} if arranged $(2^{60} + a) - 2^{100}$, but the correct sum if arranged $2^{60} + (a - 2^{100})$.

The usual way of doing numerical computations is using fixed-precision floating-point arithmetic as if it were exact, which gives rise to several important problems for our purposes, as the non-commutativity and non-associativity of the floating-point arithmetic. A better way of doing numerical computations is carrying a bound of the error in the data type. For real numbers this style is called *interval arithmetic* and it is generalizable to the complex numbers, see [vdH06a] and [vdH06b]. This style computes the error a posteriori: given data $z_1, z_2, \ldots, z_n$ and error bounds $\delta_1, \delta_2, \ldots, \delta_n$, we compute ε such that

$$\left|f(z_1, z_2, \ldots, z_n) - f(w_1, w_2, \ldots, w_n)\right| < \varepsilon \tag{2.1}$$

if $|z_i - w_i| < \delta_i$ for each i. The definition of continuity suggests the opposite: given data $z_1, z_2, \ldots, z_n$ and an admissible error bound ε, we compute $\delta_1, \delta_2, \ldots, \delta_n$ such that (2.1) holds if $|z_i - w_i| < \delta_i$ for each i. This a-priori way requires computing backwards, which is not the usual way of doing computations, but makes sense in object-oriented programming.

2.1.2 Object oriented programming

I shall introduce some basic concepts of object-oriented programming, see [Arm06], with examples in C++.[3] *Class* is a generalization of data type. *Object* is a generalization of variable. An object is an instance of a class. The definition of a class T is a declaration of variables x, y, ..., z and functions f, g, ..., h. An object a of this class is a capsule (see *encapsulation*) containing variables a.x, a.y, ..., a.z of the declared data types, which may be other classes, and functions a.f, a.g, ..., a.h of the declared input and output. The definition of T declares the visibility (see *information hiding*) of each member variable or function, which may be *public* or *private*. A public member variable or function can be accessed anywhere. A private member variable or function of a can be accessed by a.f, a.g, ..., a.h and are hidden elsewhere. The functions T::f, T::g, ..., T::h may be defined globally to the class T or some of them may be declared *virtual* and kept undefined. In the latter case, T would be a *abstract class*. The purpose of abstract classes is being *base classes* for *inheritance*, as described below.

Example 47. Simulating mechanics, it might be useful to have a class RigidBody with public member functions for the center of mass, position, linear and angular momentum and the like. As the formulae are different for the different body shapes, we would need different definitions for these shapes, thus RigidBody would be an abstract class. Classes SolidSphere, HollowSphere, SolidCylinder and the like would be declared with RigidBody as base, inheriting its member variables and functions. The classes like SolidSphere would define the virtual functions of RigidBody. An instance of SolidSphere would an instance of RigidBody. One might create objects of type SolidSphere, but not of type RigidBody. All the instances of RigidBody would be instances of SolidSphere, HollowSphere, SolidCylinder or the like. The class RigidBody would allow us to handle at once all the formulas that are independent of the shape of the body.

Remark 48. C++ is so strict that does not allow objects of type `RigidBody`, neither a declaration `void f(RigidBody b)`. The handling of objects as `RigidBody`, forgetting the non-common features, is done with pointers. A pointer `p` of type

[3] The programming language C++ is standardized by ISO/IEC 14882, latest version [ISO11a].

`RigidBody*` may point to an instance of any derived class. The type of the object `*p` is known at run time, so it is not allowed in the code. Instead of `(*p).mass()`, C++ allows `p->mass()` for the handling.

Example 47 contains the idea of the device used in [vdH06a] for defining *effective real numbers*. The same way, we may define *effective complex numbers* as an abstract class Complex with a virtual member function approx with input a tolerance ε and output an ε-approximation to the value of the represented complex number. If a represents the number a, $|\mathsf{a.approx}(\varepsilon) - a| < \varepsilon$ for any $\varepsilon > 0$. If mpfpc is a class for multiple-precision floating-point complex numbers that could represent $(\mathbb{Z} + \mathbb{Z}\mathrm{i})2^{\mathbb{Z}}$ if we had enough allocatable memory, we could declare the base class Complex_base the following way:

```
class Complex_base {
  public:
  virtual mpfpc approx(mpfpc)=0;
};
```

We may define an immediate derived class of Complex_base as a wrapping of mpfpc for a complex number that is known exactly.

```
class Complex_mpfpc:
  public Complex_base
{
  private:
  mpfpc a;
  public:
  mpfpc approx(mpfpc epsilon)
  { return a; }
};
```

For the user we define a wrapper Complex of a pointer to Complex_base.

```
class Complex {
  private:
  Complex_base* pointer;
  public:
  Complex(Complex z):
```

```
    pointer(z.pointer) {}
  mpfpc approx(mpfpc epsilon)
  { return pointer->approx(epsilon); }
};
```

For the sum we define a derived class Complex_sum in the following way.

```
class Complex_sum:
  public Complex_base
{
  private:
  Complex_base *a, *b;
  public:
  Complex_sum(Complex_base* p, Complex_base* q):
    a(p), b(q) {}
  mpfpc approx(mpfpc epsilon)
  { return a->approx(epsilon>>1) + b->approx(epsilon>>1); }
};
```

Notice that we use the bit shift `epsilon>>1` instead of `epsilon/2` because the former is exact. As we will see soon, the division in mpfpc yields a result outside mpfpc.

2.1.3 Operations

Using the class Complex_sum, we may define the operation + in the following way.

```
Complex Complex::operator+(Complex z) {
  Complex w;
  w.pointer=new Complex_sum(pointer,z.pointer);
  return w;
}
```

Notice that this device generates a lot of garbage, so we need to provide a garbage collector. Such a garbage collector may be using C++ smart pointers instead of raw pointers.

In a similar way we may define classes and operations for the substraction, multiplication and the like. The formulas like

$$(\mathsf{z}+\mathsf{w}).\mathsf{approx}(\varepsilon) = \mathsf{z}.\mathsf{approx}(\varepsilon/2) + \mathsf{w}.\mathsf{approx}(\varepsilon/2)$$

are those used for proving the continuity of the operation, like

$$(\mathsf{z}*\mathsf{w}).\mathsf{approx}(\varepsilon) = \mathsf{z}.\mathsf{approx}(\delta) * \mathsf{w}.\mathsf{approx}(\delta)$$
$$\text{for } \delta = \min\left\{1, \frac{\varepsilon}{\left|\mathsf{z}.\mathsf{approx}(1)\right| + \left|\mathsf{w}.\mathsf{approx}(1)\right| + 3}\right\}.$$

If a represents the number a, for $\delta > 0$, it may happen $|\mathsf{a}.\mathsf{approx}(\delta)| \leqslant \delta$ or $|\mathsf{a}.\mathsf{approx}(\delta)| > \delta$. The function given by $f(z) = 1/z$ transforms the disk $\mathrm{B}(\mathsf{a}.\mathsf{approx}(\delta), \delta)$ into an unbounded region in the former case (see Figure 2.3) and into a disk in the latter (see Figure 2.2).[4] If we take $\delta < |a|/2$, then $|\mathsf{a}.\mathsf{approx}(\delta)| > \delta$, but we do not know a lower bound for $|a|$. Moreover, we can only certify that $a = 0$ after infinitely many tests $|\mathsf{a}.\mathsf{approx}(\delta_n)| \leqslant \delta_n$ with $\lim_{n\to\infty} \delta_n = 0$, never after finitely many. Therefore, the division in Complex cannot be completely implemented. It may happen an error if the divisor is too close to zero, similar to the usual "division by zero" error.

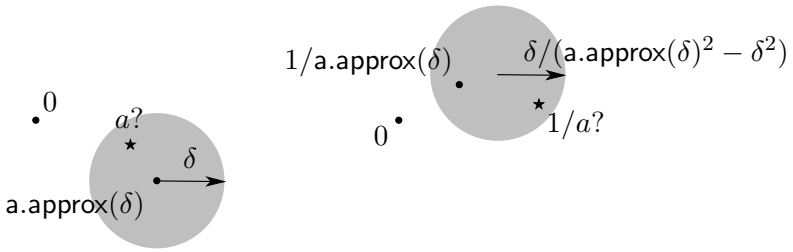


Figure 2.2: Case $|\mathsf{a}.\mathsf{approx}(\delta)| > \delta$.

Let tol be a global variable of type mpfpc; we will consider that a is *too close to zero* if $|\mathsf{a}.\mathsf{approx}(\mathsf{tol})| \leqslant \mathsf{tol}$. The division will be implemented only for divisor not too close to zero, happening the error "divisor too close to zero" otherwise. In this case, if the division in mpfpc were exact,

$$(1/\mathsf{a}).\mathsf{approx}(\varepsilon) = \frac{\overline{\mathsf{a}.\mathsf{approx}(\mathsf{tol})}}{|\mathsf{a}.\mathsf{approx}(\mathsf{tol})|^2 - \mathsf{tol}^2} \quad \text{for } \varepsilon \geqslant \frac{\mathsf{tol}}{|\mathsf{a}.\mathsf{approx}(\mathsf{tol})|^2 - \mathsf{tol}^2}$$

would be enough. Let us define a class for the division in mpfpc:

[4] $1/\mathrm{B}(a,\delta) = \mathrm{B}\left(\dfrac{\overline{a}}{|a|^2-\delta^2}, \dfrac{\delta}{|a|^2-\delta^2}\right)$

```
class Complex_div_mpfpc:
  public Complex_base
{
  private:
  mpfpc a, b;
  public:
  Complex_div_mpfpc(mpfpc x, mpfpc y):
    a(x), b(y) {}
  mpfpc approx(mpfpc);
};
```

Then we may define

$$(1/\mathsf{a}).\mathsf{approx}(\varepsilon) = \left(\overline{\mathsf{a}.\mathsf{approx}(\mathsf{tol})}/\left(|\mathsf{a}.\mathsf{approx}(\mathsf{tol})|^2 - \mathsf{tol}^2\right)\right).(\varepsilon/2)$$
$$\text{for } 2 * \mathsf{tol} \leqslant \varepsilon * \left(|\mathsf{a}.\mathsf{approx}(\mathsf{tol})|^2 - \mathsf{tol}^2\right). \tag{2.2}$$

If (2.2) does not hold, we define

$$\eta = \frac{|\mathsf{a}.\mathsf{approx}(\mathsf{tol})| - \mathsf{tol}}{2}.$$

If $\delta < \eta$, the distance between the disk $U = \mathrm{B}(\mathsf{a}.\mathsf{approx}(\delta), \delta)$ and the origin is greater than η. The function given by $f(z) = 1/z$ transforms U into another disk. For any $z, w \in U$ we have

$$\left|f(z) - f(w)\right| = \left|\frac{1}{z} - \frac{1}{w}\right| = \frac{|z-w|}{|z||w|} < \frac{2\delta}{\eta^2},$$

thus the diameter of $f(U)$ is $2\delta/\eta^2$ at most. Hence, if $\delta < \varepsilon\eta^2/2$, the radius of $f(U)$ is smaller than $\varepsilon/2$. Therefore, we may define

$$(1/\mathsf{a}).\mathsf{approx}(\varepsilon) = \left(\overline{\mathsf{a}.\mathsf{approx}(\delta)}/\left(|\mathsf{a}.\mathsf{approx}(\delta)|^2 - \delta^2\right)\right).(\varepsilon/2)$$
$$\text{for } \delta < \min\{\eta, \varepsilon\eta^2/2\}.$$

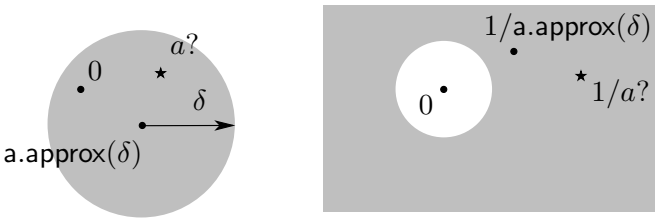


Figure 2.3: Case $|\mathsf{a}.\mathsf{approx}(\delta)| \leqslant \delta$.

This completes the elementary operations in Complex and proves that the effective complex numbers form a field.

2.1.4 Roots of polynomials

The survey [Pan97] describes different methods for approximating the roots of a polynomial at desired accuracy. Such methods start with a slow but warranted method like *quadrisection* or *splitting circle* for isolating the roots and then apply a fast iterative algorithm like *Newton-Raphson* or *Weierstrass-Durand-Kerner* for refining the roots. These iterative refining methods are used once the conditions for their convergence hold. Newton-Raphson refines each root independently, while Weierstrass-Durand-Kerner approximates all the roots simultaneously. For a survey on refining methods, see [MP12]. The complexity of such a hybrid algorithm is the complexity of the last stage (refining) method. This is like the classification of Iberian pork according to the dark pig's diet: what matters is its last-stage (fattening) diet and not the diet of the previous stages.

The *quadrisection method* was proposed by H. Weyl in [Wey24, part II], a constructive proof of the Fundamental Theorem of Algebra. He began with a square containing all the roots of the polynomial P and successively divided it into four equal subsquares, discarding the subsquares without roots. In order to count the roots in a square S, he used $\int_{\partial S} P'/P = 2\pi\mathrm{i}\#(P^{-1}(0) \cap S)$. In [Pan00] V. Y. Pan combines Weyl's algorithm with a result of P. Turán (Theorem 49) and a modified Newton-Raphson iteration. A variant of this method will be explained below.

The *splitting circle method* was introduced by A. Schönhage in the seminal preprint [Sch82]. This method looks for a circumference without roots separating roots inside the circle and roots outside, in a way as close to half and half as possible. Then we reconstruct the factor corresponding to the inner roots by means of the values $\int_{\partial C} P' x^k/P = 2\pi\mathrm{i} \sum_{z \in P^{-1}(0) \cap C} z^k$, where P is the polynomial to factor and C the splitting circle. In order to compute numerically these integrals we need a wide annulus around the splitting circumference without roots. If the annulus is narrow, we can widen it by means of the *Graeffe transformation* $P(x) \mapsto P(\sqrt{x})P(-\sqrt{x})$, which squares the roots.

Using the methods above, one may approximate the roots of a polynomial at desired accuracy, so the effective complex numbers form an algebraically closed field. The class declared below implements the root in $C = \mathrm{B}(\mathsf{center}, \mathsf{radius})$ of a polynomial with coefficients in Complex, provided that there is only one root inside

the disk. The approximations can be done by Newton-Raphson method provided radius is small enough so that the bounds of [Sma86] hold. Another method that is effective for any radius but maybe less efficient is the integral $1/(2\pi\mathrm{i})\int_{\partial C} P'x/P$, which can de computed by the techniques of §2.2.2.

```
class Complex_root_poly:
  public Complex_base
{
  private:
  size_t degree;
  Complex_base* coeff[];
  mpfpc center, radius;
  public:
  mpfpc approx(mpfpc);
};
```

The problem with this is that distinguishing between a multiple root and a cluster of roots requires exact computations. This can be done for isolating the singularities of the differential equation, but not for the eigenvalues of a monodromy matrix.

Let us consider the quadrisection method, which is based on the following result.

Theorem 49 (Turán). *For a given polynomial $P \in \mathbb{C}[x]$ and $m \in \mathbb{Z}$ we may compute δ such that $0 \leqslant \delta \leqslant \min\{|\lambda| : P(\lambda) = 0\} \leqslant 5^{1/2^m}\delta$.* [Tur75, §6]

The bound δ is computed by effective operations with effective complex numbers, which involve m iterated Graeffe transformations, solving a triangular linear system given by Newton's identities and a maximum of roots of moduli of these solutions modified, so δ is an effective complex number. Turán proposes this bounds for his algorithm [Tur75, §8] of approximating a root of a given polynomial, at desired accuracy, but only one root at a time. Turán's article [Tur75] provides versions of Theorem 49 for the eigenvalues directly from the matrix, but he does not provide proofs, which can be found in other of his works. Other authors, like [Buc67], provide similar bounds and prove them.

This algorithm proceeds by quadrisection of a candidate square, computing Turán's bound for the distance of the roots from the center of each subsquare. If Turán's bound is greater than the distance to the vertices, considering the error

margin, we know for sure that there is no root in this square, and hence we discard it. Otherwise, there may be a root in the candidate square or there may not, but in the latter case there must be a root in any of its 8 surrounding squares. So, contrary to bisection method in the real line, a candidate square may contain no root. Hence our approximations will be connected components of candidate squares.

The initial square containing all the roots of $P = a_0 + a_1 x + \cdots + a_n x^n$ can be computed, after centering on the barycenter for convenience, by applying Theorem 49 to $a_n + a_{n-1}x + \cdots + a_0 x^n$, whose roots are the inverses of the roots of of P. We may not know the number of roots in a given square, but we may count the number of roots in a connected component of candidate squares. This reduces to the computation of an integral around the component, which may be done using the techniques of §2.2.2. With these integrals one may may split P and proceed with each factor. Thus we may implement a quadrisection solver

```
class Polynomial_solver {
  private:
  size_t degree;
  Complex_base* coeff[];
  public:
  mpfpc approx(size_t,mpfpc);
};
```

where $\mathsf{approx}(i, \varepsilon)$ gives the ε-approximation of the i-th root. The solver assigns to each component as many indices as roots it contains. When the quadrisection splits a component, its indices are distributed among the new components contained in it. This allows the following implementation of the roots of a polynomial with effective coefficients.

```
class Complex_root_poly:
  public Complex_base
{
  private:
  size_t index;
  Polynomial_solver* solver;
  public:
  mpfpc approx(mpfpc epsilon)
  { return solver->approx(index,epsilon); }
};
```

2.2 Generators of the differential Galois group

In order to compute the Ramis generators of the differential Galois group (§2.2.4), I shall review how to compute a fundamental system of solutions at a singular (§2.2.3) and at a non-singular point (§2.2.1), and how to analytically continue the latter (§2.2.2). In principle we should compute all the Ramis generators, but in Chapter 3 we will take the *eurymeric* closure of the Galois group, and the identity component of a eurymeric group is *broad*, which means that it is the non-singular part of the Lie algebra. As the algebraic tori are connected, it is safe to take their *broad hull*, since we will take it anyway in the algorithm for the eurymeric closure. As the Stokes and logarithm automorphisms are unipotent, they belong to the identity component and has 1 as only eigenvalue, so it is safe to reduce the space of solutions to those invariant by these automorphisms, since we keep the Singerian solutions. Once in a Stokes-free and logarithm-free subspace, we may compute the analytic monodromy and the torus of the monodromy instead of the formal monodromy. See Chapter 3 for the details.

2.2.1 Solutions at a non-singular point

Let us consider an explicitable system of higher order differential equations

$$\mathtt{A}_r(x)\mathbf{y}^{(r)} + \mathtt{A}_{r-1}(x)\mathbf{y}^{(r-1)} + \cdots + \mathtt{A}_0(x)\mathbf{y} = \mathbf{0}, \tag{2.3}$$

where $\mathtt{A}_0(x), \mathtt{A}_1(x), \ldots, \mathtt{A}_r(x) \in \mathbb{C}[x]^{n\times n}$. Let z_0 be a non-singular point of (2.3), which means $\det \mathtt{A}_r(z_0) \neq 0$. The power-series solutions of (2.3) are of the form $\mathbf{y} = \sum_{k=0}^{\infty} \mathbf{y}_k (x-z_0)^k$, with $\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{r-1} \in \mathbb{C}^n$ freely chosen and the remaining $\mathbf{y}_k$ determined by recurrence. This yields nr linearly independent solutions, a fundamental system. These power series are convergent. We shall compute the recurrence and a majorizing geometric series.

Substituting $\mathbf{y} = \sum_{k=-\infty}^{\infty} \mathbf{y}_k (x-z_0)^k$ into (2.3), we get

$$\sum_{i=0}^{r}\sum_{j=0}^{m}\sum_{k=0}^{\infty} \frac{1}{j!}\left(\prod_{l=1}^{i}(k-j+l)\right) \mathtt{A}_i^{(j)}(z_0)\mathbf{y}_{k-j+i}(x-z_0)^k = \mathbf{0},$$

with m the maximum of the degrees of the entries of the $\mathtt{A}_i$. Hence

$$\sum_{i=0}^{r}\sum_{j=0}^{m} \frac{1}{j!}\left(\prod_{l=1}^{i}(k-j+l)\right) \mathtt{A}_i^{(j)}(z_0)\mathbf{y}_{k-j+i} = \mathbf{0} \tag{2.4}$$

for every k, which is a recurrence equation for $\mathbf{y}_k$. From (2.4) we may isolate

$$\frac{(k+r)!}{k!}\,\mathtt{A}_r(z_0)\mathbf{y}_{k+r} = \dots$$

and, as $\mathtt{A}_r(z_0)$ is invertible, we get an explicit recurrence $\mathbf{y}_{k+r} = \dots$ for $k \geqslant 0$. This recurrence yields the companion recurrence $\mathbf{u}_{k+1} = \mathtt{C}(k)\,\mathbf{u}_k$, with

$$\mathbf{u}_k = \begin{pmatrix} \mathbf{y}_k \\ \mathbf{y}_{k-1} \\ \vdots \\ \mathbf{y}_{k-m-r+1} \end{pmatrix}, \quad \mathtt{C}(k) = \begin{pmatrix} \mathtt{B}_1(k) & \mathtt{B}_2(k) & \dots & \mathtt{B}_{m+r-1}(k) \\ \mathtt{I} & & & \\ & \ddots & & \\ & & \mathtt{I} & \mathtt{0} \end{pmatrix}$$

$$\text{and} \quad \lim_{k\to\infty} \mathtt{B}_j(k) = \frac{-1}{j!}\,\mathtt{A}_r(z_0)^{-1}\mathtt{A}_r^{(j)}(z_0).$$

As $\mathtt{C}(k)$ has a finite limit, taking a bound $\|\mathtt{C}(k)\|_\infty \leqslant \mu$, we have $\|\mathbf{u}_k\|_\infty \leqslant \mu^{k-r+1}\|\mathbf{u}_{r-1}\|_\infty$, thus $\mathbf{u}_k$ (and hence $\mathbf{y}_k$) grows in ∞-norm as a geometric progression at most.

The previous paragraph is a constructive proof that $\sum_{k=0}^\infty \mathbf{y}_k(x-z_0)^k$ is convergent. This method is used in [vdH99] for scalar equations, and allows to explicitly compute for a given $\mu > 1/\rho$, with ρ the distance to the closest singularity, a $\lambda > 0$ such that the geometric progression $(\lambda\mu^k)_{k=0}^\infty$ majorizes $\big(\|\mathbf{y}_k\|_\infty\big)_{k=0}^\infty$. This allows to bound the rest of the series,

$$\left\| \sum_{k=N+1}^{\infty} \mathbf{y}_k(z-z_0)^k \right\|_\infty \leqslant \frac{\lambda}{1-\mu|z-z_0|}\big(\mu|z-z_0|\big)^{N+1}. \tag{2.5}$$

So, given $\varepsilon > 0$ and $z \in \mathrm{B}(z_0, 1/\mu)$, computing N large enough that the right hand side of (2.5) is less than ε, then the sum $\mathbf{f}$ of $\sum_{k=0}^\infty \mathbf{y}_k x^k$ satisfies

$$\left\| \mathbf{f}(z) - \sum_{k=0}^{N} \mathbf{y}_k(z-z_0)^k \right\|_\infty < \varepsilon.$$

A fast way of computing this partial sum is also given in [vdH99].

In general,[5] the rest $\mathbf{R}_N = \sum_{k=N+1}^\infty \mathbf{y}_k(x-z_0)^k$ satisfies

$$\left\|\mathbf{R}_N^{(i)}\right\|_\infty \leqslant h_N^{(i)}\big(|z-z_0|\big) \quad \text{for } h_N(x) = \frac{\lambda}{1-\mu x}(\mu x)^{N+1}.$$

[5] J. van der Hoeven does not need this generalization for scalar equations, but it is necessary when $\mathtt{A}_0$ is not invertible.

So, given $\varepsilon > 0$ and $z \in \mathrm{B}(z_0, 1/\mu)$, computing N large enough that $h_N^{(i)}\big(|z|\big) < \varepsilon$, we have

$$\left\| \mathbf{f}^{(i)}(z) - \sum_{k=0}^{N} k(k-1)\cdots(k-i+1)\mathbf{y}_k(z-z_0)^{k-i} \right\|_\infty < \varepsilon.$$

2.2.2 Effective analytic continuation

Let us define a class System for implementing a system like (2.3):

```
class System {
  public:
  size_t size, order;
  Matrix<Complex> coeff[];
};
```

A solution of (2.3) at a non-singular point is determined by the point and the initial values. It will be only necessary to consider solutions at points representable by floating-point numbers.

```
class Solution {
  public:
  System system;
  mpfpc point;
  Vector<Complex> ini[];
  Solution cont(mpfpc);
  Solution cont(Path);
};
```

If $\mathbf{f}$ is the solution, then $\mathsf{ini}[i] = \mathbf{f}^{(i)}(\mathsf{point})$. The function Solution::cont(mpfpc) performs the analytic continuation within the disc of convergence, using the method described in §2.2.1 or subdividing when necessary. The class Path represents a broken-line path avoiding the singularities. The function Solution::cont(Path) subdivides the input path and invokes Solution::cont(mpfpc). Starting with a fundamental system of solutions, we may compute the generators (of type Matrix⟨Complex⟩) of the monodromy group $\mathrm{Mon}(\mathsf{S}, \mathsf{z})$. For a detailed description on the implementation issues, see [vdH05].

2.2.3 Formal solutions at a singularity

As explained in §1.2.2, a scalar differential operator

$$L = a_r(x)\,\partial_x^r + a_{r-1}(x)\,\partial_x^{r-1} + \cdots + a_0(x),$$

with $a_0, a_1, \ldots, a_r$ polynomials, has a complete system of solutions of the form

$$\exp\left(q(x^{-1/p})\right) x^\alpha \sum_{k=0}^{\infty} \sum_{i=0}^{r} y_{ki} x^{k/p} \log^i x$$

with p the ramification index and q a polynomial with $q(0) = 0$. The set of the $q(x^{-1/p})$, which is an invariant of the equation, can be computed symbolically as described in [DDT82]. For each q, applying the substitution $\partial_x \mapsto \partial_x + \partial_x \left[q(x^{-1/p})\right]$ to L we get a differential operator L_q such that $\exp\left(q(x^{-1/p})\right) y$ is a solution of L if and only if y is a solution of L_q.

The problem is thus reduced to computing the solutions of the form

$$x^\alpha \sum_{k=0}^{\infty} \sum_{i=0}^{r} y_{ki} x^{k/p} \log^i x$$

of $$L_q = b_r(x^{1/p})\,\partial_x^r + b_{r-1}(x^{1/p})\,\partial_x^{r-1} + \cdots + b_0(x^{1/p}),$$

which is done with the classic Frobenius method described in [DT81]. In particular, this method computes symbolically the α. All the process is explained in [Tou87], refined in [Bar89, ch. 2] and [vH96, ch. 2].

Substituting $x = t^p$ and $\partial_x = (t^{1-p}/p)\,\partial_t$ in L_q, we get an operator whose coefficients are polynomials in t and t^{-1}. Multiplying by the minimal suitable power of t we get an operator L_{q0} whose coefficients are polynomials in t. For each α, applying the substitution $\partial_t \mapsto \partial_t + \alpha p\, t^{-1}$ to L_{q0} we get a differential operator $L_{q\alpha}$ such that $x^\alpha y$ is a solution of L_q if and only if y is a solution of $L_{q\alpha}$. Notice that the computation of $L_{q\alpha}$ is also valid for matrix coefficients.

For differential systems one can apply Cyclic Vector Lemma, explained in §1.1.4, but this may yield an equation with too large coefficients, as shown in [Hil87, §2.5], hence methods working with the differential systems are preferred. Let us consider a differential system $x^g \mathbf{y}' = \mathtt{A}(x)\,\mathbf{y}$ with $\mathtt{A}$ an $n \times n$ matrix of convergent power series, which may be the companion system of the explicit form of a system of higher order differential equations. Turrittin's proof [Tur55] gives

a procedure to compute the ramification index p, the $n \times n$ matrices $\mathtt{L}$ and $\mathtt{Q}$ and any term $\mathtt{H}_k$ of the fundamental matrix solution

$$\left(\sum_{k=0}^{\infty} \mathtt{H}_k x^{k/p}\right) x^{\mathtt{L}} \exp\left(\mathtt{Q}(x^{-1/p})\right)$$

of Hukuhara-Turrittin theorem. This procedure has been refined as exposed in [Was76] or [Bal00, ch. 3]. A more efficient alternative is exposed in [Bar97] using [Mos60].

In both cases, scalar and systems, we can compute the pairs (q, α) such that $x^\alpha \exp\left(q(x^{-1/p})\right)$ appears in the formal solutions.

2.2.4 The local Galois group at a singularity

Let L be a differential operator over $\mathbb{C}[x]^{n\times n}$ with a singularity at the origin, with ramification index p. Our objective is to bring the local Galois group to a non-singular point. As we are only interested in Singerian solutions, according to Theorem 44, we may restrict the group to the invariant subspace of solutions of the form

$$\exp\left(q(x^{-1/p})\right) x^\alpha \left(f_1(x^{1/p}), f_2(x^{1/p}), \ldots, f_n(x^{1/p})\right)^\intercal$$

with $f_1, f_2, \ldots, f_n$ convergent series and q a polynomial with $q(0) = 0$. These solutions are fixed by the Stokes automorphisms and by the group $G_{\log}$ described in §1.4.1.

Let μ be the monodromy around the origin based at a near non-singular point $z_0 > 0$. The space $V = \ker(\mu^p - \mathrm{id})$ contains the solutions we look for. As described in §2.2.3, we may compute the set of all the admissible pairs (q, α). Let $f_{q\alpha}$ be the function corresponding to $\exp\left(q(x^{-1/p})\right) x^\alpha$ in the principal determination of the logarithm for each admissible pair (q, α). As explained in §2.2.3, $f_{q\alpha}^{-1}V$ is a subspace of solutions of $L_{q\alpha}$. A solution $g \in f_{q\alpha}^{-1}V$ defines a p-ramified function at the origin precisely if

$$\int_{p(z_0 \circlearrowleft 0)} g(z) z^{k/p} \mathrm{d}z = 0$$

for any integer $k \geqslant 1-p$, where $p(z_0 \circlearrowleft 0)$ means p times the loop $z_0 \rightsquigarrow z_0$ around the origin counterclockwise. The proof requires considering the Laurent expansion of g unramified. These integrals can be computed by analytic continuation as described in §2.2.2, and it is not necessary to loop p times, but raise the monodromy matrix to the p-th power. This way, after computing infinitely many integrals, we get the part $W_{q\alpha}$ of $f_{q\alpha}^{-1}V$ that defines p-ramified functions. Thus the solutions we look

for are spanned by $f_{q\alpha}W_{q\alpha}$. In these bases the expression of the exponential and potential tori is simple, but it requires linear algebra over $\mathbb{Q}$ in order to compute a basis of exponential parts. The so-called *broad hull* of an algebraic torus is another torus but simpler to compute, as described in Example 59. In such a computation we only need to recognize exponential parts that are equal, something feasible since the exponential parts are computed symbolically.

The only drawback of the procedure described in the previous paragraph is that we need to compute infinitely many integrals, something unavoidable except if certain coefficients have a particular form that leads to a recurrence for the involved Laurent series, as described below. What we can do is to compute an approximated $W_{q\alpha}$, checking only $|\mathsf{integral}(k).\mathsf{approx}(\mathsf{tol})| \leqslant \mathsf{tol}$ for $0 \leqslant k \leqslant \mathsf{K}$ for certain K global as tol. This requires only a finite computation.

One of these exceptions that leads to a recurrence for the aforesaid Laurent series is when the infinity is a non-singular point of the equation, which is a case general enough[6] to be considered in detail. Let us assume that g is unramified and a solution of the equation

$$\mathtt{A}_0(x)\mathbf{y} + \mathtt{A}_1(x)\delta_x\mathbf{y} + \cdots + \mathtt{A}_r(x)\delta_x^r\mathbf{y} = \mathbf{0}, \tag{2.6}$$

with $\mathtt{A}_0(x), \mathtt{A}_1(x), \ldots, \mathtt{A}_r(x) \in \mathbb{C}[x]^{n\times n}$ and $\delta_x = x\partial_x$ the Euler derivation. Let m be the maximum of the degrees of the entries of the $\mathtt{A}_i(x)$, so we have

$$\mathtt{A}_i(x) = \mathtt{A}_{i0} + \mathtt{A}_{i1}x + \cdots + \mathtt{A}_{im}x^m$$

with $\mathtt{A}_{ij} \in \mathbb{C}$. At infinity, with $t = x^{-1}$, the equation (2.6) is transformed into

$$t^m\mathtt{A}_0(t^{-1})\mathbf{y} - t^m\mathtt{A}_1(t^{-1})\delta_t\mathbf{y} + \cdots + (-1)^r t^m\mathtt{A}_r(t^{-1})\delta_t^r\mathbf{y} = \mathbf{0},$$

where the coefficient of $\delta_t^r\mathbf{y}$ takes the value $(-1)^r\mathtt{A}_{rm}$ at $t = 0$. The equation (2.6) being non-singular at infinity is equivalent to $\det\mathtt{A}_{rm} \neq 0$.

A solution of (2.6) of the form $\sum_{k=-\infty}^{\infty}\mathbf{y}_k x^k$ yields the recurrence

$$\left(\sum_{i=0}^{r}\mathtt{A}_{im}k^i\right)\mathbf{y}_k = \cdots \tag{2.7}$$

with the r.h.s. involving $\mathbf{y}_l$ for $k < l \leqslant k+m$. Considering $\mathtt{B}(x) = \sum_{i=0}^{r}\mathtt{A}_{im}x^i$, the recurrence (2.7) can be written as $\mathtt{B}(k)\mathbf{y}_k = \cdots$. As $\lim_{k\to-\infty}k^{-r}\mathtt{B}(k) = \mathtt{A}_{rm}$, if $\det\mathtt{A}_{rm} \neq 0$, there is k_0 such that $\det(k^{-r}\mathtt{B}(k)) \neq 0$ for $k \leqslant k_0$, so $\det\mathtt{B}(k) \neq 0$

[6]Not only is it a generic case, but also any explicitable equation can be transformed into one of this case by means of a Möbius transformation.

and thus the recurrence (2.7) works. Moreover, $\det \mathtt{B}(x) \not\equiv 0$, thus we can compute such a k_0 by bounding the negative integer roots of $\det \mathtt{B}(x)$. Hence, in order to prove that $g = \sum_{k=-\infty}^{\infty} \mathbf{y}_k x^k$ is holomorphic at the origin it suffices to prove $\mathbf{y}_{-1} = \mathbf{y}_{-2} = \cdots = \mathbf{y}_{-m} = \mathbf{0}$ and that $\mathbf{y}_k = \mathbf{0}$ for $k < 0$ with $\det \mathtt{B}(k) = 0$, finitely many computations.

2.3 Rank over the complex field

The rank over $\mathbb{C}$ of an $m \times n$ matrix is generically $r = \min\{m, n\}$. Let $V_i \subset \mathbb{C}^{m \times n}$ be the vanishing set of the minors of order i, for $1 \leqslant i \leqslant r$. This defines a chain of algebraic varieties $V_1 \subset V_2 \subset \cdots \subset V_{r+1}$, with $V_{r+1} = \mathbb{C}^{m \times n}$, such that $V_{i+1} \setminus V_i$ is the set where the rank is i, with $V_0 = \varnothing$. In a neighborhood of a given matrix of rank s (in the usual topology) there will always be matrices of rank t for any $t \geqslant s$. Matrices of rank $t < s$ are avoided by taking the neighborhood small enough, but matrices of rank $t > s$ are unavoidable.

A way for computing the rank is evaluating the minors of order r, then the minors of order $r-1$ and so on, stopping when one of these minors does not vanish; the rank is the order of such a minor. Working with effective complex numbers we need to choose $\varepsilon > 0$ and use the condition $|\mathsf{minor.approx}(\varepsilon)| \leqslant \varepsilon$ instead of $\mathsf{minor} = 0$. This approximated rank may be less than the exact one, but not greater. For ε small enough, the rank is computed exactly, but we do not know beforehand how small it is necessary.

Using minors is illustrative but far from the best way of computing the rank. Gaussian reduction is more efficient, and the reduced matrix may be useful for another purpose. An $m \times n$ matrix $\mathtt{M}$ represents a linear mapping $\mathbb{C}^n \to \mathbb{C}^m$, which determines two vector subspaces: the kernel $\ker \mathtt{M} \subset \mathbb{C}^n$ and the image $\operatorname{img} \mathtt{M} \subset \mathbb{C}^m$. They correspond to the implicit and the parametric equations respectively. The reduction by rows keeps the kernel invariant, while the reduction by columns keeps the image invariant, thus we will use the suitable style of Gaussian reduction.

Carrying out Gaussian reduction of a matrix, in each step, we have to choose a pivot among a few candidates. In the following example of reduction by rows

$$\begin{pmatrix} 1 & * & * & * \\ 0 & \circledast & * & * \\ 0 & \circledast & * & * \\ 0 & \circledast & * & * \end{pmatrix}$$

the candidates are circled. It suffices to choose any nonzero candidate, but working with effective complex numbers this is undecidable. As the pivot is used as a denominator, we need to choose a pivot not too close to zero. If all the candidates are too close to zero, we cannot decide if there is a nonzero one, so the algorithm fails.

Let us consider the following variation of Gaussian elimination.

```
template<class T>
size_t Matrix<T>::reducebyrows
  (size_t pivotrow[], size_t pivotcol[])
{
  size_t npivots=0;
  while(newpivot(pivotrow,pivotcol,npivots)) {
    for(size_t i=1; i<=nrows; i++) {
      if(i!=pivotrow[npivots])
        reducerow(i,pivotrow[npivots],pivotcol[npivots]);
    }
    npivots++;
  }
  return npivots;
}
```

This function takes the arrays pivotrow and pivotcol, which are assumed to have length nrows and ncols respectively, and returns the rank of the matrix, which is the number of pivots. After running the function, the ith pivot is in the row pivotrow$[i-1]$ and the column pivotcol$[i-1]$. Notice that, in the programming languages C and C++, the elements of an array a of length n are a$[0]$, a$[1]$, ..., a$[n-1]$; contrary, the template class Matrix is assumed to number the rows from 1 through nrows and the columns from 1 through ncols. The auxiliary function Matrix$\langle$T$\rangle$::newpivot takes the arrays and the current number of pivots, and returns `true` if a new pivot was found and `false` if not. In the positive case, the new pivot is in the row pivotrow[npivots] and the column pivotcol[npivots]. I shall show a single detail of the implementation of Matrix$\langle$T$\rangle$::newpivot.

```
template<class T>
bool Matrix<T>::newpivot
  (size_t pivotrow[], size_t pivotcol[], size_t npivots)
{
  //...
```

```
  if(entry(i,j).iszero())
  //...
}
```

If T is a class with an exact zero-test T::iszero, then Matrix⟨T⟩::reducebyrows is correct. In the case of effective complex numbers we have the following approximated zero-test.

```
bool Complex::iszero()
{ return abs(approx(tol))<=tol; }
```

This is enough for avoiding the error "divisor too close to zero" in Matrix⟨Complex⟩::reducerow. If tol is small enough, Matrix⟨Complex⟩::reducebyrows is correct, but we do not know beforehand how small tol is necessary. If tol is so large that Matrix⟨Complex⟩::reducebyrows is incorrect, the error is an underestimation of the rank, never and overestimation. This means that the dimension of the kernel, dropping the deemed-zero rows, may be overestimated, but never underestimated. In any case, the approximated kernel contains the exact one.

We have considered the Gaussian reduction by rows, but we may consider the reduction by columns mutatis mutandis. We reduce a matrix by columns when we are interested in its image space. Contrary to the case of the kernel, the dimension of the image, dropping the deemed-zero columns, may be underestimated, but never overestimated. In any case, the approximated image space is contained in the exact one.

2.4 Rank over the rational field

In the case of the rank over $\mathbb{C}$, in any neighborhood of a matrix of rank r there are matrices of all the possible ranks greater than r, but the matrices of rank less than r are avoided if the neighborhood is small enough. The case of the rank over $\mathbb{Q}$ of complex numbers is different. Let us consider the simple case of $\{1, a\}$, with $a \in \mathbb{R}$, whose rank is 1 if $a \in \mathbb{Q}$ and 2 if not. In this case any neighborhood of a contains both cases of rank 1 and 2. We lack the key feature in §2.3.

For the rank over $\mathbb{C}$ we have the exact zero-test $|\mathsf{a.approx}(\delta_k)| \leqslant \delta_k$ for all k with $\lim_{k\to\infty} \delta_k = 0$, which requires an infinity of steps, and the truncation $|\mathsf{a.approx(tol)}| \leqslant \mathsf{tol}$, which is finite but allows false negatives. For the rank over $\mathbb{Q}$

we may devise a test of rationality based on the continuous fraction expansion.[7] The continuous fraction expansion of an effective complex number is computed using Euclid's algorithm,[8] which is "the granddaddy of all algorithms," according to Knuth [Knu81a, p. 318, l. 5–7], "because it is the oldest nontrivial algorithm that has survived to the present day." The algorithm is basically successive subtracting the integral part and inverting, terminating when the number to invert is zero. For effective complex numbers it terminates when the number to take integral part is too close to $m \in \mathbb{Z}$. In this case we subtract m and then the number to invert is too close to zero. If we find that any of the intermediate results is not real, its disk of uncertainty does not intersect the real line, we return "irrational" as answer.

According to Euclid,[9] a number is rational if and only if the algorithm terminates. We have another exact test needing infinitely many steps for a positive answer, so we can conceive a truncation thereof to finitely many steps. Such a truncation is deeming a number irrational if the algorithm reaches a convergent p_k/q_k, a fraction in its lowest terms according to [HW75, thm. 157], with $q_k > \mathsf{Q}$ for certain bound Q global as tol. This approximate test of rationality may give both false positives and false negatives. An irrational number too close to zero is deemed zero, and thus rational, by the approximate test. An irreducible fraction with a large denominator, greater than Q, is deemed irrational by the approximate test. Such an approximate test requires $\mathcal{O}(\log \mathsf{Q})$ steps.

Let us analyze the possible errors in the approximate test of rationality described in the previous paragraph. A false negative happens if we test an irreducible fraction with a large denominator, greater than Q. A false positive happens if we test an irrational number such that Euclid's algorithm stops at a number too close to zero that is nonzero, so this error happens only at insufficient precision. A third kind of error may happen, getting a denominator q for $\alpha \in \mathbb{Q}$ when the actual denominator is $p > q$, but this only happens at insufficient precision.

Let us consider a problem that will be of interest in the following chapters. The following is a brief description of concepts that will be introduced in §3.1. An *algebraic subgroup* of $\mathbb{C}^*$ is either a finite cyclic group (of roots of unity) or the whole $\mathbb{C}^*$. The algebraic group *generated* by $\lambda \in \mathbb{C}^*$ is the minimal algebraic

[7] For reference, the reader may refer to [HW75, Chap. X].

[8] The usual reference for Euclid's algorithm is the beginning of Book VII of the Elements, but this is Euclid's algorithm in $\mathbb{Z}$. The reference for Euclid's algorithm in $\mathbb{R}$ is the beginning of Book X. See [EH1908a] and [EH1908b] for an English translation, and [EP94] and [EP96] for a Spanish translation.

[9] The result is at the beginning of Book X of his Elements. Pappus credits Euclid with the first correct proof: "Euclid's object [...] was the attainment of irrefragable principles, which he established for commensurability and incommensurability in general." [PT30, p. 63f]

subgroup G_λ of $\mathbb{C}^*$ containing λ; if λ is a root of unity, then $G_\lambda = \lambda^{\mathbb{Z}}$; if λ is not a root of unity, then $G_\lambda = \mathbb{C}^*$. The *Lie algebra* $\mathfrak{g}$ of an algebraic subgroup G of $\mathbb{C}^*$ is a $\mathbb{C}$-linear subspace of $\mathbb{C}$; it is $\mathfrak{g} = 0$ if G is finite and it is $\mathfrak{g} = \mathbb{C}$ if $G = \mathbb{C}^*$. The problem of the Lie algebra $\mathfrak{g}_\lambda$ of G_λ is a way to convert the nonlinear problem of G_λ into a problem of linear algebra, addressed in §2.3.

If $\lambda \in \mathbb{C}^*$ is an effective complex number, testing if λ is a root of unity is equivalent to testing if α is rational for $\lambda = \mathrm{e}^{2\pi\mathrm{i}\alpha}$. Let us use the approximate test of rationality, getting the denominator $q \leqslant \mathsf{Q}$ if the test deems α rational, or $q = 1$ if the test deems α irrational. If the test were exact, then $\mathfrak{g}_\lambda = (\lambda^q - 1)\mathbb{C}$. A false positive in the test of rationality yields $\mathfrak{g}_\lambda = \mathbb{C}$ and $(\lambda^q - 1)\mathbb{C} = \mathbb{C}$. A false negative in the test of rationality yields $\mathfrak{g}_\lambda = 0$ but $(\lambda^q - 1)\mathbb{C} = \mathbb{C}$. An error of the third kind yields $\mathfrak{g}_\lambda = 0$ but $(\lambda^q - 1)\mathbb{C} = \mathbb{C}$, so this case is similar to a false negative. A way to be on the safe side is, if Euclid's algorithm stops at a number b_k too close to zero, to check that b_k is so small that, if it were not zero, the denominator of the next convergent would be greater than Q, which can be done using [HW75, thm. 149]. If this check fails, it fails due to insufficient precision and we should refine the precision as I shall explain in the following chapters.

Let me sketch the six possible cases:

1. The rationality test deems α rational with denominator q.
   (a) The check of b_k is passed.
      i. True positive: the actual denominator is q.
      ii. False positive: α is irrational.
      iii. Third-kind error: α is rational but its denominator is greater than Q.
   (b) The check of b_k is failed due to insufficient precision.
2. The rationality test deems α irrational.
   (a) True negative: α is irrational.
   (b) False negative: α is rational with denominator greater than Q.

The only cases where the $\mathfrak{g}_\lambda \neq (\lambda^q - 1)\mathbb{C}$ are cases 1(a)iii and 2b, which are precisely the cases when λ is a root of unity of order greater than Q. Let us consider the following truncation H_λ of G_λ: $H_\lambda = G_\lambda$ if λ is a root of unity of order up to Q, and $H_\lambda = \mathbb{C}^*$ for other λ. With this notation $(\lambda^q - 1)\mathbb{C}$ is the Lie algebra of H_λ.

This is a toy example for $\mathrm{GL}(1,\mathbb{C})$ of a general phenomenon in $\mathrm{GL}(n,\mathbb{C})$, and this truncation will inspire certain closure of algebraic groups in the next chapter.

This toy example suggests us to define, for $P \in \mathbb{N}$, the *P-truncated order* of $\lambda \in \mathbb{C}^*$ as its order p as a root of unity if $p \leqslant P$ and the P-truncated order is 1 if λ is a root of unity of order greater than P or λ is not a root of unity. According to this definition, the algorithm in this section computes the Q-truncated order with the exception of the third-kind error, when the Q-truncated order is 1 but we compute $q \leqslant \mathsf{Q}$. This exception occurs only at low precision, and does not matter for our use of the truncated order in §3.8.

We have shown that we cannot solve the problem of the rank over $\mathbb{Q}$ of $\{1, a\}$, so a fortiori we cannot solve the general problem of the rank over $\mathbb{Q}$, as needed in [vdH07a, §4.1].

2.5 Global parameters

In this chapter three global parameters were introduced. The tolerance tol was introduced in §2.1.3 as a global variable of type mpfpc (multiple-precision floating-point complex number) in order to determine when an effective complex number of small modulus is too close to zero and may be deemed zero in Gaussian elimination, as explained in §2.3, and in other computations that require finding the first nonzero term and divide by its coefficient, as in §4.2.2. Another global parameter, a natural number K, was introduced in §2.2.4 in order to determine how many terms in the principal part of a Laurent expansion must be zero-tested before you can deem it zero. A third global parameter, the natural number Q, was introduced in §2.4 in order to determine when a continuous fraction expansion corresponds to a rational number or not.

In the three cases we truncate an infinite process in order to yield an actual algorithm. In the first two cases, the idea of §4.5 is to restart the computation with *finer precision*, smaller tol and greater K, in a succession of tolerances converging to zero and K diverging to infinity. For each instance of number or Laurent expansion to zero-test, there is a step in the aforesaid succession such that, for all the subsequent values of tol or K, the zero-test gives the correct answer. The case of Q is different; the idea is to keep it fixed to a value with good properties discussed in §3.8.

In the following chapters I shall introduce some global parameters of the kind

of K, but not of the kind of tol or Q. The tolerance tol can be reduced to the kind of K by taking $\mathsf{tol} = 2^{-\mathsf{T}}$ for a global parameter T. So we may consider Q a *special global parameter* and the global parameters of kind of K the *general global parameters*. The tolerance tol could be considered another special global parameter, but in §4.5 it will be considered a general one through T.

Remark 50. Global variables are those variables that are defined outside any function or procedure and can be accessed from any function or procedure. Global variables are "considered harmful," cf. [WS73], same as the go-to statement. Both are taboo in structured programming, and they make the code less legible and analyzable by humans, but there are restricted usages where they can improve the code legibility. The greatest danger with global variables is that they may be modified anywhere, but the global variables proposed in this thesis are modified only by the main algorithm (in §4.5), being read-only variables elsewhere. In C++ this feature can be achieved defining the global variables in the file of the main algorithm and declaring them in the rest of the files as `const`. Anyway, gotos and global variables are often used when sketching algorithms.

HERE ENDETH THE SECOND CHAPTER ✠

Chapter 3

Linear algebraic groups

In this chapter, which is almost independent of the previous ones, we approach one of the central constructions in the thesis. Indeed, as discussed in §2.4, there are numerical issues making impossible in practice the computation of the Galois group when the data are not exact. Nevertheless, for the computation of the common eigenvectors by the identity component of the Galois group, we shall see that we can replace this group with a larger one whose identity component has the same common eigenvectors.

Example 51. The algebraic group generated by $\left(\begin{smallmatrix} 2 & 0 \\ 0 & \frac{1}{2} \end{smallmatrix}\right)$ is $\left\{\left(\begin{smallmatrix} a & 0 \\ 0 & a^{-1} \end{smallmatrix}\right) : a \in \mathbb{K}^*\right\}$, connected. Its common eigenvectors are the two axes, the same as the whole diagonal group, which is larger. Moreover, we observe that the diagonal group is the multiplicative group of the diagonal algebra, an observation that will suggest the definition of broad groups in §3.3.

In the first section we shall review the theory of linear algebraic groups. For further details one may refer to [Hum81], [Bor91] or [TY05]. Then we shall describe the augmentation of the Galois group and the computation of these groups form their generators.

3.1 Linear algebraic groups

Let $\mathbb{K}$ be an algebraically closed field (of characteristic 0) and V an n-dimensional vector space over $\mathbb{K}$. We consider $\mathrm{GL}(n, \mathbb{K})$ with the Zariski topology. Any basis of V identifies $\mathrm{GL}(V) \simeq \mathrm{GL}(n, \mathbb{K})$ and endows $\mathrm{GL}(V)$ with the Zariski topology,

independently of the basis. We define a *linear algebraic group* (or simply *algebraic group*) as a subgroup of $\mathrm{GL}(V)$ or $\mathrm{GL}(n,\mathbb{K})$ that is closed in the Zariski topology.

A linear algebraic group is a smooth algebraic variety with finitely many irreducible components, which are also its connected components, hence they will simply be called components, and they are thus disjoint. If $\mathbb{K}=\mathbb{C}$, linear algebraic groups are Lie groups and their components are also the connected components in the differentiable structure. The component of the identity (or *identity component*) of an algebraic group G is denoted by G° and is a connected algebraic subgroup. Moreover G° is a normal subgroup of G of finite index whose cosets are the components of G.

The algebraic group generated by a subset A of $\mathrm{GL}(V)$ or $\mathrm{GL}(n,\mathbb{K})$ is the Zariski closure of the group generated by A. For instance, for $n=1$, we identify $\mathrm{GL}_1(\mathbb{K})\simeq\mathbb{K}^*$. The group generated by $\lambda\in\mathbb{K}^*$ is finite if λ is a root of unity and it is infinite if not. In the first case, the finite group is an algebraic group, which is discrete. In the latter case the group is dense in the Zariski topology, thus the algebraic group generated by λ is $\mathbb{K}^*$, which is connected. The case of A a single element and general n will be considered later, in §3.2.

Example 52. The simplest example of a linear algebraic group is the general linear group $\mathrm{GL}(V)$ or $\mathrm{GL}(n,\mathbb{K})$ itself. Any finite subgroup is also an algebraic group. The special linear group $\mathrm{SL}(V)$ or $\mathrm{SL}(n,\mathbb{K})$ is the algebraic group defined by the equation $\det=1$. The orthogonal group $\mathrm{O}(n,\mathbb{K})$, which consists of the matrices $\mathtt{M}$ such that $\mathtt{M}\mathtt{M}^\intercal=\mathtt{I}$, is another algebraic group. The special orthogonal group $\mathrm{SO}(n,\mathbb{K})=\mathrm{SL}(n,\mathbb{K})\cap\mathrm{O}(n,\mathbb{K})$ is also an algebraic group. The unitary group $\mathrm{U}(n)$, which consists of the complex matrices $\mathtt{M}$ such that $\mathtt{M}\bar{\mathtt{M}}^\intercal=\mathtt{I}$, is not an algebraic group for $\mathbb{K}=\mathbb{C}$, and neither is the special unitary group $\mathrm{SU}(n)=\mathrm{SL}(n,\mathbb{C})\cap\mathrm{U}(n)$.

We associate to each algebraic group another algebraic structure called Lie algebra. A *Lie algebra* is a vector space $\mathfrak{g}$ with a bilinear operation $[\ ,\ ]$, called the *Lie bracket*, such that $[a,a]=0$ and $[a,[b,c]]+[b,[c,a]]+[c,[a,b]]=0$ for any elements $a,b,c\in\mathfrak{g}$. This is the natural structure of the set of derivations over a ring with the Lie bracket given by $[\delta_1,\delta_2]=\delta_1\circ\delta_2-\delta_2\circ\delta_1$. Any $\mathbb{K}$-algebra can be made a Lie algebra with the Lie bracket given by $[a,b]=ab-ba$. This way we construct the Lie algebras $\mathfrak{gl}(n,\mathbb{K})=\mathbb{K}^{n\times n}$ of matrices and $\mathfrak{gl}(V)=\mathcal{L}(V,V)$ of linear endomorphism.

We associate to $\mathrm{GL}(n,\mathbb{K})$ the Lie algebra $\mathfrak{gl}(n,\mathbb{K})$, and to $\mathrm{GL}(V)$ the Lie algebra $\mathfrak{gl}(V)$. Let $\mathcal{G}$ be the lattice of algebraic subgroups of $\mathrm{GL}(n,\mathbb{K})$ or $\mathrm{GL}(V)$ and $\mathcal{A}$ the lattice of Lie subalgebras of the corresponding $\mathfrak{gl}(n,\mathbb{K})$ or $\mathfrak{gl}(V)$. There is a correspondence $\mathfrak{L}:\mathcal{G}\to\mathcal{A}$ in the following way. If $G=\{\mathtt{M}\in\mathrm{GL}(n,\mathbb{K}):\forall P\in\mathcal{F},$

$P(\underline{\mathtt{M}}) = 0\}$, where $\mathcal{F}$ is a family of polynomials in n^2 variables and a matrix underlined is the list of its entries row after row,[1] is $\mathfrak{L}(G) = \{\mathtt{M} \in \mathfrak{gl}(n, \mathbb{K}) : \forall P \in \mathcal{F}, \mathrm{d}_{(\underline{\mathtt{I}})}(P)(\underline{\mathtt{M}}) = 0\}$, where $\mathrm{d}_{(\underline{\mathtt{I}})}(P)$ is the linear homogeneous polynomial tangent to P at the identity. A way to compute $\mathrm{d}_{(\underline{\mathtt{I}})}(P)$, according to [Ser92, Ch. I], is $P(\underline{\mathtt{I} + x\mathtt{M}}) = P(\underline{\mathtt{I}}) + x\mathrm{d}_{(\underline{\mathtt{I}})}(P)(\underline{\mathtt{M}}) + o(x)$. Another way is

$$\mathrm{d}_{(\underline{\mathtt{I}})}(P) = \sum_{i,j=1}^{n} \left.\frac{\partial P}{\partial x_{ij}}\right|_{(\underline{\mathtt{I}})} x_{ij}.$$

Let $\mathcal{G}_0 \subset \mathcal{G}$ be the subfamily of the connected algebraic groups and $\mathcal{A}_0$ the image of $\mathfrak{L}$. According to [Bor91, §7.1, §7.7], $\mathcal{A}_0$ is a sublattice of $\mathcal{A}$. The restriction $\mathfrak{L} : \mathcal{G}_0 \to \mathcal{A}_0$ is an order-preserving bijection whose inverse is also order-preserving, according to [Bor91, §7.1], thus $\mathfrak{L}$ yields an isomorphism of lattices $\mathcal{G}_0 \simeq \mathcal{A}_0$. By [Bor91, §7.1] and [TY05, Cor. 21.3.2], $\mathcal{G}_0$ is a sublattice of $\mathcal{G}$. The correspondence $\mathfrak{L} : \mathcal{G} \to \mathcal{A}$ factors as $\mathcal{G} \to \mathcal{G}_0 \simeq \mathcal{A}_0$, where the first correspondence is $G \mapsto G^\circ$. If $\mathbb{K} = \mathbb{C}$, $\mathfrak{L}$ is the usual correspondence between Lie groups and Lie algebras $\mathfrak{L}(G) = \{\mathtt{M} \in \mathfrak{gl}(n, \mathbb{C}) : \forall t \in \mathbb{C}, \exp(t\mathtt{M}) \in G\}$, cf. [Hal03, Def. 2.15]. There are other definitions of the Lie algebra associated to a Lie group or an algebraic group, more general than these, that use the tangent space at the identity.

Example 53. The special linear group $\mathrm{SL}(n, \mathbb{K})$ is the algebraic group given by the equation $\det = 1$ and its corresponding Lie algebra is $\mathfrak{sl}(n, \mathbb{K})$, given by the equation $\mathrm{trace} = 0$. The diagonal group is an algebraic group and its Lie algebra consists of all the diagonal matrices. The multiplicative group $(\mathbb{K}^*, \cdot)$ is identified with $\mathrm{GL}(1, \mathbb{K})$, whose Lie algebra is $\mathfrak{gl}(1, \mathbb{K})$, identified with $\mathbb{K}$. The additive group $(\mathbb{K}, +)$ is identified by means of the mapping

$$a \mapsto \begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix}$$

with the algebraic group of equation $x_{11} - 1 = x_{22} - 1 = x_{21} = 0$, whose Lie algebra is given by the equation $x_{11} = x_{22} = x_{21} = 0$.

Example 54. If $A \subset \mathbb{Z}^n$,

$$\{\mathrm{diag}(a_1, a_2, \ldots, a_n) \in \mathrm{GL}(n, \mathbb{K}) : \forall (k_1, k_2, \ldots, k_n) \in A, a_1^{k_1} a_2^{k_2} \cdots a_n^{k_n} = 1\}$$

is an algebraic group and its Lie algebra is

$$\{\mathrm{diag}(a_1, a_2, \ldots, a_n) \in \mathfrak{gl}(n, \mathbb{K}) : \forall (k_1, k_2, \ldots, k_n) \in A, k_1 a_1 + k_2 a_2 + \cdots + k_n a_n = 0\}.$$

[1] I introduce this notation in order to avoid confusion with the application of a polynomial to a matrix. This distinction is key the proof of Theorem 69.

As a particular case, consider $G_m = \{\mathrm{diag}(a,b) \in \mathrm{GL}(2,\mathbb{K}) : a^m = b^m\}$ for m natural. The Lie algebra of G_m is

$$\{\mathrm{diag}(a,b) \in \mathfrak{gl}(2,\mathbb{K}) : ma = mb\} = \{a\mathtt{I}_2 : a \in \mathbb{K}\},$$

which is identified with $\mathbb{K}$. The components of G_m are $X_\xi = \{\mathrm{diag}(a,\xi a) : a \in \mathbb{K}^*\}$ for ξ an mth root of unity, thus $G_m^\circ = G_1 = \{a\mathtt{I}_2 : a \in \mathbb{K}^*\}$, which is identified with $\mathbb{K}^*$.

We decompose $\mathtt{M} = \mathtt{DU}$ with $\mathtt{D}$ is diagonalizable and $\mathtt{U}$ unipotent, i.e., 1 is the only eigenvalue of $\mathtt{U}$. This decomposition is unique and $\mathtt{D}$ and $\mathtt{U}$ commute. An interesting property of algebraic groups is that multiplicative Jordan parts of an element belong also to the group. A proof can be found in [Bor91, §4.2]. For instance, the group generated by $\left(\begin{smallmatrix}2&1\\0&2\end{smallmatrix}\right)$ does not contain the diagonal part $\left(\begin{smallmatrix}2&0\\0&2\end{smallmatrix}\right)$ nor the unipotent part $\left(\begin{smallmatrix}1&1\\0&1\end{smallmatrix}\right)$, but it is not an algebraic group.

3.2 Derksen–van der Hoeven algorithm

The algebraic group generated by a single element is expounded in [vdH07a, §4.3] and [DJK05, §3.3]. We need the multiplicative Jordan decomposition of the generator $\mathtt{M} = \mathtt{DU}$, where $\mathtt{D}$ is diagonalizable and $\mathtt{U}$ unipotent, i.e., 1 is the only eigenvalue of $\mathtt{U}$. The algebraic group generated by $\mathtt{M}$ is the product of the (commuting) algebraic groups generated by $\mathtt{D}$ and $\mathtt{U}$, and its Lie algebra is the sum of the corresponding Lie algebras. Assume that $\mathtt{D}$ is a diagonal matrix $\mathtt{D} = \mathrm{diag}(a_1, a_2, \ldots, a_n)$ and be A the set of the *multiplicative syzygies* of $(a_1, a_2, \ldots, a_n)$, defined as $A = \{(k_1, k_2, \ldots, k_n) \in \mathbb{Z}^n : a_1^{k_1} a_2^{k_2} \cdots a_n^{k_n} = 1\}$. The algebraic group generated by $\mathtt{D}$ is

$$\{\mathrm{diag}(b_1, b_2, \ldots, b_n) \in \mathrm{GL}(n,\mathbb{K}) : \forall (k_1, k_2, \ldots, k_n) \in A, b_1^{k_1} b_2^{k_2} \cdots b_n^{k_n} = 1\},$$

of a kind studied in Example 54; see [vdH07a, Lem. 2] and [DJK05, Lem. 6]. As $\mathtt{U} - \mathtt{I}$ is nilpotent, $\log \mathtt{U} = \sum_{k=0}^{\infty} (-1)^k (\mathtt{U} - \mathtt{I})^{k+1}/(k+1)$ is a polynomial in $\mathtt{U}$. As $\log \mathtt{U}$ is nilpotent, $\exp(t \log \mathtt{U}) = \sum_{k=0}^{\infty} (t \log \mathtt{U})^k / k!$ is a polynomial in $\log \mathtt{U}$, and hence in $\mathtt{U}$, for any t. The algebraic group generated by $\mathtt{U}$ is $\{\exp(t \log \mathtt{U}) : t \in \mathbb{K}\}$ and its Lie algebra is $\mathbb{K} \log \mathtt{U}$.

The algebraic group generated by finitely many elements is expounded in [vdH07a, §4.5] and [DJK05, §3.4]. In these articles, an algorithm to compute this algebraic group is developed: I shall call it Derksen–van der Hoeven algorithm.

There are two versions of the algorithm: the version of [DJK05] works with algebraic groups as algebraic varieties given as Gröbner bases of their defining ideal, and the version of [vdH07a] works with Lie algebras, which is the setting I use. Let $\mathcal{M}$ be the finite set of generators, G the algebraic group they generate and $\mathfrak{g}$ its Lie algebra. The algorithm works with a finite family $\mathcal{F}$, initially $\mathcal{M}$, and a Lie algebra $\mathfrak{a}$, initially zero, which are augmented by a loop until they stabilize; in this case the algorithm terminates with $\mathfrak{a} = \mathfrak{g}$ and $\mathcal{F}$ a system of representatives of G/G°. Each iteration of the loop performs the following steps.

1. For each $\mathtt{A} \in \mathcal{F}$ we augment $\mathfrak{a}$, as a Lie algebra, with the Lie algebra of the algebraic group generated by $\mathtt{A}$.
2. For each $\mathtt{A} \in \mathcal{F}$ we augment $\mathfrak{a}$, as a Lie algebra, with $\mathtt{A}\mathfrak{a}\mathtt{A}^{-1}$.
3. For each $\mathtt{A} \in \mathcal{F}$ we check if $\mathtt{A}$ is equivalent to any element of $\mathcal{F}$ modulo the connected group corresponding to $\mathfrak{a}$; if it is, we eliminate $\mathtt{A}$ from $\mathcal{F}$.
4. For each ordered pair $\mathtt{A}, \mathtt{B} \in \mathcal{F}$ we check if $\mathtt{AB}$ is equivalent to any element of $\mathcal{F}$ modulo the connected group corresponding to $\mathfrak{a}$; if it is not, we add $\mathtt{AB}$ to $\mathcal{F}$.

Example 55. Let us compute the algebraic group generated by $\mathtt{M} = \left(\begin{smallmatrix} 2 & 0 \\ 0 & \frac{1}{2} \end{smallmatrix}\right)$.
Initially $\mathcal{F} = \{\mathtt{M}\}$ and $\mathfrak{a} = 0$.
First iteration, step 1: we augment $\mathfrak{a}$ with the Lie algebra of the algebraic group generated by $\mathtt{M}$. The multiplicative syzygies of $(2, \frac{1}{2})$ are $\mathbb{Z}(1, 1)$; according to Example 54, the Lie algebra is $\mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$, so now $\mathfrak{a} = \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$.
Step 2: as $\mathtt{M}\mathfrak{a}\mathtt{M}^{-1} = \mathfrak{a}$, we keep $\mathfrak{a} = \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$.
Step 3: no duplicates.
Step 4: is $\mathtt{M}^2$ a duplicate of $\mathtt{M}$? It is, thus we keep $\mathcal{F} = \{\mathtt{M}\}$.
Second iteration, step 1: we augment $\mathtt{a}$ as a Lie algebra of the algebraic group generated by $\mathtt{M}$, which is already contained, so we keep $\mathfrak{a}$ invariant.
Step 2: as $\mathtt{M}\mathfrak{a}\mathtt{M}^{-1} = \mathfrak{a}$, we keep $\mathfrak{a}$ invariant.
Step 3: same as in the previous iteration.
Step 4: same as in the previous iteration.
The algorithm ends with $\mathfrak{g} = \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$ and G/G° represented by $\{\mathtt{M}\}$.

As the example shows, an optimization could be "marking" the elements of $\mathcal{F}$ used in step 1 and then excluding the marked elements in step 1.

Let us consider the effects of §2.4 in the computation of the algebraic group G generated by a matrix (λ) with $\lambda \in \mathbb{C}^*$ effective. The multiplicative syzygies

$\{k \in \mathbb{Z} : \lambda^k = 1\}$ are $m\mathbb{Z}$, with m the order of λ as a root of unity or $m = 0$ if λ is not a root of unity. Then

$$G = \{(\mu) \in \mathrm{GL}(1, \mathbb{C}) : \mu^{mk} = 1 \ \forall k \in \mathbb{Z}\} = \{(\mu) \in \mathrm{GL}(1, \mathbb{C}) : \mu^m = 1\}$$

is the group of m-th roots of unity, with $G = \mathrm{GL}(1, \mathbb{C})$ for $m = 0$. This reduces to the rationality test of α with $\lambda = \mathrm{e}^{2\pi \mathrm{i} \alpha}$ expounded in §2.4. This test outputs a denominator q of α, or $q = 1$ if α is deemed irrational. As shown in §2.4, we must resign ourselves to computing a truncation of the algebraic group where G is estimated at $\mathrm{GL}(1, \mathbb{C})$ for $m > \mathtt{Q}$. Assuming this overestimation, G° is generated by (λ^q). The overestimated Lie algebra $\mathfrak{g}$ of G will be zero if $\lambda^q = 1$, and $\mathfrak{g} = \mathfrak{gl}(1, \mathbb{C})$ if $\lambda^q \neq 1$, so we may take $\lambda^q - 1$ as the generator of $\mathfrak{g}$. If it happens that λ^q is to close to 1, the error is now of linear algebra, discussed in §2.3.

The previous paragraph presents a toy example, since order-1 differential equations are uninteresting for our purposes. The interesting examples require to compute the multiplicative syzygies of several effective complex numbers, which is more complicated than the rationality test of §2.4. Another problem is that the test of equivalence modulo the connected group corresponding to a given Lie algebra that we need in steps 3 and 4 depends on [vdH07a, §4.4], which depends on heuristics. In order to avoid this issue, I introduce broad and eurymeric groups in the following sections.

3.3 Broad and eurymeric groups

In this section I introduce broad and eurymeric groups, certain kinds of algebraic groups that will be key in this thesis.

The $\mathbb{K}$-algebras $\mathcal{L}(V, V)$ and $\mathbb{K}^{n \times n}$ have a rich structure with three operations: addition, multiplication (or composition) and product by scalars. Their respective multiplicative groups $\mathrm{GL}(V)$ and $\mathrm{GL}(n, \mathbb{K})$ are constructed by forgetting the other two operations and dropping the elements that are not invertible. The Lie algebras $\mathfrak{gl}(V)$ and $\mathfrak{gl}(n, \mathbb{K})$ are constructed respectively from $\mathcal{L}(V, V)$ and $\mathbb{K}^{n \times n}$ by partially forgetting multiplication: we remind only Lie brackets $[\mathtt{A}, \mathtt{B}] = \mathtt{AB} - \mathtt{BA}$. I will write GL for $\mathrm{GL}(V)$ or $\mathrm{GL}(n, \mathbb{K})$. A subalgebra[2] A of $\mathcal{L}(V, V)$ or $\mathbb{K}^{n \times n}$ have the same rich structure, and allows the same forgetful transformations into a group G and into a Lie algebra $\mathfrak{g}$. The group G is $A \cap \mathrm{GL}$ as a set, according to Lemma 56, and

[2] A subalgebra of the algebra structure with addition, multiplication/composition and product by scalars, which is stronger than a Lie subalgebra. Notice that $\mathtt{I} \in A$.

A is a vector space, thus an irreducible algebraic variety, hence G is an irreducible algebraic variety of GL and thus a connected algebraic group. Its corresponding Lie algebra is given by the linearization of the equations of G. As G is defined by linear equations, its Lie algebra is defined by the same equations, thus it is $\mathfrak{g}$. Conversely, G is the connected algebraic group corresponding to $\mathfrak{g}$.

Lemma 56. *If A is a subalgebra of $\mathfrak{gl}(V)$ or $\mathfrak{gl}(n,\mathbb{K})$ and $\mathtt{M} \in A$ is invertible, then $\mathtt{M}^{-1} \in A$.*

Proof. Let $a_0 + a_1 x + \cdots + a_p x^p$ be the minimal polynomial of $\mathtt{M}$, where $a_0 \neq 0$, hence $\mathtt{M}^{-1} = -\frac{a_1}{a_0}\mathtt{I} - \frac{a_2}{a_0}\mathtt{M} - \cdots - \frac{a_p}{a_0}\mathtt{M}^{p-1}$. □

The multiplicative group of a subalgebra of $\mathfrak{gl}(V)$ or $\mathfrak{gl}(n,\mathbb{K})$ will be called a *broad group*. A Lie subalgebra of $\mathfrak{gl}(V)$ or $\mathfrak{gl}(n,\mathbb{K})$ that is also a subalgebra will be called a *broad Lie algebra*. We have proved that a broad group G can be recovered from its Lie algebra $\mathfrak{g}$ by $G = \mathfrak{g} \cap \mathrm{GL}$. The following lemma gives a converse.

Lemma 57. *If G is a broad group, its Lie algebra $\mathfrak{g}$ is the linear span of G.*

Proof. If $\mathtt{M} \in \mathfrak{g}$ and $\lambda \in \mathbb{K}^*$ is not an eigenvalue of $\mathtt{M}$, $\mathtt{M} - \lambda\mathtt{I}$ and $\lambda\mathtt{I}$ belong to $\mathfrak{g} \cap \mathrm{GL} = G$, thus $\mathtt{M}$ is the sum of two elements of G. □

The terminology of broad groups and broad Lie algebras is because the former have room enough for addition, and the latter have room enough for multiplication and the identity. Conversely, a Lie algebra with room enough for multiplication and the identity is broad. The following result is the converse for broad groups.

Lemma 58. *An algebraic group given by linear equations is broad.*

Proof. Let $G = V \cap \mathrm{GL}$ be the algebraic group and V a vector space. We have $\mathbb{K}\mathtt{I} \subset V$. If $\mathtt{M} \in V$ and $\lambda \in \mathbb{K}^*$ is not an eigenvalue of $\mathtt{M}$, $\mathtt{M} - \lambda\mathtt{I}$ and $\lambda\mathtt{I}$ belong to $V \cap \mathrm{GL} = G$. So any element of V is sum of two elements of G. If $\mathtt{A}, B \in V$, we have decompositions $\mathtt{A} = \mathtt{A}_1 + \mathtt{A}_2$ and $\mathtt{B} = \mathtt{B}_1 + \mathtt{B}_2$ in sum of elements of G, hence $\mathtt{AB} = \mathtt{A}_1\mathtt{B}_1 + \mathtt{A}_1\mathtt{B}_2 + \mathtt{A}_2\mathtt{B}_1 + \mathtt{A}_2\mathtt{B}_2$ is sum of elements of G, so $\mathtt{AB} \in V$. We have that V is a subalgebra, and thus G is a broad group. □

The broad group generated by $\mathcal{M} \subset \mathrm{GL}$ is $\mathbb{K}[\mathcal{M}] \cap \mathrm{GL}$, and its Lie algebra $\mathbb{K}[\mathcal{M}]$, where $\mathbb{K}[\mathcal{M}]$ is the the $\mathbb{K}$-algebra generated by $\mathcal{M}$. If H is a subgroup of GL, $\mathbb{K}[H]$ is simply its linear span, and we call $\mathbb{K}[H] \cap \mathrm{GL}$ its *broad hull*. The

broad Lie algebra generated by $\mathcal{M} \subset \mathfrak{gl}$ is $\mathfrak{g} = \mathbb{K}[\mathcal{M}]$. The broad groups form a lattice with the intersection and $\mathbb{K}[G_1, G_2] \cap \mathrm{GL}$ the join of G_1 and G_2. The maximal broad group is $\mathrm{GL}(V)$ or $\mathrm{GL}(n, \mathbb{K})$, and the minimal one is $\mathbb{K}^*\mathtt{I}$.

Example 59. Let us compute the broad hull of an algebraic torus

$$G = \left\{\mathrm{diag}\left(\beta_1^{k_{11}}\beta_2^{k_{12}}\cdots\beta_r^{k_{1r}}, \beta_1^{k_{21}}\beta_2^{k_{22}}\cdots\beta_r^{k_{2r}}, \ldots, \beta_1^{k_{n1}}\beta_2^{k_{n2}}\cdots\beta_r^{k_{nr}}\right) : \beta_1, \beta_2, \ldots, \beta_r \in \mathbb{K}^*\right\}$$

with $k_{ij} \in \mathbb{Z}$. First, we construct an element $\mathtt{A} = \mathrm{diag}(\alpha_1, \alpha_2, \ldots, \alpha_n) \in G$ such that $\alpha_i = \alpha_j$ if and only if $(k_{i1}, k_{i2}, \ldots, k_{ir}) = (k_{j1}, k_{j2}, \ldots, k_{jr})$. If $p_1, p_2, \ldots, p_r$ are distinct primes, then

$$\mathtt{A} = \mathrm{diag}\left(p_1^{k_{11}}p_2^{k_{12}}\cdots p_r^{k_{1r}}, p_1^{k_{21}}p_2^{k_{22}}\cdots p_r^{k_{2r}}, \ldots, p_1^{k_{n1}}p_2^{k_{n2}}\cdots p_r^{k_{nr}}\right)$$

satisfies the property. Let R be the $\mathbb{K}$-algebra of the matrices $\mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_n)$ such that $\mu_1, \mu_2, \ldots, \mu_n \in \mathbb{K}$, with $\mu_i = \mu_j$ whenever $\alpha_i = \alpha_j$. Given $\mathtt{M} = \mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_n) \in R$, we can construct the Lagrange interpolating polynomial $P(x) \in \mathbb{K}[x]$ such that $P(\alpha_i) = \mu_i$ for $1 \leqslant i \leqslant n$, so that $P(\mathtt{A}) = \mathtt{M}$. This proves that $\mathbb{K}[\mathtt{A}] = R$. As $\mathbb{K}[\mathtt{A}] \subset \mathbb{K}[G] \subset R$, we have $\mathbb{K}[G] = R$, so the broad hull H of G is the group of the matrices $\mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_n)$ such that $\mu_1, \mu_2, \ldots, \mu_n \in \mathbb{K}^*$, with $\mu_i = \mu_j$ whenever $\alpha_i = \alpha_j$, which is another algebraic torus. If the $\mathbf{k}_i = (k_{i1}, k_{i2}, \ldots, k_{ir})$ are numbered without repetition as $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_m$ and we have $\mathbf{k}_i = \mathbf{h}_{f(i)}$ for $1 \leqslant i \leqslant n$, then

$$H = \left\{\mathrm{diag}\left(\beta_{f(1)}, \beta_{f(2)}, \ldots, \beta_{f(n)}\right) : \beta_1, \beta_2, \ldots, \beta_m \in \mathbb{K}^*\right\}$$

and its Lie algebra is

$$\left\{\mathrm{diag}\left(b_{f(1)}, b_{f(2)}, \ldots, b_{f(n)}\right) : b_1, b_2, \ldots, b_m \in \mathbb{K}\right\}.$$

An algebraic group G whose identity component G° is broad will be called a *eurymeric group*, where *eury-* means 'broad' in Greek, and *mero-* 'part,'[3] in this case the identity component. Eurymeric is equivalent to *virtually*[4] *broad* and *broad-by-finite*.[5] Indeed, if G is eurymeric, G° is a normal finite-index broad subgroup, so G is broad-by-finite and thus virtually broad. If G is virtually broad, it has a finite-index broad subgroup H, so the cosets modulo H are finitely many linear varieties, thus G is an algebraic variety and therefore an algebraic group. The only finite-index connected subgroup of an algebraic group is its identity component, thus G is eurymeric.

[3]There is another word *eurymeric* where *mero-* means 'thigh,' in this case 'femur.'

[4]If P is a property, a group is virtually P if it has a finite-index P subgroup.

[5]If P is a property, a group is P-by-finite if it has a normal finite-index P subgroup.

The intersection of eurymeric groups is eurymeric. Indeed, if G and H are eurymeric groups, G has components of the form $V_i \cap \mathrm{GL}$ with V_i vector spaces, and $W_j \cap \mathrm{GL}$ for H, then $G \cap H$ has disjoint components $V_i \cap W_j \cap \mathrm{GL}$, when nonempty, and $(G \cap H)^\circ = G^\circ \cap H^\circ$ is broad. The eurymeric group generated by two eurymeric groups is more complicated to describe; it requires a modification of the Derksen–van der Hoeven algorithm introduced in §3.2. It will be discussed in §3.6.

3.4 Algebraic subgroups of $\mathrm{GL}(2,\mathbb{C})$

In this section and the next one, as an illustration of the notions of broad and eurymeric groups just introduced, we shall compute them for $n = 2$. First, for the sake of completeness, we recall the classification of all the algebraic subgroups of $\mathrm{GL}(2,\mathbb{C})$.

The algebraic subgroups of $\mathrm{GL}(2,\mathbb{C})$ are studied in [Ngu08, App. A]=[NvdPT08]. We consider the projectivizations $\mathrm{GL}(2,\mathbb{C}) \to \mathrm{PGL}(2,\mathbb{C}) = \mathrm{GL}(2,\mathbb{C})/\mathbb{C}^*\mathtt{I}$ and $\mathrm{SL}(2,\mathbb{C}) \to \mathrm{PSL}(2,\mathbb{C}) = \mathrm{SL}(2,\mathbb{C})/\{\mathtt{I},-\mathtt{I}\}$ and observe that $\mathrm{PGL}(2,\mathbb{C})$ and $\mathrm{PSL}(2,\mathbb{C})$ are naturally isomorphic. Any subgroup H of either has counterimages $H^{\mathrm{GL}(2,\mathbb{C})}$ and $H^{\mathrm{SL}(2,\mathbb{C})}$. The former is the maximal subgroup of $\mathrm{GL}(2,\mathbb{C})$ whose projective image is H. The algebraic subgroups of $\mathrm{SL}(2,\mathbb{C})$ are well known, and shall be explained below, so their projective images are the algebraic subgroups of $\mathrm{PGL}(2,\mathbb{C})$. For each algebraic subgroup H of $\mathrm{PGL}(2,\mathbb{C})$, Nguyen studies the minimal algebraic subgroups of $\mathrm{GL}(2,\mathbb{C})$ with H as projective image; the list of these minimal groups, up to conjugation, is in [Ngu08, Thm. A.2.10]=[NvdPT08, Thm. 4]. For each minimal group G, we have the maximal group $\mathbb{C}^* \cdot G$ and $\mu_k \cdot G$, where μ_k is the group of k-th roots of unity. These are all the algebraic subgroups of $\mathrm{GL}(2,\mathbb{C})$.

The algebraic subgroups of $\mathrm{SL}(2,\mathbb{C})$ are, up to conjugation, the following, cf. [MR99, Prop. 2.2]:

- a finite group, totally disconnected, with null Lie algebra,
- the multiplicative group $\left\{\begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^*\right\} \simeq (\mathbb{C}^*,\cdot)$, connected, with Lie algebra $\left\{\begin{pmatrix} b & 0 \\ 0 & -b \end{pmatrix} : b \in \mathbb{C}\right\}$,
- the additive group $\left\{\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix} : a \in \mathbb{C}\right\} \simeq (\mathbb{C},+)$, connected, with Lie algebra

$\left\{\begin{pmatrix} 0 & b \\ 0 & 0 \end{pmatrix} : b \in \mathbb{C}\right\}$,

- the Borel group $\left\{\begin{pmatrix} a & b \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^*, b \in \mathbb{C}\right\}$, connected, with Lie algebra $\left\{\begin{pmatrix} c & d \\ 0 & -c \end{pmatrix} : c, d \in \mathbb{C}\right\}$,
- $\left\{\begin{pmatrix} a & b \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^*, a^k = 1, b \in \mathbb{C}\right\}$, with $k \in \mathbb{N}$, whose identity component is the additive group,
- the dihedral group $\left\{\begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix}, \begin{pmatrix} 0 & -a \\ a^{-1} & 0 \end{pmatrix} : a \in \mathbb{C}^*\right\}$, whose identity component is the multiplicative group,
- $\mathrm{SL}(2, \mathbb{C})$, connected, with Lie algebra $\mathfrak{sl}(2, \mathbb{C})$.

The finite subgroups of $\mathrm{SL}(2, \mathbb{C})$ are of five kinds up to conjugation:

- cyclic groups $\left\{\begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^*, a^k = 1\right\}$ for $k \in \mathbb{N}$,
- dihedral groups $\left\{\begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix}, \begin{pmatrix} 0 & -a \\ a^{-1} & 0 \end{pmatrix} : a \in \mathbb{C}^*, a^k = 1\right\}$ for k even,
- the tetrahedral group $\mathbf{A}_4^{\mathrm{SL}(2,\mathbb{C})}$,
- the octahedral group $\mathbf{S}_4^{\mathrm{SL}(2,\mathbb{C})}$,
- the icosahedral group $\mathbf{A}_5^{\mathrm{SL}(2,\mathbb{C})}$.

The denominations alluding to polyhedra and permutation groups will be explained below.

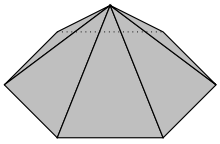
Consider $\mathfrak{sl}(2, \mathbb{C})$ as a vector space endowed with the bilinear form given by $B(\mathtt{M}, \mathtt{N}) = \frac{1}{2}\operatorname{tr}(\mathtt{MN})$. As $\mathfrak{sl}(2, \mathbb{C})$ is invariant by conjugation, the action of $\mathrm{GL}(2, \mathbb{C})$ by conjugation defines a homomorphism $\mathrm{GL}(2, \mathbb{C}) \to \mathrm{GL}(\mathfrak{sl}(2, \mathbb{C}))$. One may check that the kernel is $\mathbb{C}^*\mathtt{I}$ and that the image is contained in $\mathrm{SO}(\mathfrak{sl}(2, \mathbb{C}), B)$. Let f be the homomorphism $\mathrm{GL}(2, \mathbb{C}) \to \mathrm{SO}(\mathfrak{sl}(2, \mathbb{C}), B)$, and f' the induced[6] homomorphism $\mathfrak{gl}(2, \mathbb{C}) \to \mathfrak{so}(\mathfrak{sl}(2, \mathbb{C}), B)$ of Lie algebras. The kernel of f' is, according to

[6] As described in [TY05, §23.2.4], f' is the differential of f and the identity.

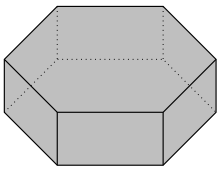
[TY05, Thm. 24.4.1], the Lie algebra of the kernel of f, thus $\mathbb{C}\cdot\mathtt{I}$. By a matter of dimensions, the image of f' is $\mathfrak{so}(\mathfrak{sl}(2,\mathbb{C}),B)$. According to [TY05, Prop. 24.5.3 (i)], the Lie algebra of the image of f is $\mathfrak{so}(\mathfrak{sl}(2,\mathbb{C}),B)$. As $\mathrm{SO}(\mathfrak{sl}(2,\mathbb{C}),B)$ is the connected algebraic group corresponding to $\mathfrak{so}(\mathfrak{sl}(2,\mathbb{C}),B)$, f is surjective, so it induces an isomorphism $\mathrm{PGL}(2,\mathbb{C}) \simeq \mathrm{SO}(\mathfrak{sl}(2,\mathbb{C}),B)$ and, in the orthonormal basis

$$\left\{\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}, \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}, \begin{pmatrix}0 & \mathrm{i}\\ -\mathrm{i} & 0\end{pmatrix}\right\},$$

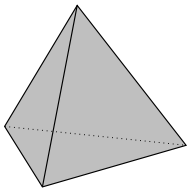
$\mathrm{PGL}(2,\mathbb{C}) \simeq \mathrm{SO}(3,\mathbb{C})$. The finite subgroups of $\mathrm{SO}(3,\mathbb{C})$ are conjugated to a finite subgroup of $\mathrm{SO}(3,\mathbb{R})$, which is the group of rotations of the sphere. Following [Wey52, App. A], we have the following five cases:



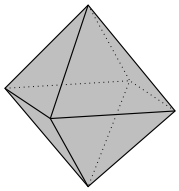
The group of rotations of a right pyramid with a regular n-gonal base. For $n = 3$ the pyramid is not a regular tetrahedron. For $n = 2$ the pyramid generates into an isosceles triangle not equilateral. This group is isomorphic to the group of planar rotations of the base, thus the cyclic group $\mathbf{C}_n$ of order n.



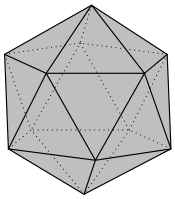
The group of rotations of a right prism with a regular n-gonal base. For $n = 4$ the prism is not a cube. For $n = 2$ it degenerates into a rectangle not a square. This group is isomorphic to the group of planar symmetries of the base, thus the dihedral group $\mathbf{D}_n$ of order $2n$.



The group of rotations of a regular tetrahedron. These rotations permute the 4 vertices of the tetrahedron. This yields an isomorphism with the alternating group $\mathbf{A}_4$.



The group of rotations of a regular octahedron. These rotations permute the 4 axes through the midpoints of opposite faces. This yields an isomorphism with the symmetric group $\mathbf{S}_4$.



The group of rotations of a regular icosahedron. There is an only way to color its faces with 5 colors such that the faces concurrent at a vertex have different colors. The midpoints of the faces of the same color define a regular tetrahedron. The rotations of the group permute these 5 tetrahedra. This yields an isomorphism with the alternating group $\mathbf{A}_5$.

As $\mathrm{SL}(2,\mathbb{C}) \to \mathrm{PGL}(2,\mathbb{C})$ covers twice, the finite subgroup $H^{\mathrm{SL}(2,\mathbb{C})}$ has double the order of H. For instance, $\mathbf{D}_2^{\mathrm{SL}(2,\mathbb{C})}$ has order 8. This group is also called the quaternion group because it corresponds to the multiplicative group of $\{\pm 1, \pm \mathtt{i}, \pm \mathtt{j}, \pm \mathtt{k}\}$ of quaternions. This corresponds to the usual identification of quaternions with 2×2 complex matrices:

$$1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathtt{i} = \begin{pmatrix} \mathrm{i} & 0 \\ 0 & -\mathrm{i} \end{pmatrix}, \quad \mathtt{j} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad \mathtt{k} = \begin{pmatrix} 0 & \mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix}.$$

3.5 Broad and eurymeric subgroups of $\mathrm{GL}(2,\mathbb{C})$

Any eurymeric subgroup of $\mathrm{GL}(2,\mathbb{C})$ has a Lie algebra that contains $\mathbb{C} \cdot \mathtt{I}$, so the group contains $\mathbb{C}^*\mathtt{I}$, thus it is the counterimage of its image in $\mathrm{PGL}(2,\mathbb{C})$. So the eurymeric closure, and thus the broad hull, of a subgroup G of $\mathrm{GL}(2,\mathbb{C})$ is the corresponding closure of $\mathbb{C}^* \cdot G$. Any projectively saturated group is the projective saturation of a subgroup of $\mathrm{SL}(2,\mathbb{C})$. Thus it is enough to compute the projective saturation of the elements of the list of subgroups of $\mathrm{SL}(2,\mathbb{C})$, which are known up to conjugation.

- The projective saturation H of a finite group G is a collection of components $\mathbb{C}^*\mathtt{M}$ for each $\mathtt{M} \in G$. The identity component is $H^\circ = \mathbb{C}^*\mathtt{I}$ and the Lie algebra $\mathfrak{h} = \mathbb{C}\mathtt{I}$. As H° is broad, H is the eurymeric closure of G.
- The projective saturation of the multiplicative group $\left\{ \begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^* \right\}$ is $H = \left\{ \begin{pmatrix} b & 0 \\ 0 & c \end{pmatrix} : b, c \in \mathbb{C}^* \right\} \simeq \mathbb{C}^* \times \mathbb{C}^*$, connected and with Lie algebra $\mathfrak{h} = \left\{ \begin{pmatrix} b & 0 \\ 0 & c \end{pmatrix} : b, c \in \mathbb{C} \right\} \simeq \mathbb{C}\mathtt{I} \oplus \mathbb{C}\mathtt{I}$. As H is broad, it is the broad hull and the eurymeric closure of the multiplicative group.
- The projective saturation of the additive group $\left\{ \begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix} : a \in \mathbb{C} \right\}$ is $H = \left\{ \begin{pmatrix} b & c \\ 0 & b \end{pmatrix} : b \in \mathbb{C}^*, c \in \mathbb{C} \right\}$, connected and with Lie algebra $\mathfrak{h} = \left\{ \begin{pmatrix} b & c \\ 0 & b \end{pmatrix} : b, c \in \mathbb{C} \right\}$. As H is broad, it is the broad hull and the eurymeric closure of the additive group.
- The projective saturation of the Borel group $\left\{ \begin{pmatrix} a & b \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^*, b \in \mathbb{C} \right\}$,

is $H = \left\{ \begin{pmatrix} c & d \\ 0 & e \end{pmatrix} : c, e \in \mathbb{C}^*, d \in \mathbb{C} \right\}$, connected and with Lie algebra $\mathfrak{h} = \left\{ \begin{pmatrix} c & d \\ 0 & e \end{pmatrix} : c, d, e \in \mathbb{C} \right\}$. As H is broad, it is the broad hull and the eurymeric closure of the Borel group.

- The projective saturation of $G = \left\{ \begin{pmatrix} a & b \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^*, a^k = 1, b \in \mathbb{C} \right\}$, with $k \in \mathbb{N}$, is $H = \left\{ \begin{pmatrix} ac & d \\ 0 & a^{-1}c \end{pmatrix} : a, c \in \mathbb{C}^*, a^k = 1, d \in \mathbb{C} \right\}$. The identity component is $H^\circ = \left\{ \begin{pmatrix} c & d \\ 0 & c \end{pmatrix} : c \in \mathbb{C}^*, d \in \mathbb{C} \right\}$ and the Lie algebra $\mathfrak{h} = \left\{ \begin{pmatrix} c & d \\ 0 & c \end{pmatrix} : c, d \in \mathbb{C} \right\}$. As H° is broad, H is the eurymeric closure of G. As $\mathbb{C}[G] = \left\{ \begin{pmatrix} e & f \\ 0 & g \end{pmatrix} : e, f, g \in \mathbb{C} \right\}$, the broad hull of G is the projective saturation of the Borel group.
- The projective saturation of the dihedral group $G = \left\{ \begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix}, \begin{pmatrix} 0 & -a \\ a^{-1} & 0 \end{pmatrix} : a \in \mathbb{C}^* \right\}$, is $H = \left\{ \begin{pmatrix} b & 0 \\ 0 & c \end{pmatrix}, \begin{pmatrix} 0 & b \\ c & 0 \end{pmatrix} : b, c \in \mathbb{C}^* \right\}$. The identity component is $H^\circ = \left\{ \begin{pmatrix} b & 0 \\ 0 & c \end{pmatrix} : b, c \in \mathbb{C}^* \right\}$ and the Lie algebra $\mathfrak{h} = \left\{ \begin{pmatrix} b & 0 \\ 0 & c \end{pmatrix} : b, c \in \mathbb{C} \right\}$. As H° is broad, H is the eurymeric closure of G. As $\mathbb{C}[H] = \mathfrak{gl}(2, \mathbb{C})$, the broad hull of H, and thus of G, is $\mathrm{GL}(2, \mathbb{C})$.
- The projective saturation of $\mathrm{SL}(2, \mathbb{C})$ is $\mathrm{GL}(2, \mathbb{C})$, which is broad, so $\mathrm{GL}(2, \mathbb{C})$ is the broad hull and eurymeric closure of $\mathrm{SL}(2, \mathbb{C})$.

The only remaining task is to compute the broad hull of the finite subgroups of $\mathrm{GL}(2, \mathbb{C})$.

- The cyclic group $G = \left\{ \begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix} : a \in \mathbb{C}^*, a^k = 1 \right\}$, for $k \geqslant 3$, generates $\mathbb{C}[G] = \left\{ \begin{pmatrix} b & 0 \\ 0 & c \end{pmatrix} : b, c \in \mathbb{C} \right\}$, the broad Lie algebra of $\left\{ \begin{pmatrix} b & 0 \\ 0 & c \end{pmatrix} : b, c \in \mathbb{C}^* \right\}$, which is the broad hull of G. For $k \in \{1, 2\}$, $\mathbb{C}[G] = \mathbb{C} \cdot \mathtt{I}$, so the broad hull of G is $\mathbb{C}^*\mathtt{I}$.
- The dihedral group $G = \left\{ \begin{pmatrix} a & 0 \\ 0 & a^{-1} \end{pmatrix}, \begin{pmatrix} 0 & -a \\ a^{-1} & 0 \end{pmatrix} : a \in \mathbb{C}^*, a^k = 1 \right\}$, for even $k \geqslant 4$, generates $\mathbb{C}[G] = \mathfrak{gl}(2, \mathbb{C})$, so the broad hull of G is $\mathrm{GL}(2, \mathbb{C})$. For $k =$

2, $\mathbb{C}[G] = \left\{ \begin{pmatrix} b & c \\ -c & b \end{pmatrix} : b, c \in \mathbb{C} \right\}$, so the broad hull of G is $\left\{ \begin{pmatrix} b & c \\ -c & b \end{pmatrix} : b, c \in \mathbb{C}, b^2 + c^2 \neq 0 \right\}$, which is a conjugate of $\left\{ \begin{pmatrix} d & 0 \\ 0 & e \end{pmatrix} : d, e \in \mathbb{C}^* \right\}$.

- Up to conjugation, the tetrahedral group is generated by two matrices $\begin{pmatrix} \xi & 0 \\ 0 & \xi^5 \end{pmatrix}$ for ξ a primitive sixth root of unity and $\varphi \begin{pmatrix} 1 & 1 \\ 2 & -1 \end{pmatrix}$ for certain $\varphi \neq 0$, according to [Kov86, Thm. 2].
- Up to conjugation, the octahedral group is generated by two matrices $\begin{pmatrix} \xi & 0 \\ 0 & \xi^7 \end{pmatrix}$ for ξ a primitive eighth root of unity and $\varphi \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ for certain $\varphi \neq 0$, according to [Kov86, Thm. 3].
- Up to conjugation, the icosahedral group is generated by two matrices $\begin{pmatrix} \xi & 0 \\ 0 & \xi^9 \end{pmatrix}$ for ξ a primitive tenth root of unity and $\begin{pmatrix} \varphi & \psi \\ \psi & -\varphi \end{pmatrix}$ for certain $\varphi \neq 0$ and $\psi \neq 0$, according to [Kov86, Thm. 4].
- In the three last cases, the group generates the algebra $\mathfrak{gl}(2, \mathbb{C})$, so its broad hull is $\mathrm{GL}(2, \mathbb{C})$.

3.6 Derksen–van der Hoeven linearized

In this section I will adapt Derksen–van der Hoeven algorithm in order to compute the eurymeric group generated by a finite number of elements. Let me first describe the eurymeric group generated by a single element $\mathtt{A} \in \mathrm{GL}$. Let λ_i be its eigenvalues. For any quotient λ_i/λ_j that is a root of unity, we consider its order. The least common multiple p of these orders and 1 will be called the *resonance order* of $\mathtt{A}$. I shall prove that the smallest eurymeric group G containing $\mathtt{A}$ has the Lie algebra $\mathbb{K}[\mathtt{A}^p]$ and $\{\mathtt{I}, \mathtt{A}, \ldots, \mathtt{A}^{p-1}\}$ are representatives of G/G°.

Lemma 60. *Let G be the eurymeric group generated by the single element $\mathtt{A} \in \mathrm{GL}$. If $\mathtt{A}^p \in G^\circ$, then $G^\circ = \mathbb{K}[\mathtt{A}^p] \cap \mathrm{GL}$. Moreover, $G = \mathtt{A}^{\mathbb{Z}} \cdot G^\circ$.*

Proof. As $H = \mathbb{K}[\mathtt{A}^p] \cap \mathrm{GL}$ is an irreducible variety and the broad group generated by $\mathtt{A}^p$, obviously $H < G^\circ$. It is easy to check that $\mathtt{A}^{\mathbb{Z}} \cdot H$ is a eurymeric group with H the identity component. As G contains $\mathtt{A}^{\mathbb{Z}}$, it contains $\mathtt{A}^{\mathbb{Z}} \cdot H$, thus $G = \mathtt{A}^{\mathbb{Z}} \cdot H$. □

Lemma 61. *If* $\mathtt{A} \in \mathrm{GL}$ *is unipotent, then* $\mathtt{A} \in \mathbb{K}[\mathtt{A}^p]$ *for any power* p.

Proof. Let G be the eurymeric group generated by $\mathtt{A}$. It is classical that the algebraic group generated by a unipotent element is connected, so $\mathtt{A}$ lies in G°. By Lemma 60, $G^\circ = \mathbb{K}[\mathtt{A}^p] \cap \mathrm{GL}$, so $\mathtt{A} \in \mathbb{K}[\mathtt{A}^p]$. □

Proposition 62. *If* G *is the eurymeric group generated by a single element* $\mathtt{A} \in \mathrm{GL}$ *and* p *is the resonance order of* $\mathtt{A}$*, then* $\mathtt{A}^p \in G^\circ$.

Proof. Assume that $\mathtt{A}$ is in Jordan form with diagonal $(\lambda_1, \lambda_2, \dots, \lambda_n)$. For each couple of indices i and j, let p_{ij} be the order of λ_i/λ_j as a root of unity or, if λ_i/λ_j is not a root of unity, $p_{ij} = 1$. By definition, p is the least common multiple of all these p_{ij}.

By Lemma 60, there exists a multiple r of p such that $G^\circ = \mathbb{K}[\mathtt{A}^r] \cap \mathrm{GL}$. Let q be the order of $\mathtt{A}$ modulo G°. Then $\mathtt{A}^q = P(\mathtt{A}^r)$ for certain $P \in \mathbb{K}[x]$. Consider a fixed couple (i, j). If λ_i/λ_j is a root of the unity, it is primitive of order p_{ij}, so $\lambda_i^{p_{ij}} = \lambda_j^{p_{ij}}$ and $\lambda_i^r = \lambda_j^r$. The diagonal of $\mathtt{A}^q$ is $(\lambda_1^q, \lambda_2^q, \dots, \lambda_n^q) = \bigl(P(\lambda_1^r), P(\lambda_2^r), \dots, P(\lambda_n^r)\bigr)$, thus $\lambda_i^q = \lambda_j^q$ and therefore p_{ij} divides q. Otherwise, if λ_i/λ_j is not a root of the unity, then $p_{ij} = 1$, so p_{ij} divides q as well.

As p_{ij} divides q for each couple (i, j), p divides q, so we can factorize $q = kp$ and, by Lemma 60, take $r = q$. Let $\mathtt{D}$ be the diagonalizable part and $\mathtt{U}$ the unipotent part of $\mathtt{A}^p$, then $\mathtt{D}^k$ is the diagonalizable part and $\mathtt{U}^k$ the unipotent part of $\mathtt{A}^q$, hence $\mathtt{D}^k$ and $\mathtt{U}^k$ belong to $\mathbb{K}[\mathtt{A}^q]$ and thus to G°. Obviously, $\mathtt{D} = \mathrm{diag}(\lambda_1^p, \lambda_2^p, \dots, \lambda_n^p)$ and $\mathtt{D}^k = \mathrm{diag}(\lambda_1^q, \lambda_2^q, \dots, \lambda_n^q)$. Consider the map $\varphi : \{\lambda_1^q, \lambda_2^q, \dots, \lambda_n^q\} \to \mathbb{K}$ given by $\varphi(\lambda_i^q) = \lambda_i^p$ for each i. Such a map exists because, if $\lambda_i^q = \lambda_j^q$ for certain i and j, then $\lambda_i^p = \lambda_j^p$. Let Q be the Lagrange interpolating polynomial for φ; then $\mathtt{D} = Q(\mathtt{D}^k)$ belongs to G°. From Lemma 61, $\mathtt{U}$ lies in $\mathbb{K}[\mathtt{U}^k]$ and therefore in $\mathbb{K}[G^\circ] \cap \mathrm{GL} = G^\circ$. As $\mathtt{A}^p = \mathtt{DU}$ belongs to G°, q divides p, thus $p = q$. □

I shall explain a modification of Derksen–van der Hoeven algorithm, introduced in §3.2, for computing the eurymeric group generated by a finite family $\mathcal{M} \subset \mathrm{GL}$ and a finite family algebraic groups, each one given as a family $\mathcal{F}_i$ of representatives of its components and its Lie algebra $\mathfrak{g}_i$, same as the output of van der Hoeven's version of Derksen–van der Hoeven algorithm. The algorithm works with a finite

family $\mathcal{F}$, initially the union of $\mathcal{M}$ and the $\mathcal{F}_i$, and a Lie algebra $\mathfrak{a}$, initially the algebra generated by the $\mathfrak{g}_i$, which are augmented by a loop until they stabilize, in which case the algorithm terminates. Each iteration of the loop performs the following steps.

1. For each $\mathtt{A} \in \mathcal{F}$ we augment $\mathfrak{a}$, as an algebra, with the Lie algebra of the eurymeric group generated by $\mathtt{A}$.
2. For each $\mathtt{A} \in \mathcal{F}$ we augment $\mathfrak{a}$, as an algebra, with $\mathtt{A}\mathfrak{a}\mathtt{A}^{-1}$.
3. For each $\mathtt{A} \in \mathcal{F}$ we check if $\mathtt{A}$ is equivalent to any element of $\mathcal{F}$ modulo the connected group corresponding to $\mathfrak{a}$, which is $\mathfrak{a} \cap \mathrm{GL}$; if it is, we eliminate $\mathtt{A}$ from $\mathcal{F}$.
4. For each ordered pair $\mathtt{A}, \mathtt{B} \in \mathcal{F}$ we check if $\mathtt{AB}$ is equivalent to any element of $\mathcal{F}$ modulo $\mathfrak{a} \cap \mathrm{GL}$; if it is not, we add $\mathtt{AB}$ to $\mathcal{F}$.

Contrary to the original Derksen–van der Hoeven algorithm, here the equivalence modulo the connected group corresponding to $\mathfrak{a}$ is easy to compute, since $\mathtt{B}, \mathtt{C} \in \mathcal{F}$ are in the same component if and only if $\mathtt{B}^{-1}\mathtt{C} \in \mathfrak{a}$, so the task is reduced to linear algebra.

Example 63. Let us compute the eurymeric group generated by $\mathtt{M} = \left(\begin{smallmatrix} 2 & 0 \\ 0 & \frac{1}{2} \end{smallmatrix}\right)$ in order to see the difference with Example 55. Initially $\mathcal{F} = \{\mathtt{M}\}$ and $\mathfrak{a} = \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$. First iteration, step 1: we augment $\mathfrak{a}$ with the Lie algebra of the eurymeric group generated by $\mathtt{M}$. The quotient $2/\frac{1}{2} = 4$ is not a root of unity so the Lie algebra is $\mathbb{K}\left[\left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)\right] = \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right) + \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$, so now $\mathfrak{a} = \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix}\right) + \mathbb{K}\left(\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right)$. Step 2: as $\mathtt{M}\mathfrak{a}\mathtt{M}^{-1} = \mathfrak{a}$, we keep $\mathfrak{a}$ invariant. Step 3: no duplicates. Step 4: is $\mathtt{M}^2$ a duplicate of $\mathtt{M}$? It is, thus we keep $\mathcal{F} = \{\mathtt{M}\}$. Second iteration, step 1: we augment $\mathtt{a}$ with the Lie algebra of the eurymeric group generated by $\mathtt{M}$, which is already contained, so we keep $\mathfrak{a}$ invariant. Step 2: as $\mathtt{M}\mathfrak{a}\mathtt{M}^{-1} = \mathfrak{a}$, we keep $\mathfrak{a}$ invariant. Step 3: same as in the previous iteration. Step 4: same as in the previous iteration. The algorithm ends with $\mathfrak{g} = \mathbb{K}\left(\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix}\right) + \mathbb{K}\left(\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ and G/G° represented by $\{\mathtt{M}\}$.

Theorem 64. *Performing the computations exactly, the modification of Derksen–van der Hoeven algorithm introduced above for computing the eurymeric group G generated by a finite family $\mathcal{M} \subset \mathrm{GL}$ and a finite family of algebraic groups, each one given as a family $\mathcal{F}_i$ of representatives of its components and its Lie algebra $\mathfrak{g}_i$, terminates with $\mathfrak{a}$ the Lie algebra of G and $\mathcal{F}$ representatives of G/G° as output.*

Proof. Assume that the algorithm does not terminate for certain particular data, so the loop iterates infinitely without stabilization. As $\mathfrak{a}$ cannot grow beyond

dimension n^2, $\mathfrak{a}$ eventually stabilizes, so $\mathcal{F}$ must grow indefinitely. At these stages of the execution, all the elements of $\mathcal{F}$ normalize $H = \mathfrak{a} \cap \mathrm{GL}$ and all of them have finite order modulo H and, by [vdH07a, Lem. 3], the group generated by $\mathcal{F}$ modulo H (within the normalizer of H) is finite, so $\mathcal{F}$ cannot grow indefinitely; at certain stage all the new products added to $\mathcal{F}$ are duplicates and thus discarded.

From the absurd in the previous paragraph we draw that the algorithm terminates for all data. At the end, the group generated by $\mathcal{F}$ and H is an algebraic group with H being the identity component, thus a eurymeric group. Each element added to $\mathcal{F}$ or $\mathfrak{a}$ must be there for any eurymeric group containing the data, so the resulting group is contained in any eurymeric group containing the data, hence this group is G, $G^\circ = H$ and the final value of $\mathcal{F}$ is a system of representatives of G/G°. □

Remark 65. This proof relies on [vdH07a, Lem. 3], communicated by J.-Y. Hée to J. van der Hoeven. It relies on the classic Burnside problem, which asks if a finitely generated group whose elements have finite order is necessarily finite. The answer to the Burnside problem is false in general, but true for algebraic groups, including quotients of algebraic groups, which are isomorphic to algebraic groups.

3.7 Other results

Proposition 66. *The eurymeric closure of an algebraic group G is*

$$H = \{\mathtt{AB} : \mathtt{A} \in G, \mathtt{B} \in \mathbb{K}[G^\circ] \cap \mathrm{GL}\},$$

with $H^\circ = \mathbb{K}[G^\circ] \cap \mathrm{GL}$.

Proof. Let $N = \mathbb{K}[G^\circ] \cap \mathrm{GL}$. As G normalizes G°, then it normalizes N, and thus $N \lhd H$. Observe that $H = GN$, so $H/N = (GN)/N \simeq G/(G \cap N)$ by the Second Isomorphism Theorem. As $G^\circ < G \cap N$ and G/G° is finite, then $G/(G \cap N)$ is finite, and so H/N. Observe that H is then union of a finite number of cosets modulo N, and each coset is an algebraic variety, so H is an algebraic group. Finally, $H^\circ = N$ since $N \lhd H$, H/N is finite and N is connected.

We have that N is a broad group and H a eurymeric group. For any eurymeric group G_+ containing G, $G_+^\circ = \mathbb{K}[G_+^\circ] \cap \mathrm{GL} \supset \mathbb{K}[G^\circ] \cap \mathrm{GL} = N$, so G_+ contains H. Therefore H is the smallest eurymeric group containing G. □

Corollary 67. *With the same notation, G° is diagonalizable/triangularizable/abelian/solvable if and only if H° is so. A vector subspace is invariant by G° if and only if it is invariant by H°.*

Proof. Solvability in a connected algebraic group is equivalent to triangularizability, according to Lie-Kolchin Theorem; see Theorem 30. □

Remark 68. In particular the result "G° is abelian if and only if H° is" is interesting for the application of Morales-Ramis and Morales-Ramis-Simó theorems, which say that the integrability of certain dynamical system implies that the Galois group of certain linearization has an abelian identity component, cf. [MRR01] and [MRRS07]. The usual application of the theorems is negative: if the identity component is not abelian, then the dynamical system is non-integrable. If we compute the eurymeric closure of the Galois group instead, we still have that, if its identity component is not abelian, then the dynamical system is non-integrable. This result may make the eurymeric closure attractive outside the scope of this thesis.

Another consequence of Proposition 66 is that the broad hull and the eurymeric closure of an algebraic group commute with the extension of the base field. First of all, in order to speak of the extension of an algebraic group by the extension of the base field, we shall prove a result showing that the variety defined by the same equations is an algebraic group.

Theorem 69. *Let F be a field extension of $\mathbb{K}$, G be an algebraic subgroup of $\mathrm{GL}(n,\mathbb{K})$ and H the subvariety of $\mathrm{GL}(n,F)$ given by the same equations as G. Then H is an algebraic subgroup of $\mathrm{GL}(n,F)$.*

Proof. Let $\mathtt{X}$ and $\mathtt{Y}$ be $n \times n$ matrices of indeterminates. Let I be the ideal of G in the ring $\mathbb{K}[\underline{\mathtt{X}}]$, J_1 be the ideal generated by $\{f(\underline{\mathtt{X}}), f(\underline{\mathtt{Y}}) : f \in I\}$ in the ring $\mathbb{K}[\underline{\mathtt{X}}, \underline{\mathtt{Y}}]$, and J_2 be the ideal generated by $\{f(\underline{\mathtt{X} \cdot \mathtt{Y}}) : f \in I\}$ in $\mathbb{K}[\underline{\mathtt{X}}, \underline{\mathtt{Y}}]$. Recall that a matrix underlined represents the list of its entries row after row. As J_1 is the ideal of $G \times G$, it is a radical ideal. The closure of G under multiplication is equivalent to the contention $J_2 \subseteq J_1$, and this contention is kept after the extension from $\mathbb{K}$ to F.

If $\mathtt{A} \in H$, the chain $H \supset \mathtt{A} \cdot H \supset \cdots \supset \mathtt{A}^m \cdot H \supset \cdots$ is stationary because $F[\underline{\mathtt{X}}]$ is Noetherian, so $\mathtt{A}^r \cdot H = \mathtt{A}^{r+1} \cdot H$ for certain r, hence $H = \mathtt{A} \cdot H$, therefore $\mathtt{I} = \mathtt{AB}$ for certain $\mathtt{B} \in H$, and thus $\mathtt{B} = \mathtt{A}^{-1}$, because $\mathtt{I} \in G \subset H$. □

Theorem 70. *Let F be a field extension of $\mathbb{K}$, G be an algebraic subgroup of $\mathrm{GL}(n,\mathbb{K})$ and H be its broad hull. Let G_F be the extension of G to the base field F. The broad hull H_F of G_F is the extension of H.*

Proof. As G and G_F are groups, $\mathbb{K}[G]$ is the $\mathbb{K}$-linear span of G and $F[G_F]$ is the F-linear span of G_F. Let S be the linear equations of $\mathbb{K}[G]$ in $\mathbb{K}^{n\times n}$. As S can be added to the equations of G, G_F satisfies S, and thus its span $F[G_F]$ also satisfies S, hence $F[G_F]$ has at most the same dimension over F as $\mathbb{K}[G]$ over $\mathbb{K}$. The dimensions are equal because $F[G_F]$ contains a basis of $\mathbb{K}[G]$, which is also a basis over F. Finally $H = \mathbb{K}[G] \cap \mathrm{GL}(n,\mathbb{K}) = F[G_F] \cap \mathrm{GL}(n,\mathbb{K}) = H_F \cap \mathrm{GL}(n,\mathbb{K})$. □

Theorem 71. *Let F be a field extension of $\mathbb{K}$, G be an algebraic subgroup of $\mathrm{GL}(n,\mathbb{K})$ and H be its eurymeric closure. Let G_F be the extension of G to the base field F. The eurymeric closure H_F of G_F is the extension of H.*

Proof. According to Lemma 72, G_F° is the extension of G° to the base field F. By Proposition 66, H° is the broad hull of G° and H_F° is the broad hull of G_F°. According to Theorem 70, these broad hulls are given by the same equations. By Proposition 66, $H = G \cdot H^\circ$ and $H_F = G_F \cdot H_F^\circ$. Each component of H is the image of the linear space H° by a matrix $\mathtt{M} \in G$, and each component of H_F is the image of the linear space H_F° by a matrix $\mathtt{M} \in G_F$. According to Lemma 72, each component of G_F has elements in G, so we can choose representatives of the components of G_F in G. As H° and H_F° are given by the same linear equations, the groups H and H_F are also given by the same equations. □

The proof of this theorem requires the following lemma to be completed.

Lemma 72. *Let F be a field extension of $\mathbb{K}$, G be an algebraic subgroup of $\mathrm{GL}(n,\mathbb{K})$ and G_F be the extension of G to the base field F. The components of G and G_F are in bijection, each pair given by the same equations. In particular, G° corresponds to G_F°.*

Proof. We shall prove the corresponding result in Commutative Algebra. Let $\mathtt{X}$ be an $n \times n$ matrix of indeterminates, and $\mathfrak{q}$ be the ideal of G in the ring $\mathbb{K}[\underline{\mathtt{X}}]$. As $\mathfrak{q}$ is a radical ideal, it admits the unique decomposition $\mathfrak{q} = \mathfrak{p}_1 \cap \mathfrak{p}_2 \cap \cdots \cap \mathfrak{p}_s$ as irredundant intersection of prime ideals. For each i, $F[\underline{\mathtt{X}}]/\mathfrak{p}_i F[\underline{\mathtt{X}}]$ is isomorphic to $\mathbb{K}[\underline{\mathtt{X}}]/\mathfrak{p}_i\mathbb{K}[\underline{\mathtt{X}}] \otimes_{\mathbb{K}} F$, which is a tensor product of two integral domains over an algebraically closed field and thus an integral domain, therefore $\mathfrak{p}_i F[\underline{\mathtt{X}}]$ is prime. So, we have the decomposition $\mathfrak{q}F[\underline{\mathtt{X}}] = \mathfrak{p}_1 F[\underline{\mathtt{X}}] \cap \mathfrak{p}_2 F[\underline{\mathtt{X}}] \cap \cdots \cap \mathfrak{p}_s F[\underline{\mathtt{X}}]$. □

3.8 Resonance truncated order

In §3.6 we have explained and proved a modification of Derksen–van der Hoeven algorithm for computing the eurymeric group generated by finite data, but the proof was done under the assumption of exact computations. We know that with complex effective numbers we may have two sources of error. One of these sources is linear algebra, and will be addressed in §3.8.2. The other source of error is the computation of the rank over $\mathbb{Q}$, explained in §2.4. According to §3.6, the algorithm depends on the computation of the order as root of unity of the quotients of eigenvalues.

Example 73. Let us consider the following toy example: the eurymeric group generated by $\mathtt{A} = \mathrm{diag}(1, \lambda)$. The quotients of eigenvalues are λ and $1/\lambda$, so we may consider only λ. Let us follow the cases described in §2.4. In the cases 1(a)i and 2a the computation of the resonance order is correct. In the case 1(a)ii, the resonance order is 1 but we get q, which is a multiple of the correct one and yields the same Lie algebra according to Lemma 60. In the cases 1(a)iii and 2b, the resonance order is $p > \mathtt{Q}$, for $\mathtt{Q}$ the bound described in §2.4, but we get $q \leqslant \mathtt{Q}$, so we compute $\mathbb{C}[\mathtt{A}^q] = \mathbb{C}\mathtt{I}_1 \oplus \mathbb{C}\mathtt{I}_1$ instead of $\mathbb{C}[\mathtt{A}^p] = \mathbb{C}\mathtt{I}_2$.

Recall the definition of *truncated order* from §2.4. For $P \in \mathbb{N}$, the *P-truncated order* of $\lambda \in \mathbb{K}^*$ is its order p as a root of unity if $p \leqslant P$ and the P-truncated order is 1 if λ is a root of unity of order greater than P or λ is not a root of unity. The test of §2.4 computes the $\mathtt{Q}$-truncated order with the exception of the case 1(a)iii, when the $\mathtt{Q}$-truncated order is 1 but we compute $q \leqslant \mathtt{Q}$, which only occurs at low precision. We may define the *resonance P-truncated order* as the least common multiple of the P-truncated orders of all the quotients of eigenvalues. The resonance $\mathtt{Q}$-truncated order is computed exactly or, at low precision, a multiple of its exact value.

If we use the resonance $\mathtt{Q}$-truncated order in the algorithm of §3.6, by Lemma 60, it does not matter that we compute a multiple of it, but the result may be augmented due to the roots of unity of order greater than $\mathtt{Q}$. According to Corollary 67, the identity component of the algebraic group and the identity component of its eurymeric closure leave invariant the same lines. The problem with the augmentation due to the resonance truncated order is that we may lose invariant lines. For instance, in Example 73, when we compute $\mathbb{C}\mathtt{I}_1 \oplus \mathbb{C}\mathtt{I}_1$ instead of $\mathbb{C}\mathtt{I}_2$, an infinity of invariant lines is lost, keeping only the two axes. I shall prove that we keep the interesting invariant lines, those giving Singerian solutions.

I define a *Singerian line* with respect to an algebraic group G as an invariant

line by G° with $I(n)$ images at most by G. Singerian lines correspond to Singerian solutions, defined by Theorem 43. I shall prove that the augmentation due to the truncated order keeps the Singerian lines.

Lemma 74. *If p is the resonance P-truncated order of $\mathtt{A} \in \mathrm{GL}$ or a multiple thereof and ℓ a line invariant by $\mathtt{A}^{pq}$ for certain q, then either ℓ is invariant by $\mathtt{A}^p$ or the length of the orbit of ℓ by $\mathtt{A}^{\mathbb{Z}}$ is greater than P.*

Proof. Let $\mathbf{v}$ be a director of ℓ; $\mathbf{v}$ is an eigenvector of $\mathtt{A}^{pq}$. There exist an eigenvalue λ of $\mathtt{A}$ such that λ^{pq} is the eigenvalue corresponding to $\mathbf{v}$. There may be several options for λ, of the form $\lambda\xi$ with ξ a root of unity; let us list these options as $\lambda, \lambda\xi_1, \lambda\xi_2, \dots, \lambda\xi_r$. The eigenspace of $\mathtt{A}^{pq}$ corresponding to λ^{pq} is spanned by the eigenspaces of $\mathtt{A}$ corresponding to $\lambda, \lambda\xi_1, \lambda\xi_2, \dots, \lambda\xi_r$; decompose $\mathbf{v} = \mathbf{v}_0 + \mathbf{v}_1 + \dots + \mathbf{v}_r$ for this direct sum of eigenspaces. If $\mathbf{v} = \mathbf{v}_i$ for certain i, then $\mathbf{v}$ is an eigenvector of $\mathtt{A}$ and thus of $\mathtt{A}^p$.

Assume that there are $i \neq j$ such that $\mathbf{v}_i$ and $\mathbf{v}_j$ are not null and that (assuming $\xi_0 = 1$) ξ_i/ξ_j is a root of unity of order greater than P. Without loss of generality, we take $i = 0$ and $j = 1$. Then $\mathtt{A}^m\mathbf{v} = \lambda^m\mathbf{v}_0 + \lambda^m\xi_1^m\mathbf{v}_1 + \lambda^m\xi_2^m\mathbf{v}_2 + \dots + \lambda^m\xi_r^m\mathbf{v}_r$ for any m, so $\mathtt{A}^m\ell$ is generated by $\mathbf{v}_0 + \xi_1^m\mathbf{v}_1 + \xi_2^m\mathbf{v}_2 + \dots + \xi_r^m\mathbf{v}_r$, hence $\mathtt{A}^m\ell = \ell$ implies $\xi_1^m = 1$ and thus $m > P$, so ℓ has more than P images by $\mathtt{A}^{\mathbb{Z}}$.

Assume now that for any $\mathbf{v}_i$ and $\mathbf{v}_j$ not null we have that ξ_i/ξ_j is a root of unity of order P at most, hence $(\xi_i/\xi_j)^p = 1$ and thus $\xi_i^p = \xi_j^p$. Without loss of generality we arrange the non-null components $\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_s$ and the null components $\mathbf{v}_{s+1}, \mathbf{v}_{s+2}, \dots, \mathbf{v}_r$. Then $\mathtt{A}^p\ell$ is generated by $\xi_0^p\mathbf{v}_0 + \xi_1^p\mathbf{v}_1 + \dots + \xi_r^p\mathbf{v}_s$. As $\xi_0^p = \xi_1^p = \dots = \xi_s^p$, $\mathtt{A}^p\ell$ is generated by $\mathbf{v}_0 + \mathbf{v}_1 + \dots + \mathbf{v}_s = \mathbf{v}$, thus $\mathtt{A}^p\ell = \ell$. □

Proposition 75. *If H is a eurymeric group, p is the resonance $I(n)$-truncated order of $\mathtt{A} \in H$ or a multiple thereof, G is the eurymeric group generated by H and $\mathbb{K}[\mathtt{A}^p] \cap \mathrm{GL}$, and ℓ is a Singerian line of H, then ℓ is a Singerian line of G.*

Proof. Let $\mathfrak{h}$ be the Lie algebra of H. According to the algorithm of §3.6, we start with $\mathcal{F}$ a system of representatives of H/H° and $\mathfrak{a} = \mathbb{K}[\mathfrak{h}, \mathtt{A}^p]$. The loop refines $\mathcal{F}$ removing duplicates and augments $\mathfrak{a}$ by $\mathtt{B}\mathtt{A}^p\mathtt{B}^{-1}$ for $\mathtt{B} \in \mathcal{F}$, so finally $\mathfrak{a} = \mathbb{K}[\mathfrak{h}, \mathtt{B}_1\mathtt{A}^p\mathtt{B}_1^{-1}, \mathtt{B}_2\mathtt{A}^p\mathtt{B}_2^{-1}, \dots, \mathtt{B}_r\mathtt{A}^p\mathtt{B}_r^{-1}]$ with $\mathtt{B}_1, \mathtt{B}_2, \dots, \mathtt{B}_r \in H$, and this is the value of the Lie algebra $\mathfrak{g}$ of G.

Let $\ell_1, \ell_2, \dots, \ell_m$ be the images of ℓ by H. By definition $m \leqslant I(n)$ and ℓ is invariant by $\mathfrak{h}$. There exists q such that $\mathtt{A}^{pq} \in H^\circ$, so ℓ in invariant by $\mathtt{A}^{pq}$. According to Lemma 74, either ℓ is invariant by $\mathtt{A}^p$ or it has more than $I(n)$

images by $\mathtt{A}^{\mathbb{Z}}$. As ℓ is Singerian of H, ℓ is invariant by $\mathtt{A}^p$. The same argument proves that any ℓ_i is invariant by $\mathtt{A}^p$.

For $\mathtt{B} \in H$, there exists i such that $\mathtt{B}\ell_i = \ell$, so $\mathtt{B}\mathtt{A}^p\mathtt{B}^{-1}\ell = \mathtt{B}\mathtt{A}^p\ell_i = \mathtt{B}\ell_i = \ell$. Hence ℓ is invariant by $\mathfrak{g}$. As $\mathcal{F}$ only decreases, the images of ℓ cannot grow, so ℓ is a Singerian line of G. □

3.8.1 The effect of resonance truncated order

Although Theorem 71 is no longer applicable when the eurymeric closure is approximately computed with resonance truncated order, Lemma 72 warrants that, if the algebraic group is defined over an algebraically closed subfield k of $\mathbb{K}$, each component has a representative defined over k. So, Theorem 71 is valid for the approximately computed eurymeric closure presented in this section, provided that the component of $\mathtt{A}$ has a representative defined over k with the same resonance truncated order as $\mathtt{A}$. This fact depends on a result of Algebraic Geometry.

Lemma 76. *Let K/k be an extension of algebraically closed fields. Let X and Y be algebraic varieties in K^r defined over k. If X has points (over K) out of Y, then it has points over k out of Y.*

Proof. Let I be the ideal of X and J the ideal of Y, both radical ideals defined over k. If X has no point over k out of Y, then J is contained in I. This contention of ideals is kept by the extension of the base field from k to K, so X would be contained in Y. □

Proposition 77. *Let K/k be an extension of algebraically closed fields. Let H be a linear subspace of $\mathfrak{gl}(n, K)$ defined over k. If $\mathtt{A} \in H \cap \mathrm{GL}(n, K)$ has resonance P-truncated order p, then there is $\mathtt{A}_0 \in H \cap \mathrm{GL}(n, k)$ with resonance P-truncated order p.*

Proof. Let $\{\mathtt{M}_1, \mathtt{M}_2, \ldots, \mathtt{M}_r\} \subset \mathfrak{gl}(n, k)$ be a basis of H. The characteristic polynomial $P(x, t_1, t_2, \ldots, t_r) = \det(t_1\mathtt{M}_1 + t_2\mathtt{M}_2 + \cdots + t_r\mathtt{M}_r - x\mathtt{I}_n)$ of $t_1\mathtt{M}_1 + t_2\mathtt{M}_2 + \cdots + t_r\mathtt{M}_r$ belongs to $k[x, t_1, t_2, \ldots, t_r]$. The resultant $R(y, t_1, t_2, \ldots, t_r)$ of $P(xy, t_1, t_2, \ldots, t_r)$ and $P(x, t_1, t_2, \ldots, t_r)$ w.r.t. x belongs to $k[y, t_1, t_2, \ldots, t_r]$. For each $\xi \in k$, $R(\xi, t_1, t_2, \ldots, t_r)$ defines an algebraic variety X_ξ in k^r, whose points are precisely the $(\alpha_1, \alpha_2, \ldots, \alpha_r)$ such that $\alpha_1\mathtt{M}_1 + \alpha_2\mathtt{M}_2 + \cdots + \alpha_r\mathtt{M}_r$ has an eigenvalue λ and $\xi\lambda$ is another eigenvalue thereof.

Let Ξ_P be the set of the roots of unity of order up to P. Let Q be the set of the quotients of eigenvalues of $\mathtt{A}$. The resonance P-truncated order p of $\mathtt{A}$ is the least common multiple of the order of the roots of unity in $\Xi_P \cap Q$. Let X be the intersection of the X_ξ for $\xi \in \Xi_P \cap Q$. Let Y be the union of $\mathfrak{gl}(n,k) \setminus \mathrm{GL}(n,k)$ (which is the variety of $\det = 0$) with the X_ξ for $\xi \in \Xi_P \setminus Q$. The extension of X to K contains $\mathtt{A}$, and $\mathtt{A}$ is out of the extension of Y. According to Lemma 76, there is $(\beta_1, \beta_2, \ldots, \beta_r) \in X \setminus Y$. Let $\mathtt{A}_0 = \beta_1 \mathtt{M}_1 + \beta_2 \mathtt{M}_2 + \cdots + \beta_r \mathtt{M}_r$. By construction, $\mathtt{A}_0 \in \mathrm{GL}(n,k)$. Let Q_0 be the set of the quotients of eigenvalues of $\mathtt{A}_0$; it must contain $\Xi_P \cap Q$ and must exclude $\Xi_P \setminus Q$, so $\Xi_P \cap Q_0 = \Xi_P \cap Q$ and thus the resonance P-truncated order of $\mathtt{A}_0$ is p. □

3.8.2 The effect of numerical linear algebra

The effect of numerical linear algebra on the computation of the eurymeric closure is simpler than the effect of resonance truncated order. A possible error is to undercompute the Lie algebra. Another possible error is to deem duplicates two elements of components too close. In both cases the error is an undercomputation of the group, and smaller groups may gain invariant lines but never lose them. Anyway, at precision fine enough the computations are exact.

Another effect of numerical linear algebra on the computation of the eurymeric closure is that the termination of the algorithm is no longer warranted. J. van der Hoeven claims in [vdH07a, §4.5] that the termination of his version of Derksen–van de Hoeven algorithm relies on [vdH07a, Lem. 3][7] without an explicit consideration of the effect of the approximated zero-test. Even if the algorithm terminated for any input and precision, there are good reasons to truncate the algorithm. For instance, the output may contain a large cyclic group in a case of no non-null common eigenvector of the Lie algebra and, hence, there are not non-zero Liouvillian solutions. For such a truncation we use a general[8] global parameter G. The truncated algorithm would stop, if it has not ended before, after G iterations. This way, the algorithm terminates for any input, tolerance and G, and it gives a correct output if G is high enough and tol is small enough. Notice that the tolerance depends on G. Moreover, for any G and tol, the output is either correct or undercomputed.

HERE ENDETH THE THIRDE CHAPTER ✠

[7] See Remark 65.
[8] See §2.5 for an introduction to general global parameters.

Chapter 4

The algorithm

In this chapter I shall explain and prove the algorithm for computing a nonzero Liouvillian solution of differential equation or system. or certifying that the system has no nonzero Liouvillian solution. This chapter also deals with the reconstruction of symbolic objects from numerical ones, which is necessary in order to give a symbolic correct output.

A Singerian solution y of a scalar equation may be given by the minimal polynomial of y'/y, where y'/y represents the line $\mathbb{K}y$, invariant by the identity component of the Galois group, and the roots of this minimal polynomial represent the orbit by the Galois group of this line. This is enough for scalar equations, but for expressing Singerian solutions of a system $\mathbf{y}' = \mathtt{A}\mathbf{y}$ we need *Darboux polynomials*, to be defined. Each solution of the system has an associated linear homogenous Darboux polynomial, and so each line of solutions. The product of the Darboux polynomials of the orbit of a Singerian line is another Darboux polynomial, which represents the orbit, and it has coefficients in the differential field of coefficients of the equation. If we construct this Darboux polynomial from a numerical Singerian solution, we get a numerical Darboux polynomial. In this chapter I shall explain the techniques for reconstructing the coefficients of the Darboux polynomial, but the reconstruction is correct only for precision good enough, so we must check if the reconstructed polynomial is Darboux and if it splits in linear factors. The latter is done with Brill equations. The factors of a Darboux polynomial are also Darboux, and in our case they represents lines whose director vectors are (alleged) Singerian solutions, so checking if the reconstructed polynomial is Darboux is equivalent to checking if this alleged solutions are true solutions.

The coefficients of the reconstructed Darboux polynomials should be rational

functions. We reconstruct them using Padé approximation. Then we reconstruct the numerical coefficients of these rational functions, which should lie in the algebraically closed field generated by the symbolic coefficients of the equation, according to §4.3, provided that the original solution was well chosen.

4.1 Darboux polynomials

According to Example 7, if K is a differential field where $\mathtt{A}$ is defined, the system is equivalent to $\nabla_{-\mathtt{A}}\mathbf{y} = \mathbf{0}$ in a differential module $(K^n, \nabla_{-\mathtt{A}})$ such that the image of the standard basis of K^n by $\nabla_{-\mathtt{A}}$ is $-\mathtt{A}$. Any connection in K^n gives rise to a derivation in the symmetric algebra of K^n in a unique way. The symmetric algebra of K^n is identified with $K[X_1, X_2, \ldots, X_n]$ where $X_1, X_2, \ldots, X_n$ represent the standard basis of K^n. So we have a derivation on $K[X_1, X_2, \ldots, X_n]$ that, restricted to the homogeneous linear polynomials, is the connection $\nabla_{-\mathtt{A}}$ via the identification of $X_1, X_2, \ldots, X_n$ with the standard basis of K^n. For instance, $\mathbf{y} = (y_1, y_2, \ldots, y_n)^\intercal$ is identified with the polynomial $H_{\mathbf{y}} = y_1X_1 + y_2X_2 + \cdots + y_nX_n$. With this notation, $H'_{\mathbf{y}} = H_{\nabla_{-\mathtt{A}}\mathbf{y}}$ and we have the following equivalence.

Proposition 78. *With the notation of this paragraph, the following statements are equivalent:*

- $\mathbf{y}$ *is a horizontal vector, i.e.,* $\nabla_{-\mathtt{A}}\mathbf{y} = \mathbf{0}$*;*
- $\mathbf{y}$ *is a solution of the system, i.e.,* $\mathbf{y}' = \mathtt{A}\mathbf{y}$*;*
- $H_{\mathbf{y}}$ *is a constant, i.e.,* $H'_{\mathbf{y}} = 0$*.*

The polynomials whose derivative is zero are called in this context *first integrals*, which are a particular case of Darboux polynomials. We say that P is a *Darboux polynomial* if P divides P'. The idea of using Darboux polynomials is due to J.-A. Weil, and his thesis [Wei95] is the reference for Darboux polynomials. The product of Darboux polynomials is another Darboux polynomial. Moreover, the irreducible factors of a Darboux polynomial are also Darboux polynomials. A proof of these properties is found in [Wei95, Lem. 12], but it is simple and I shall give it.

Proposition 79. *The product of Darboux polynomials is another Darboux polynomial.*

Proof. If P and Q are Darboux polynomials, there exist polynomials A and B such that $P' = AP$ and $Q' = BQ$; then $(PQ)' = APQ + PBQ$ is a multiple of PQ and thus PQ is a Darboux polynomial. □

Proposition 80. *The irreducible factors of a Darboux polynomial are also Darboux polynomials.*

Proof. If P is a Darboux polynomial and Q an irreducible factor thereof, there exist $k \in \mathbb{N}$ and polynomials A and R such that $P' = AP$, $P = Q^k R$ and Q does not divide R; then $P' = kQ^{k-1}Q'R + Q^k R'$ and also $P' = AQ^k R$. Eliminating Q^{k-1}, we have $kQ'R + QR' = AQR$, so Q divides $kQ'R$. As Q does not divide R, it must divide Q', thus Q is a Darboux polynomial. □

Let $\mathbf{y} = (y_1, y_2, \ldots, y_n) \in F^n$ be a non-zero Singerian solution of a linear system of differential equations. According to Theorem 43, for each i and j with $y_i \neq 0$, $y_i'/y_i \in F_0$ and $y_j/y_i \in F_0$ for an algebraic extension F_0/K of degree $I(n)$ at most. Without loss of generality, assume that $y_1 \neq 0$. The line $\mathbb{K}\mathbf{y}$ is represented by the Darboux polynomial $P_{\mathbb{K}\mathbf{y}} = H_{\mathbf{y}}/y_1$, which lies in $F_0[X_1, X_2, \ldots, X_n]$. If $\{\ell_1, \ell_2, \ldots, \ell_r\}$ is the orbit of $\mathbb{K}\mathbf{y}$ by the differential Galois group, this orbit is represented by the Darboux polynomial $P = P_{\ell_1} P_{\ell_2} \cdots P_{\ell_r}$, which contains the term X_1^r. As P is invariant by the differential Galois group, $P \in K[X_1, X_2, \ldots, X_n]$. This Darboux polynomial representing the orbit of Singerian solutions allows to get the minimal polynomial (or a power thereof) of each y_i/y_1, substituting X for X_1, -1 for X_i and 0 for the remaining X_j.

The annihilating polynomial obtained this way is a power of the minimal polynomial for the following reason. For being an annihilator, it has all the algebraic conjugates as roots. For the invariance under the differential Galois group, all the roots are algebraic conjugates. For the orbit-stabilizer theorem, all the roots have the same multiplicity. The following example shows that this annihilator can be a proper power of the minimal polynomial. In this case, computing symbolically, we can recover Q from $R = Q^r$ as $R/\gcd(R, R')$.

Example 81. Let us consider the system

$$\begin{cases} u' &= 0, \\ v' &= \dfrac{v}{2x}, \\ w' &= \dfrac{w}{3x}. \end{cases}$$

Let f be a primitive sixth root of x. A fundamental system of solutions of the system is $(1, 0, 0)$, $(0, f^3, 0)$ and $(0, 0, f^2)$. The Picard-Vessiot extension $\mathbb{C}(x, f)/\mathbb{C}(x)$

has a finite Galois group isomorphic to the group of sixth roots of unity. For each sixth root of the unity ξ, we have the Galoisian automorphism mapping f to ξf. The orbit of the solution $(y_1, y_2, y_3) = (1, f^3, f^2)$ is

$$\{(1, f^3, f^2), (1, -f^3, \omega f^2), (1, f^3, \omega^2 f^2), (1, -f^3, f^2), (1, f^3, \omega f^2), (1, -f^3, \omega^2 f^2)\},$$

with ω a primitive cubic root of the unity. The corresponding Darboux polynomial is

$$X_1^6 + x^2 X_3^6 + 2x X_1^3 X_3^3 - 3x X_1^4 X_2^2 + 3x^2 X_1^2 X_2^4 - x^3 X_2^6 + 6x^2 X_1 X_2^2 X_3^3$$

which gives the polynomials $(X^2 - x)^3$ and $(X^3 - x)^2$ for $y_2/y_1 = f^3$ and $y_3/y_1 = f^2$ respectively.

The annihilating polynomial of y_1'/y_1 is gotten as certain y_i/y_1 in the case of a companion system. In the general case, if we have the equation $y_1' = a_{11}y_1 + a_{12}y_2 + \cdots + a_{1n}y_n$, we get

$$y_1'/y_1 = a_{11} + a_{12}y_2/y_1 + a_{13}y_3/y_1 + \cdots + a_{1n}y_n/y_1,$$

which yields a symbolic expression for y_1'/y_1 from which we can compute symbolically an annihilating polynomial for y_1'/y_1, but not necessarily the minimal polynomial.

Example 82. Let us consider the system

$$\begin{cases} u' &= \dfrac{u}{x} &+ &xv, \\ v' &= u &+ &\dfrac{v}{2x}. \end{cases}$$

Let f be a square root of x, and g the exponential of an integral of f; symbolically $f = \sqrt{x}$ and $g = \exp \int \sqrt{x}$, but we will only use the relations $f^2 = x$ and $g' = fg$. A fundamental system of solutions of the system is (xg, fg) and $(x/g, -f/g)$. The Picard-Vessiot extension $\mathbb{C}(x, f, g)/\mathbb{C}(x)$ has a Galois group with two components. The identity component fixes f and acts like $\mathbb{C}^*$ on g. The other component consist of automorphisms σ_λ for $\lambda \in \mathbb{C}^*$ such that $\sigma_\lambda(f) = -f$ and $\sigma_\lambda(g) = \lambda/g$. There are two lines invariant by the action of the identity component, the lines generated by (xg, fg) and $(x/g, -f/g)$, and the other component swaps them. These vectors correspond to the first integrals $H_1 = xgX_1 + fgX_2$ and $H_2 = (x/g)X_1 - (f/g)X_2$, and the invariant lines correspond to the Darboux polynomials $P_1 = H_1/(xg) = X_1 + (f/x)X_2$ and $P_2 = H_2/(x/g) = X_1 - (f/x)X_2$. The Darboux polynomial of the orbit is $P = P_1P_2 = X_1^2 - (1/x)X_2^2$. Let us look for the minimal polynomials defining $u = xg$ and $v = fg$. The minimal polynomial of v/u is $P(X, -1) = X^2 - 1/x$, and we get u'/u from the first equation of the system, $u'/u = 1/x + x(v/u)$.

The set of the products of d homogeneous linear polynomials is an algebraic variety of the set of all the homogeneous polynomials of degree d, and the equations of this variety, called Brill equations, can be computed and have degree $d+1$; see [GKZ94, p. 140].

Example 83. Following [GKZ94, p. 139, Ex. 2.9], we shall compute the Brill equations for $d = 2$. A quadratic form splits in product of linear forms if and only if its matrix has rank 2 at most. This is equivalent to all 3×3 minors of the matrix vanishing, which are cubic equations. Thus, for $n = 3$, we have a single Brill equation, the determinant. The quadratic form $aX_1^2 + bX_1X_2 + cX_1X_3 + dX_2^2 + eX_2X_3 + fX_3^2$ is associated to the matrix

$$\begin{pmatrix} a & \frac{1}{2}b & \frac{1}{2}c \\ \frac{1}{2}b & d & \frac{1}{2}e \\ \frac{1}{2}c & \frac{1}{2}e & f \end{pmatrix},$$

so the Brill equation is

$$\begin{vmatrix} a & \frac{1}{2}b & \frac{1}{2}c \\ \frac{1}{2}b & d & \frac{1}{2}e \\ \frac{1}{2}c & \frac{1}{2}e & f \end{vmatrix} = 0.$$

For $n = 4$, the quadratic form $aX_1^2 + bX_1X_2 + cX_1X_3 + dX_1X_4 + eX_2^2 + fX_2X_3 + gX_2X_4 + hX_3^2 + iX_3X_4 + jX_4^2$ is associated to the matrix

$$\begin{pmatrix} a & \frac{1}{2}b & \frac{1}{2}c & \frac{1}{2}d \\ \frac{1}{2}b & e & \frac{1}{2}f & \frac{1}{2}g \\ \frac{1}{2}c & \frac{1}{2}f & h & \frac{1}{2}i \\ \frac{1}{2}d & \frac{1}{2}g & \frac{1}{2}i & j \end{pmatrix},$$

so the Brill equation are the 16 choices of 3×3 minor, but the symmetry of the matrix makes many redundant. For instance,

$$\begin{vmatrix} a & \frac{1}{2}b & \frac{1}{2}c \\ \frac{1}{2}b & e & \frac{1}{2}f \\ \frac{1}{2}d & \frac{1}{2}g & \frac{1}{2}i \end{vmatrix} = \begin{vmatrix} a & \frac{1}{2}b & \frac{1}{2}d \\ \frac{1}{2}b & e & \frac{1}{2}g \\ \frac{1}{2}c & \frac{1}{2}f & \frac{1}{2}i \end{vmatrix}.$$

In general, there are $\frac{1}{2}\binom{n}{3}\left(\binom{n}{3}+1\right)$ irredundant Brill equations, which is $\mathcal{O}(n^6)$.

In the main algorithm of this thesis, we construct the Darboux polynomial corresponding to a candidate of orbit of Singerian solutions. As this polynomial has coefficients in $\mathbb{K}(x)$, we reconstruct symbolically the rational functions as explained in §4.2. A reconstructed Darboux polynomial is tested with Brill equations to check if it factors in linear forms. The idea of Brill equations is taken from [SU97]. Then we check if it is an actual Darboux polynomial. As the coefficient of X_1^r is 1, we may assume the factors have coefficient 1 in X_1. These factors are Darboux

polynomials of the form P_ℓ, so each of them represents a line of solutions. The images of each line by the differential Galois group must be other lines represented in the Darboux polynomial, so they are a Singerian orbit.

Remark 84. It is necessary to check both Brill equations and Darboux polynomial because $X_1^2 - X_2X_3$ is irreducible and a first integral for the derivation defined by $X_i' = \frac{1}{2}X_i$, which is the associated to the system $\mathbf{y}' = -\frac{1}{2}\mathbf{y}$.

The Darboux polynomial is as large as the symmetric power of the equation, but easier to compute, and it does not need the latter in order to check it. Moreover, the Darboux polynomial is only computed and checked when we have a candidate solution. In the case when no candidate solution is found, we have an early termination, contrary to the classic algorithms, which would be in their worst case and would have to compute high-order symmetric powers.

4.2 Reconstruction of rational functions

In order to reconstruct a rational function from its power series expansion, J. van der Hoeven proposes in [vdH07a, §3.4] using Padé approximation. See [BGM81, Chap. 1] for a reference on Padé approximation. For a power series f and degrees r and s, the *Padé approximant* is the rational function with numerator of degree r at most, denominator of degree s at most and contact with f of maximal order. There are different definitions of Padé approximants in the literature, which may lead to apparently contradictory properties. The weaker definition is attributed to Frobenius because of his use in [Fro1881]. The stronger definition is attributed to G. A. Baker Jr.; see [Bak75, §2.B] and [BGM81, §1.4]. I shall first introduce our requisites of Padé approximation and later discuss our use for reconstructing the rational functions that appears as coefficients of the Darboux polynomials described in §4.1.

4.2.1 Introduction to Padé approximation

For a power series f and degrees r and s, a *strong Padé approximant* is a rational function

$$g(x) = \frac{a_0 + a_1x + \cdots + a_rx^r}{b_0 + b_1x + \cdots + b_sx^s} \tag{4.1}$$

such that

$$g(0) = f(0),\ g'(0) = f'(0),\ \ldots,\ g^{(r+s)}(0) = f^{(r+s)}(0). \tag{4.2}$$

The expression (4.1) of g has $r+s+1$ degrees of freedom (in the well behaved case, we impose $b_0 = 1$ and the rest of the parameters are free) and the system (4.2) of $r+s+1$ equations is the most we can ask g to satisfy.

The naive way to solve (4.2) is developing the quotient $g(x) = \sum_{k=0}^{\infty} G_k x^k$, with each G_k in $\mathbb{Z}[a_0, a_1, \ldots, a_r, b_1, b_2, \ldots, b_s]$, and considering the system $S_{frs} = \{G_k = f_k\}_{k=0}^{r+s}$ for $f = \sum_{k=0}^{\infty} f_k x^k$, but this system is non-linear because $G_1 = a_1 - a_0 b_1$. Moreover, the system S_{frs} is not compatible for all f, r and s. For instance, $S_{1+x+x^4,2,2}$ is an incompatible system. A way to linearize the problem is developing the product

$$(b_0 + b_1 x + \cdots + b_s x^s)\, f(x) = \sum_{k=0}^{\infty} H_{fsk} x^k,$$

with each H_{fsk} in $\mathbb{K}[b_0, b_1, \ldots, b_s]$ linear homogeneous for $f(x) \in \mathbb{K}[[x]]$, and impose $\sum_{k=0}^{r+s} H_{fsk} x^k = a_0 + a_1 x + \cdots + a_r x^r$. This yields a homogeneous linear system

$$\{H_{fsk} = a_k\}_{k=0}^{r} \cup \{H_{fsk} = 0\}_{k=r+1}^{r+s}. \tag{4.3}$$

The subsystem $\{H_{fsk} = a_k\}_{k=0}^{r}$ gives explicit formulae for $a_0, a_1, \ldots, a_r$ in terms of $b_0, b_1, \ldots, b_s$. The subsystem $L_{frs} = \{H_{fsk} = 0\}_{k=r+1}^{r+s}$ is a homogeneous linear system in $b_0, b_1, \ldots, b_s$. For been homogeneous linear, L_{frs} is consistent for all f, r and s, and so is (4.3). Moreover, L_{frs} always has a non-null solution, and so does (4.3).

A solution of S_{frs} is always a solution of (4.3), but not the converse, as we saw with the example $f(x) = 1 + x + x^4$ and $r = s = 2$. So I define a *weak Padé approximant* for a power series f and degrees r and s as a rational function g satisfying (4.1) and (4.3). Contrary to strong Padé approximants, the weak ones always exist. A strong Padé approximant is always a weak one, which justifies the terminology, and a weak Padé approximant with $b_0 \neq 0$ is a strong one.

If we have two (weak) Padé approximants g_1 and g_2 for the same f, r and s, then they are equal. Indeed, if $g_1 = A_1/B_1$ and $g_2 = A_2/B_2$ with A_1 and A_2 polynomials of degree r at most, B_1 and B_2 polynomials of degree s at most and $B_1 f \equiv A_1$ and $B_2 f \equiv A_2$ modulo x^{r+s+1}, then $B_1 B_2 f \equiv A_1 B_2 \equiv A_2 B_1$ modulo x^{r+s+1}. As $A_1 B_2$ and $A_2 B_1$ are polynomials of degree $r+s$ at most, we have $A_1 B_2 = A_2 B_1$ and hence $g_1 = g_2$. So we may speak of *the* Padé approximant of f with degrees r and s, usually denoted $[r/s]_f$. We have existence and uniqueness for weak approximants and also uniqueness for strong approximants. If the weak approximant g satisfies (4.2), then it is the strong approximant. If it does not, Padé defined the *deficiency index* ω_{frs} such that f and g have contact of order $r + s - \omega_{frs}$.

How are the Padé approximants of a rational function?

Theorem 85. *We have $f(x) \in \mathbb{K}(x)$ of the form*

$$f(x) = \frac{\alpha_0 + \alpha_1 x + \cdots + \alpha_\rho x^\rho}{1 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_\sigma x^\sigma} \tag{4.4}$$

if and only if $[r/s]_f = f$ for all $r \geqslant \rho$ and $s \geqslant \sigma$. [Bak75, thm. 2.2]

According to the cited reference, this result was proved for the weak definition of Padé approximants by H. Padé in [Pad1892], and was later proved for the strong one. The easy half of the theorem, that (4.4) implies $[r/s]_f = f$ for all $r \geqslant \rho$ and $s \geqslant \sigma$, which is the part the we need for this thesis, is easily proved by uniqueness of $[r/s]_f$ and that f has the required contact order for being a strong Padé approximant of itself.

Remark 86. Theorem 85 guarantees the equality $f = g$, from (4.4) and (4.1) respectively, as rational functions, but it does not guarantee the equality of numerators and denominators, even if $b_0 = 1$. See the following counterexamples.

Example 87. Consider the power series expansion $f(x) = \sum_{k=0}^{\infty} 2^k x^k$ of the rational function $1/(1-2x)$. The equations for the Padé approximant $g = [2/2]_f$ are

$$\begin{aligned} a_0 &= b_0, \\ a_1 &= 2b_0 + b_1, \\ a_2 &= 4b_0 + 2b_1 + b_2, \\ 0 &= 8b_0 + 4b_1 + 2b_2, \\ 0 &= 16b_0 + 8b_1 + 4b_2. \end{aligned}$$

Notice that the fifth equation is double the fourth, so L_{f22} has rank 1. Taking b_0 and b_1 as parameters, the solution is

$$\begin{aligned} a_0 &= b_0, \\ a_1 &= 2b_0 + b_1, \\ a_2 &= 0, \\ b_2 &= -4b_0 - 2b_1. \end{aligned}$$

Notice that the choice $b_0 = 1$ is possible but it does not make the system determined. In any case $(b_0, b_1) \neq (0, 0)$, we have

$$g(x) = \frac{b_0 + (2b_0 + b_1)\,x}{b_0 + b_1 x + (-4b_0 - 2b_1)\,x^2} = \frac{\big(b_0 + (2b_0 + b_1)\,x\big)}{\big(b_0 + (2b_0 + b_1)\,x\big)(1 - 2x)},$$

which simplifies to the exact value of $f(x)$. Only $(b_0, b_1) = (1, -2)$ gives the equality of numerator and denominator.

Example 88. Let us compute $g = [2/2]_f$ for $f(x) = 1 + x + x^4$. The equations are the following:

$$\begin{aligned} a_0 &= b_0, \\ a_1 &= b_0 + b_1, \\ a_2 &= \quad b_1 + b_2, \\ 0 &= \quad\quad b_2, \\ 0 &= b_0. \end{aligned}$$

Notice that L_{f22} reduces to $b_0 = b_2 = 0$, so the only choice of parameter is b_1 and the solution is $a_0 = b_0 = b_2 = 0$ and $a_1 = a_2 = b_1$. In any case $b_1 \neq 0$, we have

$$g(x) = \frac{b_1 x + b_1 x^2}{b_1 x} = \frac{(b_1 x)(1 + x)}{(b_1 x)},$$

which simplifies to $1 + x$, but the numerator and the denominator are different.

4.2.2 The problem of the order

The origin may be a pole of the rational function to reconstruct. This may happen even for a rational function $h(x)$ that appears as coefficients of the Darboux polynomials described in §4.1, since a regular point of the differential equation may be a pole of $h(x)$, as shown in the following example.

Example 89. Let us consider the differential equation $y'' = 0$, which admits the Singerian solution x. The Darboux polynomial associated to x is $H = xX_1 + X_2$ and, as x is not the null function, we may normalize the Darboux polynomial as $P = X_1 + \frac{1}{x}X_2$, which is associated to the line $\mathbb{K}x$. As P is defined in $\mathbb{K}(x)$, it is a final Darboux polynomial of the equation. Notice that the origin is a regular point of $y'' = 0$ but a pole of $1/x$, a coefficient of P.

If we know a bound k of the order of the pole of $h(x)$, the problem is solved by taking $f = x^k h(x)$, but we do not know such a bound even in the case of the coefficients of the Darboux polynomials constructed in §4.1. In this case, recall, we start with a Singerian solution $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ with $y_1 \neq 0$. If $\{\ell_1, \ell_2, \ldots, \ell_t\}$ is the orbit of $\mathbb{K}\mathbf{y}$ by the differential Galois group, we may choose Galois automorphism $\sigma_1, \sigma_2, \ldots, \sigma_t$ such that each ℓ_i is generated by $(\sigma_i(y_1), \sigma_i(y_2), \ldots, \sigma_i(y_n))$. Each ℓ_i is represented by the Darboux polynomial

$$P_{\ell_i}(X_1, X_2, \ldots, X_n) = X_1 + \frac{\sigma_i(y_2)}{\sigma_i(y_1)}X_2 + \cdots + \frac{\sigma_i(y_n)}{\sigma_i(y_1)}X_n,$$

and the orbit is represented by the Darboux polynomial

$$P = P_{\ell_1} P_{\ell_2} \cdots P_{\ell_t} = \frac{\prod_{i=1}^{t} (\sigma_i(y_1)\, X_1 + \sigma_i(y_2)\, X_2 + \cdots + \sigma_i(y_n)\, X_n)}{\sigma_1(y_1)\, \sigma_2(y_1) \cdots \sigma_t(y_1)},$$

whose coefficients are members of $\mathbb{Z}[\sigma_i(y_j); 1 \leqslant i \leqslant t, 1 \leqslant j \leqslant n]$ divided by the non-null product $\sigma_1(y_1)\, \sigma_2(y_1) \cdots \sigma_t(y_1)$. The $\sigma_i(y_j)$ are *effective power series*, a power series with an algorithm for computing any desired term (as an effective complex number), but their order may be unknown. All we can do is computing their approximated order, considering null the coefficients that are too close to zero. This order (as a series) is computed exactly for precision fine enough, but it may be overcomputed for poor precision. In any case, we have a non-null (pseudo)leading term in order to compute a (pseudo)quotient of power series, an effective power series divided by x to the sum of the (pseudo)orders of the $\sigma_i(y_1)$.

Another way to deal with the situation is the Padé approximation of a quotient of series. In our case, the denominator is $\sigma_1(y_1)\, \sigma_2(y_1) \cdots \sigma_t(y_1)$. The condition (4.3) of weak Padé approximation, for $f(x)$ the quotient $N(x)/D(x)$ of formal power series, yields

$$(b_0 + b_1 x + \cdots + b_s x^s)\, \frac{N(x)}{D(x)} \equiv a_0 + a_1 x + \cdots + a_r x^r \mod x^{r+s+1}. \tag{4.5}$$

If $D(0) \neq 0$, this is equivalent to

$$(b_0 + b_1 x + \cdots + b_s x^s)\, N(x) \equiv (a_0 + a_1 x + \cdots + a_r x^r)\, D(x) \mod x^{r+s+1}, \tag{4.6}$$

but (4.6) is weaker than (4.5) if $D(0) = 0$. The former case is considered in [Gra72, §4], and the latter case may be a path to explore.

4.2.3 Computation of Padé approximants

Padé approximants can be computed by direct solution of the system (4.3), which consists of the subsystem L_{frs}, a homogeneous linear system in $b_0, b_1, \ldots, b_s$, and explicit formulae for $a_0, a_1, \ldots, a_r$ in terms of $b_0, b_1, \ldots, b_s$. We pick a non-null solution of L_{frs} (see §4.3 on how to pick it) and compute $[r/s]_f$, which is the same regardless of the choice of solution of L_{frs}. When the computation is done with effective complex numbers, a coefficient too close to zero may be deemed zero and, if the computation is not aborted and restarted at finer precision, the result may be completely erroneous. Anyway, when tol is small enough, the computation is exact and the result is correct.

Gaussian elimination, or any other solver of (4.3), is an effective way of computing Padé approximants, but there are more efficient methods, as described in [BGM81, §2.4], some of them involving the extended Euclidean algorithm, as briefly explained below. Moreover, L_{frs} has a particular structure, a so-called *Toeplitz* system, and Toeplitz systems can be solved with specific methods, but one of these methods, described in [BGY80], precisely reduces a Toeplitz systems to a problem of Padé approximation, which is solved using a fast version of the extended Euclidean algorithm.

The Padé approximants are usually arranged in an infinite matrix called the *Padé table*. The antidiagonals of the Padé table correspond to the Padé approximants of the same sum $r + s$. These approximants appear as byproducts of the extended Euclidean algorithm for x^{r+s+1} and the truncation $F(x)$ of $f(x)$ at degree $r + s$, as described in [BGY80, §4] and [vzGG03, §5.9]. The first Padé approximant found is $[r+s \,/\, 0]_f = F(x)$, and then the algorithm proceeds by decreasing the degree of the numerator and increasing the degree of the denominator, keeping the sum $r+s$. This increasing and decreasing of the degrees may be done by steps of 1, but may not. Each computed Padé approximants has a multiplicity in the antidiagonal equal to the degree of the partial quotient in the Euclidean algorithm, according to [BGY80, lem. 2].

There exists a fast version of the extended Euclidean algorithm, described in [BGY80, §3] and [vzGG03, §11.1], which exploits the fact that most computations in the extended Euclidean algorithm need only a few leading terms instead the whole polynomial. This algorithm can compute fast and correctly the intermediate values that determine any Padé approximant in the antidiagonal, as stated in [vzGG03, corol. 11.6], but not all of them together, as said in [vzGG03, p. 309]. As we will see later, we will not need different Padé approximants in the same antidiagonal, so the fast algorithm can be applied.

Same as with the direct solution of (4.3), when the computation with the extended Euclidean algorithm is done with effective complex numbers, a coefficient too close to zero may be deemed zero and, if the computation is not aborted and restarted at finer precision, the result may be completely erroneous. Anyway, when tol is small enough, the computation is exact and the result is correct.

Computing by direct solution of (4.3), as shown in Example 87, may output the Padé approximant not in its lowest terms. In particular, the computation for the same f for greater r and s may yield new numbers to reconstruct as in §4.4. A way to find the solution of (4.3) yielding a Padé approximant in its lowest terms is to choose as parameters b_s, b_{s-1} and so on, as long as possible, and to assign them

to zero. Computing by the extended Euclidean algorithm may output the Padé approximant not in its lowest terms, but not as bad as Example 87, only as in Example 88. According to [BGY80, p. 271, ll. 13–17], the greatest common divisor of numerator and denominator is a power of x. Even if the Padé approximant is not simplified, once $r \geqslant \rho$ and $s \geqslant \sigma$ according to Theorem 85, it will yield the same numbers to reconstruct when the computation is redone for greater r and s.

In order to compute a Padé approximant, we needs bounds for the degree of the numerator and of the denominator. Some bounds are discussed in [vHW97]. In case of ignorance, we use a general[1] global parameter B. Fixing $r = s = \mathsf{B}$, there are algorithms, as in [BGY80, §5], that allow to easily compute the Padé approximants in the main diagonal of the Padé table.

4.3 Algebraicity of the numeric coefficients

If we start with a field of constants $\mathbb{K}$ algebraically closed, after analytic operations we cannot warrant that the result is defined over $\mathbb{K}$, only over $\mathbb{C}$. We have examples that show how we find constants transcendental over $\mathbb{K}$ even if all the vertices of analytic continuation belong to $\mathbb{K}$.

Example 90. Let us continue analytically $\{y' = y, y(0) = 1\}$, defined over $\mathbb{Q}$, to 1, also rational. The solution is $\{y' = y, y(1) = \mathrm{e}\}$, which is not defined over $\overline{\mathbb{Q}}$.

Even returning to the same rational point, we may find transcendental numbers.

Example 91. Let us continue analytically $\{y' = 1/x, y(1) = 0\}$, defined over $\mathbb{Q}$, to 1 after a loop around the origin counterclockwise. The solution is $\{y' = 1/x, y(1) = 2\pi\mathrm{i}\}$, which is not defined over $\overline{\mathbb{Q}}$.

Despite these examples, the numeric coefficients that appear in the rational functions reconstructed in the previous section lie in $\mathbb{K}$, as I shall prove. According to [vdH07a, §2.2¶4], if we express in the same fundamental system of solutions (defined over $\mathbb{K}$) the differential Galois groups G with the base field $\mathbb{K}$ and $G_{\mathbb{C}}$ with $\mathbb{C}$, then $G = G_{\mathbb{C}} \cap \mathrm{GL}(n, \mathbb{K})$; in the language of §3.7, $G_{\mathbb{C}}$ is the extension of G to the complex base field. If H and $H_{\mathbb{C}}$ are the respective eurymeric closures of G and $G_{\mathbb{C}}$, as Theorem 71 proves, $H_{\mathbb{C}}$ is the extension of H to the complex base field. According to Proposition 77, the truncated linearized Derksen–van der Hoeven algorithm over $\mathbb{C}$ is performed in the same way as over $\mathbb{K}$. In the construction

[1] See §2.5 for an introduction to general global parameters.

described in §4.1, just as in [vdH07a, §3.4], we normalize our object so that it must be defined over $\mathbb{K}$, provided that the chosen common eigenvector is defined over $\mathbb{K}$. The following example shows that this choice is a necessary condition.

Example 92. The 2×2 system $\{\mathbf{y}' = \mathbf{y}\}$ is defined over $\mathbb{Q}$. The fundamental system of solutions $\{f\,\mathbf{e}_1, f\,\mathbf{e}_2\}$, where $\{\mathbf{e}_1, \mathbf{e}_2\}$ is the standard basis of $\mathbb{K}^2$ and f the solution of $\{y' = y, y(0) = 1\}$ from Example 90, is also defined over $\mathbb{Q}$. The differential Galois group is $\mathbb{K}^*\mathtt{I}_2$, which is broad (thus connected and eurymeric) and defined over $\mathbb{Q}$. For $\mathbb{K} = \mathbb{C}$, all the elements of $\mathbb{C}^2$ are eigenvectors of $\mathbb{C}^*\mathtt{I}_2$, so we may pick $(1, \mathrm{e})$, which corresponds to the solution $f\,\mathbf{e}_1 + \mathrm{e}f\,\mathbf{e}_2$. The associated Darboux polynomial is $X_1 + \mathrm{e}X_2$, which is not defined over $\overline{\mathbb{Q}}$.

Remark 93. The existence of such an eigenvector is granted, but we need to find it effectively. If we chose the eigenvector $(1, \mathrm{e})$ as in Example 92, the reconstruction of the numbers would fail because e is transcendental, and this would cause the algorithm not to terminate.

The penultimate paragraph of [vdH07a, §4.5] describes a method for computing a basis over $\mathbb{K}$ of a given basis of a complex vector subspace of $\mathbb{C}^n$ that is defined over $\mathbb{K}$. If the vectors are given as rows, the device is reducing to the echelon form the resulting matrix. For a vector subspace of $\mathfrak{gl}(n, \mathbb{C})$, the row vector corresponding to a matrix is the juxtaposition of its rows.

There are two ways for finding a common eigenvector of a complex vector subspace of $\mathfrak{gl}(n, \mathbb{C})$ defined over $\mathbb{K}$. The first way is to find a basis over $\mathbb{K}$ before computing the common eigenvectors, so that we compute in $\mathfrak{gl}(n, \mathbb{K})$. The second way is to reduce the basis of the chosen eigenspace so that any vector lies in $\mathbb{K}^n$. We may apply any of the methods; applying both is redundant, but we need one in order that the algorithm terminates, as Remark 93 says.

4.4 Reconstruction of numbers

In order to reconstruct the symbolic expression of an effective complex number β in $\mathbb{K} = \overline{\mathbb{Q}(\alpha_1, \alpha_2, \ldots, \alpha_r)}$, we need to find a polynomial $P \in \mathbb{Z}[X_0, X_1, \ldots, X_r]$ such that $P(\beta, \alpha_1, \alpha_2, \ldots, \alpha_r) = 0$, This is equivalent to looking for *additive syzygies*[2] among the products of $\alpha_1, \alpha_2, \ldots, \alpha_r$ and β. If $\{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ is a transcendence basis of $\mathbb{K}$, then β must be present in any syzygy.

[2] An additive syzygy of $(a_1, a_2, \ldots, a_s) \in \mathbb{C}^s$ is $(k_1, k_2, \ldots, k_s) \in \mathbb{Z}^s$ such that $k_1a_1 + k_2a_2 + \cdots + k_sa_s = 0$.

There are different algorithms for looking for additive syzygies among the real numbers, and some are generalizable to complex numbers. If we have got a bound of the degree, as in [Hen96, §2.5]=[HvdP95, §5], we use it. If we have not, we use a general[3] global parameter D.

J. van der Hoeven proposes in [vdH07a, §3.4] using the LLL algorithm, which is not an algorithm for additive syzygies, but can be used trickily to get some, as I shall describe in §4.4.2. It was described in [LLL82, §1] as an auxiliary algorithm for factorization in $\mathbb{Q}[x]$ in polynomial time, and has plenty applications, especially in cryptanalysis, as exposed in [JS98]. According to [KLL88], reconstructing symbolically a number is essentially analyzing its binary expansion as a pseudo-random number generator. Moreover, [KLL88] introduces an algorithm for factorization in $\mathbb{Q}[x]$ in polynomial time simpler than [LLL82]. The idea is to construct the minimal polynomial of a root of the polynomial to factor, which is possible with the application of the LLL algorithm in [KLL88, (1.16)].

Another algorithm (related to the LLL) is HJLS, which is an actual algorithm for additive syzygies, but the most renowned of these algorithms is PSLQ, which has a version for complex numbers exposed in [FBA99]. Actual algorithms for additive syzygies, under exact arithmetic, either compute a syzygy or give a inferior bound of the coefficients of a syzygy. They are used for reconstructing symbolic expressions of numbers, for instance in the Inverse Symbolic Calculator [Plo95]; see also [BHM02]. They are found to be very useful in the so-called experimental mathematics, computing candidate syzygies numerically and then proving them rigorously. For instance, as told in [BB01, §4], these algorithms were used to find a formula that gives an individual hexadecimal digit of π without computing the previous ones, as described in [Pet09, §4], which was proved afterward.

The next subsections will be devoted to explain roughly these algorithms, in the context useful for our needs, and to compare them.

4.4.1 About the LLL algorithm

The LLL algorithm is an algorithm for *reducing* the *basis* of a *lattice*. Let me introduce these terms. If $B = \{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_r\}$ is a system of r vectors in $\mathbb{R}^n$ linearly independent over $\mathbb{R}$, we define the additive subgroup $L(B) = \mathbb{Z}\mathbf{a}_1 + \mathbb{Z}\mathbf{a}_2 + \cdots + \mathbb{Z}\mathbf{a}_r$ of $\mathbb{R}^n$. We call $L(B)$ a *lattice*,[4] B a *basis* thereof, and r its *rank*. If $r = n$, we speak

[3] See §2.5 for an introduction to general global parameters.

[4] Notice that, in this context, "lattice" has a different meaning from in the previous chapters.

of a *full-rank* lattice. Some expositions, as the original LLL article, are restricted to full-rank lattices, but this is not an essential restriction, since $L(B)$ can be embedded in $\mathbb{R}^r$. The vector space V spanned by B can be identified with $\mathbb{R}^r$ by an orthonormal basis. The image of $L(B)$ in $\mathbb{R}^r$ is a full-rank lattice and has all the properties of $L(B)$. Others expositions, as the present one, consider lattices for $r \leqslant n$. This latter option makes clearer to deal with the lattices constructed when looking for syzygies.

Lattices are discrete: there is a *minimum distance* between points of a lattice. With any norm of $\mathbb{R}^n$, it suffices to prove that there is a minimum norm among the nonzero vectors of a lattice, which is done in Proposition 94 for $M = \{\mathbf{0}\}$. A vector that achieves this minimum length is called the *shortest vector* of the lattice, by a little abuse of language.

Proposition 94. *If M is a subset of the lattice $L(B)$, there is a minimum λ_M of the norm (any norm of $\mathbb{R}^n$) in $L(B) \setminus M$, and $\lambda_M > 0$ if $\mathbf{0} \in M$.*

Proof. Let $\lambda_M = \inf\{\|\mathbf{x}\| : \mathbf{x} \in L(B) \setminus M\}$. There exists a succession $(\mathbf{x}_k)_{k=1}^{\infty}$ in $L(B) \setminus M$ with $\lim_k \|\mathbf{x}_k\| = \lambda_M$. As the length of $\mathbf{x}_k$ is bounded, this succession is contained in a compact subset of $\mathbb{R}^n$ and, so, it admits a convergent subsuccession $(\mathbf{y}_k)_{k=1}^{\infty}$ with limit $\mathbf{y}_0$. By continuity of the norm, $\|\mathbf{y}_0\| = \lambda_M$. Decomposing each $\mathbf{y}_k$ in B as $\mathbf{y}_k = \sum_{i=1}^{r} y_{ki}\mathbf{a}_i$, we have $\lim_k y_{ki} = y_{0i}$ for each i. As each y_{ki} is integer, there is k_0 such that $y_{ki} = y_{0i}$ for each $k \geqslant k_0$ and $1 \leqslant i \leqslant r$, so $\mathbf{y}_k = \mathbf{y}_0$ for $k \geqslant k_0$. Consequently, $\mathbf{y}_0 \in L(B) \setminus M$, so λ_M is a minimum. Moreover, if $\mathbf{0} \in M$, $\mathbf{y}_0 \neq \mathbf{0}$, so $\lambda_M > 0$. □

Remark 95. The definition of λ_M does not depend on the basis B, but only on the lattice $L(B)$ and the norm in $\mathbb{R}^n$.

Among the different bases a lattice has, we look for an analog of orthonormal bases, with the Euclidean metric inherited from $\mathbb{R}^n$. The operations on a basis that yield another basis of the same lattice are a restricted kind of Gaussian reduction: swapping two vectors, changing the sign of a vector and adding to a vector an integer multiple of another vector. As the length is discrete in a lattice, we look for minimal-length vectors instead of unit ones, a condition codified by the concept of *successive minima*. For $1 \leqslant i \leqslant r$, the i-th *successive minimum* of a lattice L is the minimum radius λ_i of the closed ball K centered at the origin such that $K \cap L$ has $\mathbb{R}$-rank i at least. The existence of λ_i is granted by Proposition 94, with M the vectors $\mathbf{x} \in L$ such that the closed ball of center the origin and radius $|\mathbf{x}|$ has rank lower than i, so the successive minima are achieved. The first minimum of a lattice is the length of the shortest vector thereof.

The ideal basis of a lattice would be an orthogonal basis of successive minima, but this is hard for two reasons. Computing the shortest vector in a lattice is hard (see [MG02, Chap. 4] for a detailed discussion of the computational-complexity hardness), so computing a basis of successive minima is harder, if not impossible. For instance, the lattice $(2\mathbb{Z})^n \cup (2\mathbb{Z}+1)^n$ from [MG02, p. 126] has all its successive minima equal to 2 if $n \geqslant 5$, but a basis of successive minima only generates the sublattice $(2\mathbb{Z})^n$. Moreover, a lattice may lack orthogonal bases, even of pairs of orthogonal vectors, as the lattice generated by the basis $B = \{(1,0), (\sqrt{2}, 1)\}$. So we look for bases consisting of almost orthogonal, almost minimal-length vectors. This concept is codified in the definition of a reduced basis.

Let $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_r\}$ be the Gram-Schmidt orthogonalization of B, where each $\mathbf{b}_i$ is the projection of $\mathbf{a}_i$ orthogonal to $V_i = \mathbb{R}\mathbf{a}_1 + \mathbb{R}\mathbf{a}_2 + \cdots + \mathbb{R}\mathbf{a}_{i-1}$ (onto $V_i^\perp$). We say that B is (δ, η)-*reduced*, for $\frac{1}{4} < \delta \leqslant 1$ and $\frac{1}{2} \leqslant \eta < \sqrt{\delta}$, if $|\mu_{ij}| \leqslant \eta$ and $\delta|\mathbf{b}_j|^2 \leqslant |\mathbf{c}_{j+1}|^2$, where $\mu_{ij} = (\mathbf{a}_i \cdot \mathbf{b}_j)/|\mathbf{b}_j|^2$ and $\mathbf{c}_i$ is the projection of $\mathbf{a}_i$ orthogonal to V_{i-1} (onto $V_{i-1}^\perp$), for $1 \leqslant j < i \leqslant r$. The basis B is orthogonal if and only if every μ_{ij} is zero, so the condition $|\mu_{ij}| \leqslant \eta$ codifies the almost orthogonality. The parameter $\delta = \frac{3}{4}$ is a trick that the authors of the LLL algorithm introduced in order to grant that their algorithm takes polynomial time, but it works whenever $\frac{1}{4} < \delta < 1$. The ideal would be $\delta = 1$, but this only works for $r = 2$, as explained below. The parameter η is a trick introduced for dealing with implementations of the LLL algorithm in inexact arithmetic. The ideal would be $\eta = \frac{1}{2}$, but in practice we need $\eta = \frac{1}{2} + \varepsilon$ for small tolerance ε.

How far from the shortest is the length of $\mathbf{a}_1$ in a (δ, η)-reduced basis?

Proposition 96. *If B is a (δ, η)-reduced basis,*

$$|\mathbf{a}_1| \leqslant \left(\frac{1}{\delta - \eta^2}\right)^{\frac{r-1}{2}} \min\{|\mathbf{x}| : \mathbf{x} \in L(B),\ \mathbf{x} \neq \mathbf{0}\}.$$

Proof. This proof is adapted from [Mei09, p. 56, thm. 2.2]=[Mei01, p. 9, thm. 2.1] to the case of general δ and η. For $1 < i \leqslant r$, we have

$$\mathbf{c}_i = \mathbf{b}_i + \mu_{i,i-1}\mathbf{b}_{i-1}.$$

As $\mathbf{b}_i \perp \mathbf{b}_{i-1}$,

$$|\mathbf{c}_i|^2 = |\mathbf{b}_i|^2 + |\mu_{i,i-1}|^2|\mathbf{b}_{i-1}|^2.$$

As $\delta|\mathbf{b}_j|^2 \leqslant |\mathbf{c}_{j+1}|^2$,

$$\delta|\mathbf{b}_{i-1}|^2 \leqslant |\mathbf{b}_i|^2 + |\mu_{i,i-1}|^2|\mathbf{b}_{i-1}|^2.$$

As $|\mu_{i,i-1}| \leqslant \eta$,

$$\delta|\mathbf{b}_{i-1}|^2 \leqslant |\mathbf{b}_i|^2 + \eta^2|\mathbf{b}_{i-1}|^2$$

and

$$(\delta - \eta^2)|\mathbf{b}_{i-1}|^2 \leqslant |\mathbf{b}_i|^2.$$

Notice that $0 < \delta - \eta^2 \leqslant \frac{3}{4}$. By induction

$$(\delta - \eta^2)^{i-1}|\mathbf{b}_1|^2 \leqslant |\mathbf{b}_i|^2$$

and thus

$$|\mathbf{b}_i|^2 \geqslant (\delta - \eta^2)^{r-1}|\mathbf{b}_1|^2 = (\delta - \eta^2)^{r-1}|\mathbf{a}_1|^2.$$

Notice that the last formula holds also for $i = 1$.

Let $\mathbf{x}$ be the shortest[5] vector of $L(B)$ and $\mathbf{x} = \sum_{i=1}^{N} x_i\mathbf{a}_i$ its decomposition in B with $x_N \neq 0$. If we decompose $\mathbf{x}$ in the orthogonalized basis, we have $\mathbf{x} = \sum_{i=1}^{N} y_i\mathbf{b}_i$ with $y_N = x_N$. By Pythagoras, $|\mathbf{x}|^2 = \sum_{i=1}^{N} |y_i|^2|\mathbf{b}_i|^2$. As y_N is a nonzero integer,

$$|\mathbf{x}|^2 \geqslant |y_N|^2|\mathbf{b}_N|^2 \geqslant |\mathbf{b}_N|^2 \geqslant (\delta - \eta^2)^{r-1}|\mathbf{a}_1|^2,$$

so

$$|\mathbf{a}_1| \leqslant \left(\frac{1}{\delta - \eta^2}\right)^{\frac{r-1}{2}} |\mathbf{x}|. \qquad \square$$

The case $r = 2$ is much simpler and illustrative and, if we work with exact arithmetic, we can compute a $(1, \frac{1}{2})$-reduced basis of successive minima in polynomial time. In this case, a reduced basis $\{\mathbf{a}_1, \mathbf{a}_2\}$ has $|\mathbf{a}_1| \leqslant |\mathbf{a}_2|$ and the angle θ between $\mathbf{a}_1$ and $\mathbf{a}_2$ satisfies $\frac{1}{3}\boldsymbol{\pi} \leqslant \theta \leqslant \frac{2}{3}\boldsymbol{\pi}$; see [Vaz01, §27.2] and [Dwo98, Ch. 5]. Unfortunately the converse is not true, as the counterexample $\{(1, 0), (1, 2)\}$ shows. For this case, an efficient basis reduction algorithm is known since Gauss, who included it in his *Diquisitiones Arithmeticae* [Gau1801], though [Ngu10] attributes it to Lagrange. It is a generalization to vectors of Euclid's algorithm[6]. The algorithm is the combination of a rounded-off version of the Gram-Schmidt orthogonalization process, in order to keep $|\mu_{12}| \leqslant \frac{1}{2}$, and a reordering of the basis in order to keep $|\mathbf{a}_1| \leqslant |\mathbf{a}_2|$. As the Gram-Schmidt orthogonalization process is $\mathbf{a}_2 := \mathbf{a}_2 - \mu_{12}\mathbf{a}_1$, its lattice version is $\mathbf{a}_2 := \mathbf{a}_2 - m\mathbf{a}_1$, where m the nearest integer to μ_{12}. After this step, the new value of μ_{12} satisfies $|\mu_{12}| \leqslant \frac{1}{2}$. Moreover, the rounding-off of

[5] Recall that this means any minimum of $L(B) \setminus \{\mathbf{0}\}$ in Euclidean norm.

[6] See footnote 8 on page 80.

the old value of μ_{12} is the only[7] value of m that makes the new value of μ_{12} satisfy $|\mu_{12}| \leqslant \frac{1}{2}$. If, after this step, the new values satisfy $|\mathbf{a}_1| \leqslant |\mathbf{a}_2|$, these values are invariant under the two steps of the algorithm, so the algorithm terminates. Else, when $|\mathbf{a}_1| > |\mathbf{a}_2|$, we swap $\mathbf{a}_1 \leftrightarrow \mathbf{a}_2$ and repeat the process. The algorithm terminates in polynomial time, according to [Dwo98, prop. 5.2.1], [Vaz01, p. 278] and [Beu99b, Ex. 3.3]. According to [Vaz01, thm. 27.5], [Dwo98, prop. 5.1.2] and [Beu99b, Alg. 3.1], a Gauss-reduced basis consists of a shortest vector and a successive minimum.

The LLL algorithm was introduced in [LLL82, §1] as an auxiliary of the first algorithm for factoring in $\mathbb{Q}[x]$ in polynomial time, which they introduced in [LLL82, §3]. The name LLL comes from the initials of the authors: Lenstra, Lenstra[8] and Lovász. They considered only the full-rank case, but the algorithm works anyway for $r < n$. A detailed exposition of [LLL82] is found in the BSc thesis [Pet09]. Other expositions are found in [Beu99b]+[Beu99a], [vzGG03, Ch. 16], [Car02, §3.2.1+§4.2.1] and [Nov08, §1.2]. The LLL algorithm is also exposed in [Dwo98, Ch. 6], [Coh93, §2.6], [Bor02, App. B] and [Mei01, §2.1]=[Mei09, §2.1].

As the Gaussian algorithm, which is a simplification for $r = 2$, the LLL algorithm is the combination of a rounded-off version of the Gram-Schmidt orthogonalization process and a reordering of the basis. The first step is to compute the μ_{ij} and the corresponding nearest integer m_{ij} and to reduce $\mathbf{a}_i := \mathbf{a}_i - \sum_{j<i} m_{ij}\mathbf{a}_j$ for $1 < i \leqslant r$. Notice that each reduction changes the values of the μ_{ij}, but not of the $\mathbf{b}_i$, so the μ_{ij} must be recomputed. This step grants that, at the end, $|\mu_{ij}| \leqslant \frac{1}{2}$ for $1 \leqslant j < i \leqslant r$. The second step is to check if $\delta|\mathbf{b}_i|^2 \leqslant |\mathbf{c}_{i+1}|^2$ for $1 \leqslant i < r$. If the test is passed, the basis is $(\delta, \frac{1}{2})$-reduced. Else, for the least i with $\delta|\mathbf{b}_i|^2 > |\mathbf{c}_{i+1}|^2$, we swap $\mathbf{a}_i \leftrightarrow \mathbf{a}_{i+1}$ and repeat the process, beginning with the Gram-Schmidt process in order to recompute the $\mathbf{b}_i$ and the μ_{ij}. This exposition is not the most efficient,[9] but my only aim is to explain how the algorithm works. The algorithm terminates in polynomial time, according to [LLL82, prop. 1.26], [Car02, thm. 4] and [Vaz01, thm. 16.11].

A $(\frac{3}{4}, \frac{1}{2})$-reduced basis is enough for many applications, including the factorization in $\mathbb{Q}[x]$. The LLL authors construct a suitable lattice in [LLL82, §2] for the algorithm given in [LLL82, §3]. For a survey on factorization methods depending on lattice reduction, see [Klü10]; also [Nov08, Ch. 1]. Another application, introduced in [LLL82, prop. 1.39] and surveyed in [Han10], is the simultaneous

[7] In the case $\mu_{12} \in \mathbb{Z} + \frac{1}{2}$, both nearest integers hold the condition, and they are the only integers holding the condition.

[8] A. K. Lenstra and H. W. Lenstra Jr. are brothers.

[9] For instance, only a partial Gram-Schmidt process is needed after a swap.

Diophantine approximation, which is a generalization of continued fraction approximation with common denominator. But the application we are interested in is looking for syzygies, which is explained below.

4.4.2 Finding syzygies with the LLL algorithm

Though the LLL algorithm is an algorithm for reducing a lattice basis and its first proposed application was factorization in $\mathbb{Q}[x]$, its applicability for computing minimal polynomials of algebraic numbers is already mentioned in [LLL82] at the end of §1. The basic idea, exposed in [JS98, §2.5], is, for α real and C large enough, to reduce the basis given by the columns of the matrix

$$\begin{pmatrix} C & C\alpha & \dots & C\alpha^d \\ 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix},$$

where d is the guess for the degree of the minimal polynomial of α. The shortest vector, which is approximated by the first vector in the reduced basis, is of the form $(CP(\alpha), p_0, p_1, \dots, p_d)$ with $P(x) = p_0 + p_1 x + \dots + p_d x^d$ and $C^2 P(\alpha)^2 + p_0^2 + p_1^2 + \dots + p_d^2$ small. As C is large, $P(\alpha)$ is very small, thus α is approximately a root of $P(x)$. The LLL article fails to explain better this idea because of its restriction to full-rank lattices, when the lattice above has rank $d+1$ in $\mathbb{R}^{d+2}$. For α complex, also mentioned at the end of [LLL82, §1] but not explained, the basis to reduce is given by the columns of the following matrix

$$\begin{pmatrix} C & C\,\mathrm{Re}(\alpha) & \dots & C\,\mathrm{Re}(\alpha^d) \\ 0 & C\,\mathrm{Im}(\alpha) & \dots & C\,\mathrm{Im}(\alpha^d) \\ 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}.$$

The adequate value of C is given in [KLL88, (1.16)] in terms of the degree and the height[10] of α. If H is a bound of the height of α, [KLL88, (1.16)] takes $C = 2^s$

[10] The height of an algebraic number is the largest absolute value of the coefficients of its primitive minimal polynomial over $\mathbb{Z}$, that is, the supremum norm of the minimal syzygy that determines its minimal polynomial.

with s the least integer such that

$$2^s \geqslant 2^{d^2/2}(d+1)^{(3d+4)/2}H^{2d}.$$

If we do not have a bound H of the height of α, we use a general[11] global parameter H.

In order to reconstruct the minimal polynomial of $\beta \in \overline{\mathbb{Q}(\alpha_1, \alpha_2, \ldots, \alpha_r)}$, with $\{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ a transcendence basis, we consider the row vector $\mathtt{m}$ of all the monic monomials of degree d at most of $\alpha_1, \alpha_2, \ldots, \alpha_r$ and β, and the matrix

$$\begin{pmatrix} C\,\mathrm{Re}(\mathtt{m}) \\ C\,\mathrm{Im}(\mathtt{m}) \\ \mathtt{I} \end{pmatrix}.$$

We proceed with this matrix as in the previous paragraph. If $r = 1$ and α_1 has a certain shape, like π and the exponential or the logarithm of an algebraic number,[12] then one may find the adequate value of C, and [KLL88, §2¶11] claims that this result may extend for general α_1 and general r, but it is not further developed. What we know in any case is that, for C large enough, we find an exact syzygy provided it exists.

Proposition 97. *If $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{C}^n$ has (non-null) syzygies, then the LLL method will find one for C large enough.*

Proof. This proof is adapted from [Mei01, p. 12, lem. 2.1]=[Mei09, p. 59, lem. 2.1] to the complex case. We may discard the case $\mathbf{x} = \mathbf{0}$, where any $\mathbf{y} \in \mathbb{Z}^n$ is a syzygy. Let

$$S = \{\mathbf{y} \in \mathbb{Z}^n : \mathbf{y} \neq \mathbf{0},\ \mathbf{y} \cdot \mathbf{x} = 0\}$$

be the set of non-null syzygies of $\mathbf{x}$. According to Proposition 94 with B the standard basis of $\mathbb{R}^n$ and $M = \mathbb{Z}^n \setminus S$, the minimum

$$\lambda = \min\{|\mathbf{y}| : \mathbf{y} \in L(B) \setminus M\} = \min\{|\mathbf{y}| : \mathbf{y} \in S\}$$

is achieved by $\mathbf{m} \in \mathbb{Z}^n$. Let

$$R = \left(\frac{1}{\delta - \eta^2}\right)^{\frac{n-1}{2}} \lambda$$

for the parameters δ and η of the LLL algorithm. Consider the finite set of vectors

$$A = \{\mathbf{y} \in \mathbb{Z}^n : |\mathbf{y}| \leqslant R,\ \mathbf{y} \cdot \mathbf{x} \neq 0\},$$

[11] See §2.5 for an introduction to general global parameters.

[12] In general, it works for α_1 with known transcendence measure; see [Cij74b]+[Cij74a].

which is not empty because, if $x_i \neq 0$, the i-th vector $\mathbf{e}_i$ of the standard basis belongs to A. Choose a vector $\mathbf{y}_0 \in A$ minimizing $|\mathbf{y}_0 \cdot \mathbf{x}|$ and, for this $\mathbf{y}_0$, define $C_0 = R/|\mathbf{y}_0 \cdot \mathbf{x}|$. For $C > C_0$, let L be the lattice defined by the basis[13] $\{(C\operatorname{Re}(x_i), C\operatorname{Im}(x_i), \underline{\mathbf{e}_i})\}_{i=1}^n$, which is $L = \{\varphi(\mathbf{y}) : \mathbf{y} \in \mathbb{Z}^n\}$ with

$$\varphi : \mathbb{Z}^n \to L : \mathbf{y} \mapsto (C\operatorname{Re}(\mathbf{y} \cdot \mathbf{x}), C\operatorname{Im}(\mathbf{y} \cdot \mathbf{x}), \underline{\mathbf{y}}).$$

For $\mathbf{y} \in \mathbb{Z}^n$, the length of $\varphi(\mathbf{y})$ is

$$|\varphi(\mathbf{y})| = \sqrt{C^2|\mathbf{y} \cdot \mathbf{x}|^2 + |\mathbf{y}|^2} \geqslant \max\{C\,|\mathbf{y} \cdot \mathbf{x}|\,, |\mathbf{y}|\}.$$

As $|\varphi(\mathbf{m})| = |\mathbf{m}| = \lambda$, the shortest vector in L has length λ at most. If $\mathbf{y}$ is not a syzygy of $\mathbf{x}$, then $|\varphi(\mathbf{y})| > R$ (because, if $|\mathbf{y}| \leqslant R$, then $\mathbf{y} \in A$ and $C\,|\mathbf{y} \cdot \mathbf{x}| > C_0\,|\mathbf{y}_0 \cdot \mathbf{x}| = R$), so

$$|\varphi(\mathbf{y})| > \left(\frac{1}{\delta - \eta^2}\right)^{\frac{n-1}{2}} \min\{|\mathbf{v}| : \mathbf{v} \in L,\ \mathbf{v} \neq \mathbf{0}\}$$

and, according to Proposition 96, $\varphi(\mathbf{y})$ cannot be the first vector in a (δ, η)-reduced basis of L. Thus, the first vector $\varphi(\mathbf{z})$ of such a basis determines a non-null syzygy $\mathbf{z}$ of $\mathbf{x}$. □

Remark 98. Notice that C_0 depends on $\mathbf{x}$ and cannot be computed beforehand, and that the behavior for $C \leqslant C_0$ is unknown. There is no guarantee that the set of the C is connected. For instance, it might happen that, for $C_1 < C_2 < C_0$, $C = C_1$ finds a syzygy while $C = C_2$ finds none. Such a numerical accident is not excluded. For instance, in [FB92, §2¶4] they report a example where the HJLS algorithm works with low precision, fails with medium precision and works with high precision.

As we do not know how large C need to be for a successful syzygy computation, we use a general[14] global parameter C.

4.4.3 The LLL algorithm with effective real numbers

The LLL algorithm is implemented both in exact rational arithmetic and in finite-precision numerics. Both [JS98, §2.5] and [Jus90, §3] propose to round off the data and operate with exact arithmetic, but [Beu99b, §6] does not. For a survey on the

[13] Recall that a vector underlined represents the list of its entries.
[14] See §2.5 for an introduction to general global parameters.

topic, see [Ste10]. The implementation with effective real numbers would be easy because there is no zero-test in the LLL algorithm. There are only rounding μ_{ij} to the nearest integer and checking if $\delta|\mathbf{b}_i|^2 \leqslant |\mathbf{c}_{i+1}|^2$. Everything works exactly except when μ_{ij} is about the midpoint between two integer numbers and when $\delta|\mathbf{b}_i|^2$ is too close to $|\mathbf{c}_{i+1}|^2$. Moreover, for precision fine enough, the computations are exact. Nevertheless, the LLL algorithm with inexact arithmetic may loop forever, as [Ste10, p. 186] warns, if we choose $\eta = \frac{1}{2}$. Indeed, Stehlé provides in the webpage of his PhD thesis [Ste05] some examples of lattice bases that make a popular floating-point implementation of the LLL algorithm loop forever.

Assume we are working with tolerance $\varepsilon > 0$. A straightforward implementation of the LLL algorithm with $\eta = \frac{1}{2}$, when checking if $|\mu_{ij}| \leqslant \frac{1}{2}$, may yield $\frac{1}{2} < |\mu_{ij}|.\mathsf{aprox}(\varepsilon) < \frac{1}{2} + \varepsilon$ for $\frac{1}{2} - \varepsilon < |\mu_{ij}| \leqslant \frac{1}{2}$, and tries to reduce vectors that are already reduced and should be kept untouched. This might be the cause of Stehlé's infinite loop. So we must choose $\eta > \frac{1}{2} + \varepsilon$; this grants that $|\mu_{ij}| \leqslant \frac{1}{2}$ implies $|\mu_{ij}|.\mathsf{aprox}(\varepsilon) \leqslant \eta$ and hence already reduced vectors are kept untouched. What an implementation of the LLL algorithm with parameter (δ, η) computes is a $(\delta - \varepsilon, \eta + \varepsilon)$-reduced basis, though with some "noise" what forces more reductions and swaps than necessary for a $(\delta - \varepsilon, \eta + \varepsilon)$-reduced basis, but necessary for a $(\delta+\varepsilon, \eta-\varepsilon)$-reduction. This makes Proposition 96 hold for parameters $(\delta-\varepsilon, \eta+\varepsilon)$, and thus Proposition 97, but we must adapt the proof of termination of the LLL algorithm for parameters $(\delta + \varepsilon, \eta - \varepsilon)$.

The standard proof of termination of the LLL algorithm uses the determinant of a lattice, which I shall introduce. The following definitions are equivalent for a lattice basis B:

- the square root of the Gramian determinant of B,
- the absolute value of the determinant of the matrix of B, only for full-rank lattices,
- the r-dimensional volume of the parallelotope determined by B.

So, we define the *determinant* $d(B)$ of B as the real number determined by any of the definitions above. Any of these definitions is invariant by a the following Gaussian-reduction operations: swapping two vectors, changing the sign of a vector and adding to a vector a multiple of another vector. These operations include both the transformations between bases of the same lattice and the operations of the Gram-Schmidt orthogonalization process. Therefore, we define the *determinant* $d(L)$ of a lattice L as the determinant of any of its bases. Also, the determinant of

a lattice basis B is equal to the determinant of its Gram-Schmidt orthogonalization B', and hence equal to the product of the lengths of the vectors of B'.

Another ingredient of the proof of termination of the LLL algorithm for general lattices is Minkowski's bound; see, for instance, [Len08, p. 137].

Theorem 99 (Minkowski)**.** *If L is a lattice of rank r,*

$$\min\{|\mathbf{x}| : \mathbf{x} \in L,\ \mathbf{x} \neq \mathbf{0}\} \leqslant \sqrt{r}\, d(L)^{1/r}.$$

Now we can follow the proof of termination from [Beu99b, §4].

Theorem 100. *The LLL algorithm with parameters δ and η, implemented in effective real numbers with tolerance $\varepsilon > 0$, terminates in a finite number of steps, provided δ, η and ε are compatible (i.e., $\eta - \varepsilon \geqslant \frac{1}{2}$, $\eta + \varepsilon < \sqrt{\delta - \varepsilon}$, $\delta - \varepsilon > \frac{1}{4}$ and $\delta + \varepsilon < 1$).*

Proof. Let us use the terminology of §4.4.1 and consider the product

$$D = d(\mathbf{a}_1)\, d(\mathbf{a}_1, \mathbf{a}_2) \cdots d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{r-1}).$$

As the Gram-Schmidt orthogonalization of $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_i\}$ is $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_i\}$, for $1 \leqslant i \leqslant r$, we have

$$d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_i) = |\mathbf{b}_1|\, |\mathbf{b}_2| \cdots |\mathbf{b}_i|\,.$$

Let $M = \min\{|\mathbf{x}| : \mathbf{x} \in L(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_r),\ \mathbf{x} \neq \mathbf{0}\}$. For $1 \leqslant i \leqslant r$, we have

$$M \leqslant \min\{|\mathbf{x}| : \mathbf{x} \in L(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_i),\ \mathbf{x} \neq \mathbf{0}\} \leqslant \sqrt{i}\, d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_i)^{1/i}$$

by Theorem 99, so

$$d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_i) \geqslant \left(\frac{M}{\sqrt{i}}\right)^i \geqslant \left(\frac{M}{\sqrt{r}}\right)^i$$

and thus

$$D \geqslant \frac{M}{\sqrt{r}} \left(\frac{M}{\sqrt{r}}\right)^2 \cdots \left(\frac{M}{\sqrt{r}}\right)^{r-1} = \left(\frac{M}{\sqrt{r}}\right)^{\frac{r(r-1)}{2}}.$$

This is a lower bound for D independent of the lattice basis.

Let us consider the effect in D of the basis transformation performed in the LLL algorithm. Adding to a vector $\mathbf{a}_i$ a multiple $m\mathbf{a}_j$ of another vector may

modify the partial determinants $d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k)$ with $i \leqslant k < j$, but the LLL algorithm, imitating the Gram-Schmidt process with rounded-off coefficients, only performs transformations like $\mathbf{a}_i := \mathbf{a}_i + m\mathbf{a}_j$ for $j < i$, which keeps all the partial determinants $d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k)$ invariant. Swapping two consecutive basis vectors $\mathbf{a}_i \leftrightarrow \mathbf{a}_{i+1}$ only affects to the partial determinant $d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_j)$ for $i = j$, so

$$D' = D\,\frac{d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{i-1}, \mathbf{a}_{i+1})}{d(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_i)} = D\,\frac{|\mathbf{b}'_i|}{|\mathbf{b}_i|},$$

where the prime means the value after the swap. It is easy to check that $\mathbf{b}'_i = \mathbf{c}_{i+1}$, so

$$D' = D\,\frac{|\mathbf{c}_{i+1}|}{|\mathbf{b}_i|} < D\,\sqrt{\delta + \varepsilon}.$$

The lower bound for D at any stage and the upper bound of the factor of decreasing of D at an LLL swap, as the swap is the only basis transformation in the LLL algorithm that modifies D, imply that the LLL algorithm performs only a finite number of swaps and thus a finite number of iterations, so the LLL algorithm terminates in a finite number of steps. □

4.4.4 About the HJLS algorithm

Related to the LLL algorithm, there is the HJLS algorithm, which is an actual algorithm for finding syzygies. The HJLS name comes from the initials of the authors: Håstad, Just (née Helfrich), Lagarias and Schnorr. The HJLS algorithm was introduced in [HHLS86] and explained in [HJLS89]. They called it "Small Integer Relation Algorithm" in [HJLS89], Algorithm A in [HHLS86], where *integer relation* is what we have called additive syzygy. Another synonym is "Diophantine relation", used in [Ber80, §2, ¶4].

The HJLS algorithm combines ideas from [FF79], explained in [Ber80], and from [LLL82]. Apart from [HHLS86] and [HJLS89], the HJLS algorithm is explained in [Rom07, §2.2], [Mei01, §2.3], [Mei09, §2.3] and [Bor02, App. B]. Geometrically, as explained in [HJLS89, §1, ¶3], in order to find the syzygies of $\mathbf{x} \in \mathbb{R}^n$, they construct a sequence of bases of the lattice $\mathbb{Z}^n$ whose elements converge to $\mathbb{R}\mathbf{x}$ in the following way; the maximum distance between an element of the basis and the line $\mathbb{R}\mathbf{x}$ converges to zero. Starting with the standard basis, each new basis is worse in terms of the LLL algorithm, but its orthogonal projection onto $\mathbf{x}^\perp$ imitates the LLL reduction. As [HJLS89, §1, ¶3] says, the elements of the basis B are better and better Diophantine approximations of $\mathbf{x}$, seeing the vectors

as projective coordinates. For a variant of HJLS for computing good Diophantine approximations, see [Jus92]. The HJLS algorithm plays with the dual basis of B, which consists of the rows of the inverse of the matrix of B by columns, whose elements are closer and closer to syzygies of $\mathbf{x}$, as Diophantine approximation is the dual problem of syzygies, according to [HJLS89, §1, ¶6]. Instead of computing the inverse of the matrix of B when needed, the HJLS algorithm computes it step by step after each elementary transformation of B, applying the corresponding elementary transformation. Instead of applying the Gram-Schmidt orthogonalization process to $B = \{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\}$, it is applied to $\{\mathbf{a}_0, \mathbf{a}_1, \dots, \mathbf{a}_n\}$ with $\mathbf{a}_0 = \mathbf{x}$; the result $\{\mathbf{b}_0, \mathbf{b}_1, \dots, \mathbf{b}_n\}$ makes $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$ "almost" a basis of $\mathbf{x}^\perp$ ("almost" in the sense that we get a basis by suppressing the zero vectors). Along the HJLS algorithm, $\mathbf{b}_n = \mathbf{0}$; if $\mathbf{b}_n \neq \mathbf{0}$, then $\mathbf{b}_i = \mathbf{0}$ for certain $i < n$, so $\mathbf{x} \in \mathbb{R}\mathbf{a}_1 + \mathbb{R}\mathbf{a}_2 + \cdots + \mathbb{R}\mathbf{a}_i$ and thus the $n - i$ last elements of the dual basis are linearly independent syzygies; cf. [HJLS89, prop. 3.1, proof (1)]. If the HJLS algorithms does not terminate finding a syzygy, as explained below, the iteration of the reduction would produce systems $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$ converging to zero, so eventually $|\mathbf{b}_i| < 2^{-k}$ for $1 \leqslant i \leqslant n$, where k is an input datum, and, according to [HJLS89, prop. 3.1.(2)], any syzygy would have length longer than 2^k. If a syzygy is found, then its length is at most $2^{n/2-1} \min\{|\mathbf{m}| : \mathbf{m} \in \mathbb{Z}^n, \mathbf{m} \neq \mathbf{0}, \mathbf{m} \cdot \mathbf{x} = 0\}$, according to [HJLS89, prop. 3.1.(3)].

The main loop of the HJLS algorithm does the following. It chooses i maximizing $2^i|\mathbf{b}_i|^2$, reduces $\mathbf{a}_{i+1} := \mathbf{a}_{i+1} - m\mathbf{a}_i$ and swaps $\mathbf{a}_i \leftrightarrow \mathbf{a}_{i+1}$, with m the nearest integer to $(\mathbf{a}_{i+1} \cdot \mathbf{b}_i)/|\mathbf{b}_i|^2$. After the swap, a partial Gram-Schmidt process is necessary in order to update the $\mathbf{b}_i$. Notice that $i < n$ because, if $i = n$ maximizes $2^i|\mathbf{b}_i|^2$, then there would be $j < n$ with $\mathbf{b}_j = \mathbf{0}$ and the algorithm would have terminated. For $n = 2$, the HJLS algorithm reduces to continued fractions, as explained below. If $\mathbf{x} = (a, b) \neq \mathbf{0}$, then[15]

$$\mathbf{b}_1 = \mathbf{a}_1 - \frac{\mathbf{a}_1 \cdot \mathbf{x}}{\mathbf{x} \cdot \mathbf{x}}\,\mathbf{x} = \left(\frac{b^2}{a^2 + b^2}, \frac{-ab}{a^2 + b^2}\right)$$

and so $\mathbf{b}_1 = \mathbf{0}$ if and only if $b = 0$. If $b = 0$, we find the syzygy $(0, 1)$ and we are done. Suppose $b \neq 0$, so $\mathbf{b}_2 = \mathbf{0}$ and $i = 1$ maximizes $2^i|\mathbf{b}_i|^2$ for $1 \leqslant i \leqslant 2$. If we start the main loop with $\mathbf{a}_1 = (c, d)$ and $\mathbf{a}_2 = (e, f)$, then

$$\mu = \frac{af - be}{ad - bc},$$

and $\mathbf{a}_1 = (e, f) - m(c, d)$ and $\mathbf{a}_2 = (c, d)$ for m the nearest integer to μ. This process constructs a sequence of $(c, d), (e, f), (e, f) - m(c, d), \dots$, as in Euclid's

[15] Recall that $\{\mathbf{a}_1, \mathbf{a}_2\}$ is the standard basis at the beginning of the algorithm.

algorithm, such that each item (c, d) represents a convergent c/d of the nearest-integer continued fraction expansion of a/b, which are like the ordinary continued fractions, but rounding toward the nearest integer instead of the floor function. If the algorithm terminates with $\mathbf{b}_1 = 0$, then $\mathbf{x}$ and $\mathbf{a}_1 = (c, d)$ are collinear, so $a/b = c/d$ with the r.h.s. in reduced terms, hence $(d, -c)$ is a syzygy of $\mathbf{x}$. If the algorithm terminates with $\mathbf{b}_1 = (c, d) \neq \mathbf{0}$ and $|\mathbf{b}_1| \leqslant 2^k$, then any syzygy of $\mathbf{x}$ will be longer than $(d, -c)$; compare this fact with the properties of best approximations of the convergents.

The HJLS article presents a few variants of the HJLS algorithm, which is called "Small Integer Relation Algorithm" and described in [HJLS89, §3]. The "Several Relations Algorithm" is described in [HJLS89, §4] and exploits [HJLS89, prop. 3.1, proof (1)], which gives $n - i$ independent syzygies if $\mathbf{b}_i = \mathbf{0}$. The "Simultaneous Relations Algorithm" is described in [HJLS89, §5] and generalizes the "Several Relations Algorithm" finding common syzygies of given vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l$. The Simultaneous Relations Algorithm performs the Gram-Schmidt orthogonalization on $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l, \mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n\}$, with exactly l zero vectors. In particular, the Simultaneous Relations Algorithm allows to compute syzygies of complex vectors, as explained in [HJLS89, p. 876, Rem. (i)]. The trick is the same used for the LLL algorithm: computing the simultaneous syzygies of the vectors of real and imaginary parts.

I shall expose a version of the last algorithm for computing a common syzygy for $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l$, which allows a few simplifications. Along the execution of the algorithm $\mathbf{b}_n = \mathbf{0}$, and when $\mathbf{b}_n \neq \mathbf{0}$, the algorithm terminates, as the original HJLS algorithm. Instead of choosing i maximizing $2^i|\mathbf{b}_i|^2$, this variant maximizes $2^{\tau_i}|\mathbf{b}_i|^2$. The exponent τ_i is defined in [HJLS89, §5] in two equivalent ways: $\tau_i^< = \#\{j : 1 \leqslant j < i,\ \mathbf{b}_j \neq \mathbf{0}\}$ and $\tau_i^\leqslant = \#\{j : 1 \leqslant j \leqslant i,\ \mathbf{b}_j \neq \mathbf{0}\}$. These two definitions of τ_i are equivalent because $\tau_i^\leqslant = \tau_i^< + 1$ except when $\mathbf{b}_i = \mathbf{0}$, in which case $2^{\tau_i^\leqslant}|\mathbf{b}_i|^2 = 2^{\tau_i^<}|\mathbf{b}_i|^2 = 0$ and i cannot maximize $2^{\tau_i}|\mathbf{b}_i|^2$. The choice of $\tau_i^\leqslant$ is consistent with the original HJLS algorithm, but I will chose $\tau_i^<$ because it is consistent with the proof of [HJLS89, thm. 5.1]. Once chosen i maximizing $2^{\tau_i}|\mathbf{b}_i|^2$, we reduce $\mathbf{a}_{i+1} := \mathbf{a}_{i+1} - m\mathbf{a}_i$ and swaps $\mathbf{a}_i \leftrightarrow \mathbf{a}_{i+1}$, with m the nearest integer to $(\mathbf{a}_{i+1} \cdot \mathbf{b}_i)/|\mathbf{b}_i|^2$, as in the original HJLS algorithm. The algorithm terminates when $\mathbf{b}_n \neq \mathbf{0}$ or $|\mathbf{b}_j| < 2^{-k}$ for $1 \leqslant j \leqslant n$, as the original HJLS algorithm.

The HJLS algorithm is said to be "extremely numerically unstable" in [FB92, §2], where it is speculated to derive from its reliance on the Gram-Schmidt orthogonalization process, whose numerical instability is invoked from [GVL96, §5.2.8]. Contrary, A. Meichsner says in his MSc thesis [Mei01] and insists in his PhD thesis [Mei09] that, if the HJLS algorithm is implemented "with full reductions" that he

defines, it is not different from PSLQ with $\gamma = \sqrt{2}$, and so they show the same numerical stability. He says that the reductions HJLS omits are redundant under exact arithmetic, but that this omission is crucial under inexact arithmetic, like ordinary numerics, and leads to numerical instability. In [HJLS89, p. 864] they admit this omission with the aim of improving the computational complexity, considering only exact arithmetic. Another version of HJLS, proposed in [RS94][16], is said to be numerically stable. Its authors attribute the numerical instability of the original HJLS to its approximation of the discontinuous function λ, where $\lambda(\mathbf{x})$ is the minimum length of a syzygy of $\mathbf{x}$. This is comparable to the numerical instability of the rank of a matrix. The problem with [RS94] is that it changes the goal of the algorithm. This is comparable with our change of the goal from the algebraic group to the eurymeric group. My experiments seem to show that the numerical instability is due to the implementation of the Gram-Schmidt process. A direct implementation of the HJLS algorithm uses the so-called classical Gram-Schmidt process, but implementing the so-called modified Gram-Schmidt process, see [GVL96, §5.2.8] for reference, showed in my experiments a numerical stability similar to PSLQ.

The implementation with effective real numbers has the usual issue of zero-testing. Contrary to ordinary Gram-Schmidt, the version used in HJLS needs to test if a given vector is zero. If, computed $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_i\}$, we find that $|\mathbf{b}_i \cdot \mathbf{b}_i| \leqslant \mathsf{tol}$, we make $\mathbf{b}_i := \mathbf{0}$ a proceed, because the computation of $\mathbf{b}_{i+1}$ requires a division by $\mathbf{b}_i \cdot \mathbf{b}_i$. For tol small enough, the vectors deemed zero are actually zero and the computation is exact. Else, we compute a fake syzygy or find erroneously that any possible syzygy is longer than 2^k. As the appropriate k is unknown, we use a general[17] global parameter S. It would be absurd to have $\mathsf{tol} > 2^{-k}$ so we assume $\mathsf{tol} \leqslant 2^{-\mathsf{S}}$. According to [HJLS89, thm. 5.1], the algorithm HJLS takes $\mathcal{O}(n^3(n+k))$ operations over $\mathbb{R}$ even for simultaneous syzygies.

The HJLS algorithm and its variants are explained in [HJLS89] for exact arithmetic in $\mathbb{R}$. Computing with effective real numbers requires, as happened with the LLL algorithm, an adapted proof of termination. I shall adapt [HJLS89, thm. 5.1] for effective real numbers. Instead of an additive tolerance ε as in Theorem 100, I shall use a multiplicative tolerance δ, lower than but as close to 1 as desired.

Theorem 101. *The HJLS algorithm for a common syzygy implemented in effective real numbers terminates in a finite number of steps, provided the multiplicative tolerance δ satisfies*

$$\frac{\sqrt{3}}{2} < \delta < 1. \tag{4.7}$$

[16] See also [RS95].

[17] See §2.5 for an introduction to general global parameters.

Proof. We construct the product $D = \alpha_1^{n-1}\alpha_2^{n-2}\cdots\alpha_{n-1}$ with $\alpha_i = \max\{2^n|\mathbf{b}_i|^2, 2^{-2k}\}$. Defined this way, $D \geqslant \prod_{i=1}^{n-1}(2^{-2k})^{n-i} = 2^{-kn(n-1)}$ throughout the algorithm. The reduction $\mathbf{a}_{i+1} := \mathbf{a}_{i+1} - m\mathbf{a}_i$ does not affect the Gram-Schmidt orthogonalization, and thus D. Noting with primes the values after a swap $\mathbf{a}_i \leftrightarrow \mathbf{a}_{i+1}$ of the HJLS algorithm, we shall prove

$$\frac{D'}{D} \leqslant \frac{3}{4\delta^2} < 1. \tag{4.8}$$

As i is chosen to maximize $2^{\tau_i}|\mathbf{b}_i|^2$ for $1 \leqslant i \leqslant n$, we have $\mathbf{b}_i \neq \mathbf{0}$ and thus $\tau_{i+1} = \tau_i + 1$, with exact arithmetic we would have $|\mathbf{b}_{i+1}|^2 \leqslant \frac{1}{2}|\mathbf{b}_i|^2$, but with effective real numbers we can only grant that $\delta^2|\mathbf{b}_{i+1}|^2 \leqslant \frac{1}{2}|\mathbf{b}_i|^2$. As the reduction $\mathbf{a}_{i+1} := \mathbf{a}_{i+1} - m\mathbf{a}_i$ with m the closest integer to $\mu_{i+1,i}$ is performed just before the swap $\mathbf{a}_i \leftrightarrow \mathbf{a}_{i+1}$, we would have $|\mu_{i+1,i}| \leqslant \frac{1}{2}$ with exact arithmetic, but with effective real numbers we can only grant $\delta|\mu_{i+1,i}| \leqslant \frac{1}{2}$. As $\mathbf{b}_i' = \mathbf{b}_{i+1} + \mu_{i+1,i}\mathbf{b}_i$, instead of HJLS's (3.2), we have

$$|\mathbf{b}_i'|^2 = |\mathbf{b}_{i+1}|^2 + \mu_{i+1,i}^2\,|\mathbf{b}_i|^2 \leqslant \frac{1}{2\delta^2}\,|\mathbf{b}_i|^2 + \left(\frac{1}{2\delta}\right)^2 |\mathbf{b}_i|^2 = \frac{3}{4\delta^2}\,|\mathbf{b}_i|^2. \tag{4.9}$$

Hence $|\mathbf{b}_i'|^2 \leqslant |\mathbf{b}_i|^2$, so

$$\alpha_i' \leqslant \alpha_i. \tag{4.10}$$

As the algorithm has not terminated, with exact arithmetic we would have $|\mathbf{b}_j| > 2^{-k}$ for certain $j \geqslant 1$, but with effective real numbers we can only grant $|\mathbf{b}_j| > 2^{-k}\delta$. Also, instead of $2^{\tau_i}|\mathbf{b}_i|^2 \geqslant 2^{\tau_j}|\mathbf{b}_j|^2$, we have $2^{\tau_i}|\mathbf{b}_i|^2 \geqslant 2^{\tau_j}\delta|\mathbf{b}_j|^2$. So, instead of HJLS's (3.3), we have

$$2^n|\mathbf{b}_i|^2 \geqslant 2^{\tau_i+1}|\mathbf{b}_i|^2 \geqslant 2^{\tau_j+1}\delta|\mathbf{b}_j|^2 > 2^{\tau_j+1-2k}\delta^3 \geqslant 2^{-2k+1}\delta^3. \tag{4.11}$$

Hence, as (4.7) implies $2\delta^3 > 1$, we have $2^n|\mathbf{b}_i|^2 \geqslant 2^{-2k}$, so

$$\alpha_i = 2^n|\mathbf{b}_i|^2. \tag{4.12}$$

As $\mathbf{a}_{i+1}' = \mathbf{a}_i$, we have that $\mathbf{b}_i$ and $\mathbf{b}_{i+1}'$ are orthogonal projections of the same vector onto different subspaces. As the subspace defining $\mathbf{b}_i$ contains the subspace defining $\mathbf{b}_{i+1}'$, $|\mathbf{b}_i| \geqslant |\mathbf{b}_{i+1}'|$, so we get HJLS's (3.4)

$$\alpha_i \geqslant \alpha_{i+1}'. \tag{4.13}$$

Let $V = \mathbb{R}\mathbf{b}_{1-l} + \mathbb{R}\mathbf{b}_{2-l} + \cdots + \mathbb{R}\mathbf{b}_{i-1}$. The projections of $\mathbf{a}_i = \mathbf{a}_{i+1}'$ and $\mathbf{a}_{i+1} = \mathbf{a}_i'$ orthogonal to V are $\mathbf{b}_i$ and $\mathbf{b}_i'$ respectively. The projection of $\mathbf{b}_i'$ orthogonal to $\mathbf{b}_i$ is $\mathbf{b}_{i+1}$. The projection of $\mathbf{b}_i$ orthogonal to $\mathbf{b}_i'$ is $\mathbf{b}_{i+1}'$. Thus,

$$|\mathbf{b}_i \cdot \mathbf{b}_i'| = |\mathbf{b}_i|\,|\mathbf{b}_{i+1}| = |\mathbf{b}_i'|\,|\mathbf{b}_{i+1}'|. \tag{4.14}$$

Recall that, by definition, $\alpha_{i+1} \geqslant 2^{-2k}$ and $\alpha_{i+1} \geqslant 2^n|\mathbf{b}_{i+2}|^2$. If $\alpha'_i = 2^{-2k}$, then

$$\frac{\alpha'_i\alpha'_{i+1}}{\alpha_i\alpha_{i+1}} = \frac{2^{-2k}\alpha'_{i+1}}{\alpha_i\alpha_{i+1}} \overset{(4.13)}{\leqslant} \frac{2^{-2k}\alpha_i}{\alpha_i 2^{-2k}} = 1.$$

If $\alpha'_{i+1} = 2^{-2k}$, then

$$\frac{\alpha'_i\alpha'_{i+1}}{\alpha_i\alpha_{i+1}} = \frac{\alpha'_i 2^{-2k}}{\alpha_i\alpha_{i+1}} \overset{(4.10)}{\leqslant} \frac{\alpha_i 2^{-2k}}{\alpha_i 2^{-2k}} = 1.$$

If $\alpha'_i = 2^n|\mathbf{b}'_i|^2$ and $\alpha'_{i+1} = 2^n|\mathbf{b}'_{i+1}|^2$, then $|\mathbf{b}'_i|\,|\mathbf{b}'_{i+1}| \neq 0$ and, by (4.14), $|\mathbf{b}_i|\,|\mathbf{b}_{i+1}| \neq 0$, so

$$\frac{\alpha'_i\alpha'_{i+1}}{\alpha_i\alpha_{i+1}} \overset{(4.12)}{=} \frac{2^n|\mathbf{b}'_i|^2 2^n|\mathbf{b}'_{i+1}|^2}{2^n|\mathbf{b}_i|^2\alpha_{i+1}} \leqslant \frac{2^n|\mathbf{b}'_i|^2 2^n|\mathbf{b}'_{i+1}|^2}{2^n|\mathbf{b}_i|^2 2^n|\mathbf{b}_{i+1}|^2} \overset{(4.14)}{=} 1.$$

Thus, gathering the three cases, we recover HJLS's (3.5)

$$\frac{\alpha'_i\alpha'_{i+1}}{\alpha_i\alpha_{i+1}} \leqslant 1. \tag{4.15}$$

Finally, we have

$$\frac{D'}{D} = \frac{(\alpha'_i)^{n-i}(\alpha'_{i+1})^{n-i-1}}{\alpha_i^{n-i}\alpha_{i+1}^{n-i-1}} \overset{(4.15)}{\leqslant} \frac{\alpha'_i}{\alpha_i} \overset{(4.12)}{=} \frac{\max\{2^n|\mathbf{b}'_i|^2, 2^{-2k}\}}{2^n|\mathbf{b}_i|^2} = \max\left\{\frac{2^n|\mathbf{b}'_i|^2}{2^n|\mathbf{b}_i|^2}, \frac{2^{-2k}}{2^n|\mathbf{b}_i|^2}\right\}$$
$$\overset{(4.11)}{\leqslant} \max\left\{\frac{|\mathbf{b}'_i|^2}{|\mathbf{b}_i|^2}, \frac{2^{-2k}}{2^{-2k+1}\delta^3}\right\} \overset{(4.9)}{\leqslant} \max\left\{\frac{3}{4\delta^2}, \frac{1}{2\delta^3}\right\} \overset{(4.7)}{=} \frac{3}{4\delta^2} \overset{(4.7)}{<} 1,$$

which proves our claim (4.8). As $D \geqslant 2^{-kn(n-1)}$ at any stage and the upper bound of the factor of decreasing of D at a swap, as the swap is the only basis transformation in the HJLS algorithm that modifies D, imply that the HJLS algorithm performs only a finite number of swaps and thus a finite number of iterations, so the HJLS algorithm terminates in a finite number of steps. □

4.4.5 About the PSLQ algorithm

Another algorithm for computing additive syzygies is PSLQ, presented in [FB92] and described in [FBA99]. In the name PSLQ, PS is after "partial sums" and LQ after the LQ factorization, both distinctive ingredients of the PSLQ algorithm. The algorithm starts with a vector $\mathbf{x} \in \mathbb{R}^n$ whose syzygies we look for. In the framework of effective real numbers, if $|\mathbf{x}|$ is too close to zero, we terminate saying

that $\mathbf{x} = \mathbf{0}$ and thus any element of $\mathbb{Z}^n$ is a syzygy. In the case of computing the minimal polynomial of $\alpha \in \mathbb{R}$, $\mathbf{x} = (1, \alpha, \ldots, \alpha^d)^\intercal$ is always nonzero because of its first entry. If $|\mathbf{x}|$ is not too close to zero, we normalize $\mathbf{x} := \mathbf{x}/|\mathbf{x}|$, so we may assume that $\mathbf{x} = (x_1, x_2, \ldots, x_n)^\intercal$ has unit length. We assume also that $x_n \neq 0$; otherwise $(0, \ldots, 0, 1)$ would be a syzygy.

The aforesaid "partial sums" $s_1, s_2, \ldots, s_n$ are defined in the following way: $s_i = \sqrt{x_i^2 + x_{i+1}^2 + \cdots + x_n^2}$. Notice that no s_i is zero because $x_n \neq 0$. With these numbers we construct the matrix

$$\mathtt{H} := \begin{pmatrix} \frac{s_2}{s_1} & & & \\ -\frac{x_1 x_2}{s_1 s_2} & \frac{s_3}{s_2} & & \\ -\frac{x_1 x_3}{s_1 s_2} & -\frac{x_2 x_3}{s_2 s_3} & \ddots & \\ \vdots & \vdots & \ddots & \frac{s_n}{s_{n-1}} \\ -\frac{x_1 x_n}{s_1 s_2} & -\frac{x_2 x_n}{s_2 s_3} & \cdots & -\frac{x_{n-1} x_n}{s_{n-1} s_n} \end{pmatrix}.$$

The columns of this matrix form an orthonormal basis of $\mathbb{R}^n$ together with $\mathbf{x}$. I know two ways to generate these column vectors from $\mathbf{x}$. The first method is the Gram-Schmidt orthonormalization process starting with $\mathbf{x}$ and following with the standard basis of $\mathbb{R}^n$. This is similar to HJLS, but here we use the Gram-Schmidt orthonormalization, rather than the plain orthogonalization. The second method produces the columns of $\mathtt{H}$ from right to left. For $n = 2$, it is usual to take $(x_n, -x_{n-1})$ as an orthogonal vector. This second method extends this approach to $n > 2$. We start by taking a vector orthogonal to $\mathbf{x}$ in $\mathbf{0}^{n-2} \times \mathbb{R}^2$. The next step is to take a vector orthogonal to both in $\mathbf{0}^{n-3} \times \mathbb{R}^3$, and so on. If we choose the unit vectors with the first nonzero component positive, the process generates the columns of $\mathtt{H}$ from right to left.

Another ingredient in the PSLQ algorithm is the Hermite reduction. In the same way the LLL reduction is Gram-Schmidt with coefficients rounded to the nearest integer, the Hermite reduction is Gaussian reduction with coefficients rounded to the nearest integer. According to [Str10, §2.2], if $\mathtt{H}$ is a lower-trapezoidal matrix with nonzero entries on the diagonal, there exists a unique matrix $\mathtt{D}_0$ reducing by rows that yields $\mathtt{D}_0\mathtt{H}$ the diagonal of $\mathtt{H}$. This matrix $\mathtt{D}_0$ is lower triangular with ones in the diagonal. The matrix $\mathtt{D}$ is the nearest-integer roundoff of $\mathtt{D}_0$. As [FBA99, dfn. 3] says, there are recursive formulae for the entries of $\mathtt{D}$ where some coefficients depend on others, so it must be solved in a particular order. In [FBA99, dfn. 4] they give an iterative algorithm for the Hermite reduction. This is a kind of Gaussian reduction of $\mathtt{H}$ in a particular order and rounding off the

coefficients m_{ij} in the reductions $\mathtt{H}_{i*} := \mathtt{H}_{i*} - m_{ij}\mathtt{H}_{j*}$. The order of these reductions would not matter if there were no rounding off, but there is rounding off and this order minimizes the impact of the nonzero reminders.

I shall present a slightly modified version of the PSLQ algorithm. It starts with the matrices $\mathtt{H}$ defined above, $\mathtt{y} := \mathbf{x}^\intercal$ and $\mathtt{B} := \mathtt{I}_n$. With their initial values $\mathtt{yH} = \mathtt{0}$ and $\mathtt{y} = \mathbf{x}^\intercal\mathtt{B}$. We want to keep these equalities along the algorithm. For each transformation $\mathtt{L}$ on the rows of $\mathtt{H}$, we have $(\mathtt{yL}^{-1})(\mathtt{LH}) = \mathtt{0}$, so we redefine $\mathtt{y} := \mathtt{yL}^{-1}$, $\mathtt{B} := \mathtt{BL}^{-1}$ and $\mathtt{H} := \mathtt{LH}$. A transformation $\mathtt{Q}$ on the columns of $\mathtt{H}$ affects only to $\mathtt{H}$, so we only redefine $\mathtt{H} := \mathtt{HQ}$. As we did in the HJLS algorithm, keeping the matrices $\mathtt{L}$ elementary, their inverse $\mathtt{L}^{-1}$ are immediate. The matrices $\mathtt{L}$ in the PSLQ algorithm are only elementary components of the Hermite reduction and row swaps. The matrices $\mathtt{Q}$ in the PSLQ algorithm are orthogonal, which grants some properties to $\mathtt{H}$ along the process.

The process starts with the Hermite reduction of $\mathtt{H}$. Then, we choose i maximizing $\gamma^i\mathtt{H}_{ii}$, with $\gamma > 2/\sqrt{3}$ a parameter of the PSLQ algorithm, and swap $\mathtt{H}_{i*} \leftrightarrow \mathtt{H}_{i+1,*}$. This reminds the HJLS algorithm with $\gamma = \sqrt{2}$. Indeed, as cited above, Meichsner claims that HJLS is equivalent to PSLQ with $\gamma = \sqrt{2}$. The interchange of rows may break the trapezoidal shape of $\mathtt{H}$, so the next step is an orthogonal transformation in its columns restoring the trapezoidal shape. This orthogonal transformation is a rotation in the plane spanned by the columns i and $i+1$, for the same i used for swapping the rows. In the case $i = n-1$, the trapezoidal shape is kept and thus no orthogonal transformation is performed. The next step is the Hermite reduction, and then we exit if there is any null component in $\mathtt{y}$. In this case, if $\mathtt{y}_{1j} = 0$, then $\mathtt{B}_{*j}$ is a syzygy of $\mathbf{x}$. If $\max_j |\mathtt{H}_{jj}| < 1/M$, where $M > 0$ is an input parameter of the algorithm, then it is granted that any syzygy of $\mathbf{x}$ is longer than M, and we stop. If the algorithm does not terminate with a positive or negative answer, we return to the interchange step.

The PSLQ algorithm takes polynomial time. According to [FBA99, thm. 3], if it gives a syzygy $\mathbf{m}$ and $\mathbf{m}_0$ is the shortest (non-null) syzygy, then $|\mathbf{m}| \leqslant \gamma^{n-2}|\mathbf{m}_0|$.

There is an error in the original PSLQ algorithm, described in [Heß11, p. 203]. If x_{n-1} is an integer multiple of x_n, then $\mathtt{H}_{n,n-1}$ is an integer multiple of $\mathtt{H}_{n-1,n-1}$, so $\mathtt{H}_{n,n-1} = 0$ after the first Hermite reduction. If $n = 2$ or γ is large enough, the i maximizing $\gamma^i|\mathtt{H}_{ii}|$ is $i = n-1$, so we swap rows i and $i+1$, obtaining $\mathtt{H}_{n-1,n-1} = 0$. There is no orthogonal transformation, thus Hermite reduction is performed with $\mathtt{H}_{n-1,n-1} = 0$. The solution I propose is checking before the Hermite reduction if its pivots $\mathtt{H}_{ii}$ are nonzero. According to [FBA99, lem. 5], if $\mathtt{H}_{ii} = 0$ for certain i, then $i = n-1$, $\mathtt{y}_j = 0$ for certain j and $\mathtt{B}_{*j}$ is a syzygy of $\mathbf{x}$. So, if the null $\mathtt{H}_{ii}$ is $i = n-1$,

then we find a syzygy or abort due to poor precision. If $\mathtt{H}_{ii} = 0$ for $i < n-1$, then we abort due to poor precision. Another feature of the implementation with effective real numbers is that $\mathsf{tol} < 1/M$. As the appropriate M is unknown, we use a general[18] global parameter M.

As we saw for LLL and HJLS algorithms, the proof of termination of the PSLQ algorithm [FBA99, §4] is proved for exact arithmetic and needs to be adapted for effective real numbers. This proof uses the parameters γ and ρ and defines

$$\tau = \sqrt{\frac{1}{\rho^2} + \frac{1}{\gamma^2}}. \tag{4.16}$$

The parameter ρ is introduced in [FBA99, lem. 4] for bounding the error of rounding a real/complex/quaternion number to the nearest ordinary/Gaussian/Hamiltonian integer, considering all the cases simultaneously with ρ equal to 2, $\sqrt{2}$ or 1 respectively. The two sources of effective-numeric error that may affect the termination of the PSLQ algorithm are the following. First, we have to maximize $\gamma^i|\mathtt{H}_{ii}|$, so that, instead of $|\alpha| \geqslant \gamma|\lambda|$, we have $|\alpha| \geqslant (\gamma-\varepsilon)|\lambda|$ for a tolerance ε. The other source of effective-numeric error is rounding $\mathtt{H}_{ij}/\mathtt{H}_{jj}$ to the nearest integer. So, instead of $|\alpha| > \rho|\beta|$, we have $|\alpha| > (\rho-\varepsilon)|\beta|$. Thus, the PSLQ runs with γ and $\rho = 2$, but the proof needs $\gamma' = \gamma - \varepsilon$ and $\rho' = 2 - \varepsilon$. So, instead of τ defined in (4.16), the termination of the algorithm is ruled by

$$\tau' = \sqrt{\frac{1}{(\rho-\varepsilon)^2} + \frac{1}{(\gamma-\varepsilon)^2}}.$$

With these considerations, it seems possible to adapt the proof of termination of the PSLQ algorithm [FBA99, §4] in a similar manner as the proof of termination of the LLL algorithm was adapter for effective real numbers in §4.4.3.

The case $n = 2$ is simpler and reduces to the extended Euclidean algorithm. Let $\mathbf{x} = (a, b)^\intercal$ with $b > 0$, so $\mathtt{H} = \begin{pmatrix} b \\ -a \end{pmatrix}$. The Hermite reduction is equivalent to the reduction performed in Euclid's algorithm, both in $\mathtt{y}$ and in $\mathtt{H}$. The swap in $\mathtt{H}$ and $\mathtt{y}$ is another ingredient of Euclid's algorithm. There is no rotation because the i maximizing $\gamma^i|\mathtt{H}_{ii}|$ is $i = 1 = n-1$. If a and b are commensurable, then we will eventually find a zero in $\mathtt{H}$ and another in $\mathtt{y}$. In this case the matrix $\mathtt{B}$ keeps track of the transformations, as in the extended Euclidean algorithm. The column of $\mathtt{B}$ corresponding to the null element of $\mathtt{y}$ is a syzygy of $\mathbf{x}$. The other column of $\mathtt{B}$ corresponds to the nonzero element of $\mathtt{y}$, which is the g.c.d. of a and b, and it

[18] See §2.5 for an introduction to general global parameters.

yields Bézout's identity. If a and b are incommensurable, then the elements of $\mathtt{H}$ and of $\mathtt{y}$ become smaller and smaller, converging to zero, eventually $|\mathtt{H}_{11}| < 1/M$ and we stop with a negative answer.

The PSLQ algorithm was conceived for finding the syzygies of a single real vector, so the trick of looking for the syzygies common to the real and imaginary part of a complex vector is not applicable in principle, but it may be adapted for syzygies of vectors over the complexes and even over the quaternions. This adaptation is very straightforward, and [FBA99] exposes the algorithm in a general form valid for the three fields, but PSLQ over the complexes outputs relations over the Gaussian integers, and PSLQ over the quaternions outputs relations over the Hamiltonian integers, rather than relations over the integers. There are some adaptations of PSLQ for common syzygies, like [Mei09, §2.5], [CFQZ10a] and [CFQZ10b], so we may use the trick for computing integer relations even for complex input.

4.4.6 Comparison of these algorithms

I have exposed three algorithms that can be used to get additive syzygies: LLL, HJLS and PSLQ. The first of them is not an algorithm for syzygies but can be adapted as explained in §4.4.2, whilst the other two are actual algorithms for computing syzygies. Together with a guess of the degree of the minimal polynomial of a number to reconstruct symbolically, which is implemented as a general global parameter D, the three algorithms need a second general global parameter, as described in the previous sections. Such a second general global parameter, except C of LLL, is a bound on the size of the syzygy to return, so that, if no syzygy is found below this bound, the algorithm returns the bound instead of the syzygy. The general global parameter H of LLL is a bound on the supremum norm, whist the bounds used in HJLS and PSLQ are bound on the Euclidean norm. The LLL algorithm with the general global parameter C fails to get a bound on the size of a possible syzygy, but the other methods also need a second general global parameter, so this is not a disadvantage in our context.

The three algorithms run in polynomial time under exact arithmetic, but a complexity analysis seems difficult under the effective real numbers because we would need to know the precision necessary for the algorithm to give an answer. The current fast implementations of the PSLQ algorithm are in a similar situation, according to [vH12], since they compute with some different precisions and a complexity analysis would also need to know the precision necessary for the algorithm to succeed. The HJLS is said in [FB92] (the report where PSLQ was introduced)

to be numerically instable, but there are some proposals to avoid such an instability. My experiments with these two algorithms seem to show that PSLQ is faster than HJLS, but I tested a naive implementation of HJLS against an optimized implementation of PSLQ. According to M. van Hoeij [vH12], LLL is empirically faster than PSLQ and this speed of LLL is founded in both theoretical and practical advances, like [NSV11a] (with an extended abstract in [NSV11b]). Under effective-numerical arithmetic, as far as I know, there is no study of these algorithms, and because of that I provided proofs of termination of these algorithms.

Meichsner proved in his thesis that HJLS and PSLQ are essentially equivalent, and there is another perspective of both algorithms in [CSV13], based on the more general setting of intersections of lattices and vector spaces. Notice that the syzygies of $\mathbf{x}$ are precisely $\mathbb{Z}^n \cap \mathbf{x}^\perp$.

4.5 The main algorithm

Let G be the differential Galois group of the differential equation or system Δ of order r and size n. We look for a Singerian solution, which is a solution y whose line $\mathbb{K}y$ is invariant by G° and has at most $I(rn)$ images by G, where I is the function of Theorem 33. Let me sketch the main algorithm.

1. According to §2.2.4, for each singularity z of Δ, based at a near regular point z_0, compute both a subspace of solutions V_z containing the Singerian solutions and the action of the local Galois group on V_z.

2. Move, by analytic continuation, all these local groups to a common regular point defined over $\mathbb{K}$, computing both the intersection V where all of them are defined and the action of G on it.

3. Take the generators of G and apply the algorithm of §3.8, computing a truncation (in the sense described in §3.8.1 and §3.8.2) of a group H greater than G (its eurymeric closure, described in §3.3) but such that the Singerian solutions are invariant by H°.

4. Compute the common eigenspaces of H°, pick a non-null vector defined over $\mathbb{K}$ and compute its orbit by H.

5. Reconstruct symbolically the solutions and check if they are solutions of Δ in the way explained below.

- If we find a non-null solution, we have succeeded.
- If we find that the space containing the Singerian solutions is zero, we say that zero is the only Liouvillian solution of Δ.
- If the space is nonzero but our candidate solutions are wrong, then we have computed at poor precision and then we restart the algorithm at finer precision.

If the algorithm terminates with the first possibility, as the candidate solution was checked, it is a nonzero Liouvillian solution of Δ. According to §3.8.2, the errors in linear algebra may lead to an undercomputation of H, so to an overcomputation of the subspace of Singerian solutions, but never to an overcomputation of H and an undercomputation of the subspace of Singerian solutions. Thus, if the algorithm terminates with the second possibility, then it is granted that zero is the only Liouvillian solution of Δ. The third possibility may only happen a finite number of times, and the algorithm terminates with the first or the second possibilities, depending on whether there is a nonzero Liouvillian solution or not, as it will be proved.

The key idea is (as expressed in the proof of [vdH07a, thm. 8]) to consider the execution trace of the algorithm with oracles for the general global parameters, excluding the tolerance, and another oracle for the zero-test. In this execution trace there are finitely many exact zero-tests, so finitely many nonzero numbers to be deemed nonzero by the zero-test. If we fix a tolerance less than the modulus of any of these nonzero numbers, the approximate zero-test will perform the same as the exact one. Notice that such a performance of the algorithm under exact arithmetic and oracles is correct and never ends with the third possibility.

Remark 102. An *oracle* is a black box that decides the membership of a language with no computational cost, a *language* being an arbitrary set of finite strings over an *alphabet*, which is an arbitrary nonempty finite set. For instance, an oracle for the zero-test decides the membership of the language of the codes of the null effective complex numbers. Using the device of [Rog67, p. 347, ll. 13–17], an oracle may compute a function with domain a language and image a countable set, like integers, with no computational cost. For example, for an oracle for the degree of an algebraic number, the domain would be the language of the codes of the algebraic effective complex numbers.

The expression "finer precision" refers to all the general global parameters, including the tolerance via $\mathsf{tol} = 2^{-\mathsf{T}}$, as introduced in §2.5. The simplest idea is to increase all the general global parameters simultaneously, each parameter following a succession diverging to infinity, as J. van der Hoeven proposes in [vdH07a, §3.4],

but this way is not granted to be correct. For example, the method for computing syzygies with the LLL algorithm explained in §4.4.2 that uses the general global parameter C and Proposition 97 is not granted to work if C grows together with the general global parameter D, since each new value $\mathsf{D} = d$ poses a new lattice with a corresponding set $\mathcal{C}_d$ of the successful values C for $\mathsf{D} = d$ and $\inf \mathcal{C}_d$ might grow faster than C. This issue can be avoided by using another method for computing syzygies among the algorithms exposed in §4.4, but the tolerance is harder to be kept growing linked to another general global parameter. Each time a general global parameter increases, it poses a new problem with a new execution trace of the algorithm with and oracle for zero-test, as explained above, and thus a new bound for the tolerance in order to follow the correct trace.

Let me consider the toy example of the aforesaid method for computing syzygies with the LLL algorithm, which uses the general global parameter C and Proposition 97, with oracles for the rest of general global parameters, excluding D. Let us consider successions $(c_k)_{k=0}^{\infty}$ and $(d_k)_{k=0}^{\infty}$ diverging to infinity for C and D respectively. As shown above, the pair (C, D) may take the values (c_k, d_k) and fail. My proposal is to start with (c_0, d_0), to follow with (c_1, d_0) and (c_0, d_1), and so on, following by diagonals $\{(c_i, d_j) : i + j = k\}$. As only a finite amount of numbers will be reconstructed, there exists an index i such that $\mathsf{D} = d_i$ works with an oracle for C. According to Proposition 97, there exists an index j such that $(\mathsf{C}, \mathsf{D}) = (c_j, d_i)$ works. We may observe that, though in the proof one parameter is clearly subordinated to the other, their role in the algorithm is interchangeable.

The cautious way would be to make the general global parameters follow a generalization of this diagonal argument to many variables. If we have the general global parameters $\mathsf{X}_1, \mathsf{X}_2, \ldots, \mathsf{X}_s$, with divergent successions of proposed values $(x_{1k})_{k=0}^{\infty}, (x_{2k})_{k=0}^{\infty}, \ldots, (x_{sk})_{k=0}^{\infty}$, any enumeration of the $(s-1)$-simplex $\{(x_{1i_1}, x_{2i_2}, \ldots, x_{si_s}) : i_1 + i_2 + \cdots + i_s = k\}$ would work. If two or many parameters are proved to safely grow simultaneously, they can be merged and treated as a single one in the previous scheme.

Let me compile the list of the general global parameters introduced in this thesis:

- K (in §2.2.4) for bounding how many terms in the principal part of the Laurent expansion of a function must be zero-tested before you can deem it zero,
- G (in §3.8.2) for bounding how far the linearized Derksen–van der Hoeven algorithm (of §3.6) must be carried out before truncating,

- T (in §2.5) for treating the tolerance tol as a general global parameter by taking $\mathsf{tol} = 2^{-\mathsf{T}}$,
- B (in §??) for bounding the degree of numerator and denominator in Padé approximation,
- D (in §4.4) for bounding the degree of the algebraic numbers to reconstruct symbolically,
- C (in §4.4.2) for allowing the LLL method for syzygies to work properly,
- H (in §4.4.2), S (in §4.4.4) and M (in §4.4.5) for bounding the size of the syzygies.

Notice that all these parameters have an ideal value, given by their respective oracle, but the algorithm also works with any greater value.

Let us consider the execution trace of the first step of the main algorithm with oracles for the zero-test of numbers and for the order of a power series. There is only a finite number of functions to test for the order of their principal part at their respective singularity, so it is safe to take for K any bound of all these orders. For G we may take any bound on the number of iterations of the Dersen–van der Hoeven algorithm, but in practice the algorithm may work with a much smaller value, since it is not necessary to compute a complete set of representatives of H/H°. In a similar manner to what we did for K, for B it is valid any bound of the degree of numerator and denominator of all the rational functions to reconstruct in the algorithm trace under oracle-aided exact computation. As a greater value of B yields the same reconstructed rational function but not necessarily the same numerator and denominator, as explained in Remark 86, we need to consider now the trace of the algorithm under exact computation except for the parameter B. In this trace, we reconstruct a finite amount of algebraic numbers, so we may take for D any bound of their degrees. If we follow the method for syzygies that uses C, their choice and dependence of D (and, thus, of B) is explained above. If we follow another method, as there is only a finite amount of syzygies to compute, we may take for H, S or M any bound of the size of these syzygies. Let us now consider the trace of the algorithm with valid values of all the general global parameters, except the tolerance, and an oracle for the zero-test. As described above, we may take a tolerance smaller than the modulus of any of these nonzero numbers to test. This completes the proof that, for precision fine enough, the algorithm performs the same as under exact computation. As such a performance of the algorithm is correct, the algorithm terminates, so this proves termination and correction of the main algorithm, summarized in the following theorem.

Theorem 103. *The main algorithm described in this section terminates with a nonzero Liouvillian solution, if such a solution exists, or with the statement that zero is the only Liouvillian solution if this is the case.*

This is not the most efficient version of the algorithm, but a simple one for explanation and proof. I shall introduce some devices for speeding up the algorithm in §4.6.1.

4.6 Final remarks

4.6.1 Devices to speed up the algorithm

Using the notation of the previous section, G is the differential Galois group and H its eurymeric closure, $\mathcal{M}$ is the finite set of generator of G as an algebraic group, $\mathcal{F}$ and $\mathfrak{a}$ are the auxiliary objects in the algorithm of §3.6, the candidate of representatives of H/H° and the candidate Lie algebra $\mathfrak{a}$, and V is the candidate space of Singerian solutions.

In principle we should write down H/H°, which may be very large, but we may avoid it by some devices. Instead of carrying the space V along the process and computing the invariant lines at the very end, we may restrict V to the subspace spanned by the joint eigenvectors of $\mathfrak{a}$. This way allows an early termination when the answer is negative and may reduce the working dimension. In any case there is no need to store the explicit system of representatives of H/H°, only its generators, computing the orbit of an H°-invariant line by successive application of the generators, until the set is saturated.

Another device, which helps the previous one, is to compute, apart of systematically the products $\mathtt{AB}$, commutators $\mathtt{ABA}^{-1}\mathtt{B}^{-1}$ as in [vdH07a, §4.6] and random elements as in [BBR93]. This way we expect to get quickly a matrix to add to $\mathfrak{a}$ that reduces the candidate subspace V. This device is complemented by multiplying the random product of generators by a random element of $\mathfrak{a}\cap\mathrm{GL}$. Notice that a resonance of eigenvalues of $\mathtt{A}\in\mathrm{GL}$ is either common to $\mathtt{A}(\mathfrak{a}\cap\mathrm{GL})$ or particular of a proper subvariety thereof, so the eigenvalues of a random element of $\mathtt{A}(\mathfrak{a}\cap\mathrm{GL})$ will have only common resonances almost surely. Moreover, if we consider only the resonances of order $I(rn)$ at most, the subset of $\mathtt{A}(\mathfrak{a}\cap\mathrm{GL})$ of matrices with non-common resonances is a proper algebraic subvariety.

This restricted V splits in direct sum $V_1 \oplus V_2 \oplus \cdots \oplus V_s$ of joint eigenspaces of $\mathfrak{a}$. This splitting defines a block structure such that $\mathfrak{a}$ is the algebra of diagonal scalar-by-blocks matrices. For the elements of $\mathcal{F}$, the macrostructure by blocks is a permutation of the V_i spaces. If they are not transitively permuted, we may restrict V to each subspace spanned by the orbits of the permutation. In this case the problem is restricted to several problems of lower dimension. Notice that each V_i of the same orbit has the same dimension.

If $\mathtt{M} \in \mathcal{F}$ keeps each V_i invariant, then $\mathtt{M}$ is block-diagonal and the problem restricted to each V_i is the problem of $\mathfrak{a} = \mathbb{K}\mathtt{I}$, which is the hardest, but in a lower dimension. If we do not want to split the problem, we may consider only intra-block resonances and discard inter-block resonances. We can multiply $\mathtt{M}$ by a diagonal scalar-by-blocks matrix in order to get a substitute for $\mathtt{M}$ such that any eigenvalue of its $(i+1)$-th block has greater module than any eigenvalue of its i-th block, avoiding inter-block resonances.

If $\mathtt{M} \in \mathcal{F}$ permutes transitively (therefore cyclically) the spaces V_i, we will find only generic resonances. Indeed, any substitute for $\mathtt{M}$ obtained multiplying $\mathtt{M}$ by a diagonal scalar-by-blocks matrix $\mathrm{diag}(\lambda_1\mathtt{I}, \lambda_2\mathtt{I}, \ldots, \lambda_s\mathtt{I})$ has the same characteristic polynomial as $\mu\mathtt{M}$ with $\mu^s = \lambda_1 \cdots \lambda_s$. The proof is easy but a bit technical.

If $\mathtt{M} \in \mathcal{F}$ permutes the spaces V_i, grouping the orbits together, we have a super-block structure and we may apply the aforesaid device in order to consider only intra-super-block resonances and discard inter-super-block resonances. Inside each super-block, we are in the transitive case, thus we cannot discard resonances.

With this structure by blocks, testing if a given matrix belongs to a given component of the group is easier than solving linear systems. We need to classify the elements of $\mathcal{F}$ modulo $\mathfrak{a} \cap \mathrm{GL}(V)$. The first classifying parameter is the permutation of V_i. For the same permutation, the comparable blocks are proportional, but the factor may be different for each block.

It is well known[19] that the worst case for computing lines invariant by G° is when the group generated by $\mathcal{M}$ is finite, and its eurymeric closure correspond to a finite subgroup of PGL, because these finite groups might be very large and we do not know if eventually will appear a generator to put in the Lie algebra $\mathfrak{a}$ or not. If such a group is reducible, by Maschke's Theorem, it is completely reducible, so we can split the problem into two smaller ones.

In the case corresponding to a finite subgroup of $\mathrm{PGL}(V)$, there exists a Her-

[19] See [Cor01] for a detailed study for dimension up to 5.

mitian metric on V invariant by G, and so by H. Moreover, if the action of G (and thus of H) on V is irreducible, the vector space of sesquilinear forms on V invariant by G (and therefore by H) has dimension 1 over $\mathbb{C}$. This is an easy variant of [FH91, Exer. 1.14]. If the computed dimension is higher it might be a symptom of overcomputed dimension or of reducibility. If the actual dimension is m, V splits as direct sum of m irreducible subspaces. If the computed dimension is zero, it means that H does not correspond to a finite subgroup of $\mathrm{PGL}(V)$.

In the case of the Ramis generators of Galois group, we have privileged information we may use. We are granted that exponential tori and Stokes multipliers belong to the identity component. The monodromy around a singularity z_0 raised to the p-th power, where p is the ramification index at z_0, belongs to the identity component. Instead of initializing $\mathcal{F}$ with Ramis generators and $\mathfrak{a} = \mathbb{K}\mathtt{I}$, we may initialize $\mathcal{F}$ with monodromy matrices at points with non-trivial ramification and $\mathfrak{a}$ with the algebra generated by exponential (and potential) tori. Notice that Stokes multipliers are the identity on V.

4.6.2 Open questions

The study of the computational complexity is an open question. Not knowing the precision necessary for finding an answer, it is hard to bound the complexity. It is similar to the multilevel implementation of the PSLQ algorithm, whose computational complexity, according to [vH12], is unknown precisely because we do not know beforehand the precision necessary for finding an answer.

The algorithm of this thesis, in the case of non-null Liouvillian solutions, returns such a solution, but only one. An open question is to extend this algorithm to compute a complete basis of the space of Liouvillian solutions. As observed in §1.5.4, the d'Alembert reduction method yields an equation defined no longer over $\mathbb{C}(x)$, but over an algebraic extension thereof. This would require to extend the algorithm of this thesis from the Riemann sphere to a finite ramified covering thereof. These ideas will be developed in further work.

HERE ENDETH THE FOWERTH CHAPTER ✠

Index

Bibliography

[Abe1824] Niels H. Abel, 1824. *Mémoire sur les équations algébriques où on démontre l'impossibilité de la résolution de l'équation générale du cinquième dégré*. Groendahl, Christiania [Oslo]. Cited twice, on pages 6 and 17.

[AH1897] Archimedes and T. L. Heath, 1897. *The works of Archimedes*. Cambridge University Press. Cited once, on page 61.

[AOG05] Archimedes and Paloma Ortiz Garcia, 2005. *Tratados II*. No. 378 in Biblioteca Clásica Gredos. Editorial Gredos, Madrid. Cited once, on page 61.

[Arm06] Deborah J. Armstrong, 2006. «The quarks of object-oriented development.» *Commun. ACM*, 49(2):123–128. Cited once, on page 63.

[Bak75] George A. Baker Jr., 1975. *Essentials of Padé approximants*. Academic Press. Cited twice, on pages 112 and 114.

[Bal94] Werner Balser, 1994. *From divergent power series to analytic functions*. No. 1582 in Lecture Notes in Mathematics. Springer-Verlag. Cited 4 times, on pages 43 and 46.

[Bal00] ———, 2000. *Formal power series and linear systems of meromorphic ordinary differential equations*. Universitext. Springer-Verlag. Cited once, on page 75.

[Bar89] M. A. Barkatou, 1989. *Contribution à l'étude des équations différentielles et aux différences dans le champ complexe*. Thèse, Institut National Polytechnique de Grenoble, France. Cited once, on page 74.

[Bar97] ———, 1997. «An algorithm to compute the exponential part of a formal fundamental matrix solution of a linear differential system.»

Applicable Algebra in Engineering, Communication and Computing, 8(1):1–23. Cited once, on page 75.

[BB01] David H. Bailey and David J. Broadhurst, 2001. «Parallel integer relation detection: techniques and applications.» *Mathematics of Computation*, 70(236):1719–1736. Cited once, on page 120.

[BBR93] Lázló Babai, Robert Bealsl and Daniel Rockmore, 1993. «Deciding finiteness of matrix groups in deterministic polynomial time.» In *Proccedings of ISSAC'93*. pp. 117–126. Cited once, on page 144.

[BBRS91] Werner Balser, Boele L. J. Braaksma, Jean-Pierre Ramis and Yasutaka Sibuya, 1991. «Multisummability of formal power series solutions of linear ordinary differential equations.» *Asymptotic Analysis*, 5(1):27–45. Cited once, on page 43.

[BCEB09] Moulay A. Barkatou, Thomas Cluzeau and Carole El Bacha, 2009. «Algorithms for regular solutions of higher-order linear differential systems.» In *Proceedings of the ISSAC'09*. Seoul, Korea, pp. 7–14. Cited once, on page 33.

[BCEB11] ———, 2011. «Simple forms of higher-order linear differential systems and their applications in computing regular solutions.» *Journal of Symbolic Computation*, 46(6):633–658. Cited once, on page 33.

[Ber80] George M. Bergman, 1980. «Notes on Ferguson and Forcade's generalized Euclidean algorithm.» Unpublished notes, http://math.berkeley.edu/~gbergman/papers/unpub/FF_Euc.pdf, University of California, Berkeley. Cited twice, on page 130.

[Beu99a] Frits Beukers, 1999. «Factorisation of polynomials.» In Cohen et al. [CCS99], pp. 78–90. Cited once, on page 124.

[Beu99b] ———, 1999. «Lattice reduction.» In Cohen et al. [CCS99], pp. 66–77. Cited 5 times, on pages 124, 127, and 129.

[BGM81] George A. Baker Jr. and Peter Graves-Morris, 1981. *Padé approximants. Part I: Basic theory*, vol. 13 of *Encyclopedia of Mathematics and its Applications*. Addison-Wesley. Cited thrice, on pages 112 and 117.

[BGY80] Richard P. Brent, Fred G. Gustavson and David Y. Y. Yun, 1980. «Fast solution of Toeplitz systems of equations and computation of Padé approximants.» *Journal of Algorithms*, 1(3):259–295. Cited 6 times, on pages 117 and 118.

[BHM02] Peter Borwein, Kevin G. Hare and Alan Meichsner, 2002. «Reverse symbolic computations, the identify function.» In *Proceedings from the Maple Summer Workshop*. Maple Software, Waterloo, Canada. Cited once, on page 120.

[BM91] Carl B. Boyer and Uta C. Merzbach, 1991. *A History of Mathematics*. John Wiley. Revised reprint of the second edition. Cited twice, on pages 6 and 17.

[Bor91] Armand Borel, 1991. *Linear algebraic groups*. No. 126 in Graduate Texts in Mathematics, 2nd ed. Springer-Verlag. Cited 5 times, on pages 84, 86, and 87.

[Bor02] Peter Borwein, 2002. *Computational excursions in analysis and number theory*. No. 10 in CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC. Springer-Verlag, New York. Cited twice, on pages 124 and 130.

[Bre11] Emmanuel Breuillard, 2011. «An exposition of Jordan's original proof of his theorem on finite subgroups of $\mathrm{GL}_n(\mathbb{C})$.» Available at http://www.math.u-psud.fr/~breuilla/Jordan.pdf. Notes from the lecture "Jordan's theorem on finite linear groups and its approximate analogues" given at the Isaac Newton Institute (University of Cambridge) available at http://sms.cam.ac.uk/media/1093825. Cited once, on page 53.

[Buc67] J. D. Buckholtz, 1967. «Sums of powers of complex numbers.» *Journal of Mathematical Analysis and Applications*, 17:269–279. Cited once, on page 69.

[Cal82] Jacques Calmet, ed., 1982. *Computer algebra*, vol. 144 of *Lecture Notes in Computer Science*. Springer-Verlag. Cited once, on page 154.

[Car1545] G. Cardano, 1545. *Ars Magna, sive de regulis algebraicis*. Nuremberg, Germany. Cited twice, on pages 6 and 17.

[Car02] Matthew C. Cary, 2002. «Lattice basis reduction: Algorithms and applications.» http://uqu.edu.sa/files2/tiny_mce/plugins/filemanager/files/4281667/hamdy/hamdy1/lattice.pdf. Cited twice, on page 124.

[CCS99] Arjeh M. Cohen, Hans Cuypers and Hans Sterk, eds., 1999. *Some tapas of computer algebra*, vol. 4 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin. Cited twice, on page 151.

[CFQZ10a] Jing-wei Chen, Yong Feng, Xiao-lin Qin and Jing-zhong Zhang, 2010. «Simultaneous integer relation detection and its an application.» ArXiv:0906.4917v2. Cited once, on page 139.

[CFQZ10b] Jingwei Chen, Yong Feng, Xiaolin Qin and Jingzhong Zhang, 2010. «Detecting simultaneous integer relations for several real vectors.» ArXiv:1010.1982v1. Cited once, on page 139.

[Cij74a] P. L. Cijsouw, 1974. «Transcendence measures of certain numbers whose transcendency was proved by A. Baker.» *Compositio Mathematica*, 28:179–194. Cited once, on page 126.

[Cij74b] ———, 1974. «Transcendence measures of exponentials and logarithms of algebraic numbers.» *Compositio Mathematica*, 28:163–178. Cited once, on page 126.

[CK00] R. C. Churchill and Jerald J. Kovacic, 2000. «Cyclic vectors.» In *Proceedings [GCKS02] of Differential Algebra and Related Topics*. New Jersey, pp. 191–218. Cited twice, on page 31.

[CL72] Earl A. Coddington and Norman Levinson, 1972. *Theory of ordinary differential equations*. Tata McGraw-Hill, New Delhi. Cited once, on page 58.

[Coh93] Henri Cohen, 1993. *A course in computational algebraic number theory*, vol. 138 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin. Cited once, on page 124.

[Col07] Michael J. Collins, 2007. «On Jordan's theorem for complex linear groups.» *Journal of Group Theory*, 10(4):411–423. Cited once, on page 52.

[Col08] ———, 2008. «Bounds for finite primitive complex linear groups.» *Journal of Algebra*, 319(2):759–776. Cited once, on page 53.

[Cor01] Olivier Cormier, 2001. *Résolution des équations différentielles linéaires d'ordre 4 et 5: applications à la théorie de Galois classique*. Ph.D. thesis, Université de Rennes I, France. Cited 5 times, on pages 8, 19, 54, 59, and 145.

[CR62] Charles W. Curtis and Irving Reiner, 1962. *Representation theory of finite groups and associative algebras*. Wiley Interscience. Cited twice, on pages 52 and 53.

[CSV13] Jingwei Chen, Damien Stehlé and Gilles Villard, 2013. «A new view on HJLS and PSLQ: sums and projections of lattices.» In *Proceedings of ISSAC '13*. pp. 149–156. Cited once, on page 140.

[DDT82] J. Della Dora, Cl. Di Crescenzo and E. Tournier, 1982. «An algorithm to obtain formal solutions of a linear homogeneous differential equation at an irregular singular point.» In *[Cal82], Proceedings of the EUROCAM'82*. Marseille, France, pp. 273–280. Cited once, on page 74.

[DJK05] Harm Derksen, Emmanuel Jeandel and Pascal Koiran, 2005. «Quantum automata and algebraic groups.» *Journal of Symbolic Computation*, 39(3–4):357–371. Cited 4 times, on pages 87 and 88.

[Dor71] Larry Dornhoff, 1971. *Group representation theory. Part A: Ordinary representation theory*. Pure and Applied Mathematics. Marcel Dekker, New York. Cited once, on page 52.

[DT81] J. Della Dora and E. Tournier, 1981. «Formal solutions of differential equations in the neighborhood of singular points (regular and irregular).» In *Proceedings of the SYMSAC'81*. Snowbird, Utah, pp. 25–29. Cited once, on page 74.

[Dwo98] Cynthia Dwork, 1998. «Lecture notes on lattices and their applications to cryptography.» Lecture notes, Standford University. http://www.cmm.uchile.cl/~mkiwi/topicos/00/dwork-lattice-lectures.ps. Cited 4 times, on pages 123 and 124.

[EH1908a] Euclides and T. L. Heath, 1908. *The thirteen books of Euclid's Elements translated from the text of Heiberg. Vol. II: Books III–IX*. Cambridge University Press. Cited once, on page 80.

[EH1908b] ———, 1908. *The thirteen books of Euclid's Elements translated from the text of Heiberg. Vol. III: Books X–XIII and Appendix*. Cambridge University Press. Cited once, on page 80.

[EP94] Euclides and M. L. Puertas Castaños, 1994. *Elementos. Libros V–IX*. No. 191 in Biblioteca Clásica Gredos. Editorial Gredos, Madrid. Cited once, on page 80.

[EP96] ———, 1996. *Elementos. Libros X–XIII*. No. 228 in Biblioteca Clásica Gredos. Editorial Gredos, Madrid. Cited once, on page 80.

[Eul1760] Leonhard Euler, 1760. «De seriebus divergentibus.» *Novi Commentarii academiae scientiarum Petropolitanae*, 5:205–237. Cited once, on page 43.

[Fab1885] Eugène Fabry, 1885. *Sur les intégrales des équations différentielles linéaires à coefficients rationnels*. Thèse de doctorat, Faculté des Sciences, Paris. Published by Gauthier-Villars. Cited once, on page 33.

[Fab1888] ———, 1888. «Réductibilité des équations différentielles linéaires.» *Bulletin de la Société Mathématique de France*, 16:135–142. NumDAM: BSMF_1888__16__135_1. Cited once, on page 34.

[FB92] Helaman R. P. Ferguson and David H. Bailey, 1992. «A polynomial time, numerically stable integer relation algorithm.» Tech. rep., RNR Technical Report RNR-91-032. http://crd.lbl.gov/~dhbailey/dhbpapers/pslq.pdf. Cited 4 times, on pages 127, 132, 135, and 139.

[FBA99] Helaman R. P. Ferguson, David H. Bailey and Steve Arno, 1999. «Analysis of PSLQ, an integer relation finding algorithm.» *Mathematics of Computation*, 68(225):351–369. Cited 10 times, on pages 120, 135, 136, 137, 138, and 139.

[FF79] H. R. P. Ferguson and R. W. Forcade, 1979. «Generalization of the Euclidean algorithm for real numbers to all dimensions higher than two.» *Bull. AMS. New Series*, 1(6):912–914. Cited once, on page 130.

[FH91] William Fulton and Joe Harris, 1991. *Representation theory: A first course*. No. 129 in Graduate Texts in Mathematics. Springer-Verlag. Cited twice, on pages 54 and 146.

[Fro1881] G. Frobenius, 1881. «Ueber Relationen zwischen den Näherungsbrüchen von Potenzreihen.» *Journal für die reine und angewandte Mathematik*, 90:1–17. Cited once, on page 112.

[Gal1846] Évariste Galois, 1846. «Mémoire sur les conditions de résolubilité des équations par radicaux.» *Journal de mathématiques pures et appliquées*, 11:417–433. Cited twice, on pages 6 and 17.

[Gau1799] Karl F. Gauss, 1799. *Demonstratio nova theorematis omnem functionem algebraicam rationalem integram unius variabilis in factores reales primi vel secundi gradus resolvi posse*. Ph.D. thesis, Academia Julia Carolina, Helmstedt, Germany. Cited twice, on pages 6 and 17.

[Gau1801] ———, 1801. *Disquisitiones Arithmeticae*. Fleischer, Leipzig. Cited once, on page 123.

[GCKS02] Li Guo, Phyllis J. Cassidy, William F. Keigher and William Y. Sit, eds., 2002. *Differential Algebra and Related Topics*. World Scientific. Cited once, on page 153.

[GKZ94] I. M. Gelfand, M. M. Kapranov and A. V. Zelevinsky, 1994. *Discriminants, resultants, and multidimensional determinants*. Mathematics: Theory & Applications. Birkhäuser, Boston. Cited twice, on page 111.

[Gra72] W. B. Gragg, 1972. «The Padé table and its relation to certain algorithms of numerical analysis.» *SIAM Review*, 14(1):1–16. Cited once, on page 116.

[GVL96] Gene H. Golub and Charles F. Van Loan, 1996. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences, 3rd ed. Johns Hopkins University Press, Baltimore. Cited twice, on pages 132 and 133.

[Hal03] Brian C. Hall, 2003. *Lie groups, Lie algebras, and representations*, vol. 222 of *Graduate Texts in Mathematics*. Springer-Verlag, New York. An elementary introduction. Cited once, on page 86.

[Han10] Guillaume Hanrot, 2010. «LLL: A tool for effective Diophantine approximation.» In Nguyen and Vallée [NV10], pp. 215–263. Cited once, on page 124.

[Har49] G. H. Hardy, 1949. *Divergent Series*. Oxford University Press. Cited once, on page 46.

[Hen96] Peter Anne Hendriks, 1996. *Algebraic Aspects of Linear Differential and Difference Equations*. Ph.D. thesis, Rijksuniversiteit Groningen, The Netherlands. Http://irs.ub.rug.nl/ppn/153769580. Cited thrice, on pages 35, 120, and 157.

[Heß11] Hermann Hessling, 2011. «On the Euler scale and the μEuclidean integer relation algorithm.» *Journal of Mathematics and Music*, 5(3):195–215. Cited once, on page 137.

[HHLS86] Johan Hastad, Bettina Helfrich, J. C. Lagarias and Claus-Peter Schnorr, 1986. «Polynomial time algorithms for finding integer relations among real numbers.» In Burkhard Monien and Guy Vidal-

Naquet, eds., *Proceedings of STACS'86*, vol. 210 of *Lecture Notes in Computer Science*. Springer, pp. 105–118. Cited thrice, on page 130.

[Hil87] Abdelaziz Hilali, 1987. *Solutions formelles de systèmes différentiels linéaires au voisinage d'un point singulier*. Thèse d'etat, Université Joseph-Fourier, Grenoble, France. Cited once, on page 74.

[HJLS89] Johan Hastad, Bettina Just, J. C. Lagarias and Claus-Peter Schnorr, 1989. «Polynomial time algorithms for finding integer relations among real numbers.» *SIAM Journal on Computing*, 18(5):859–881. Cited 20 times, on pages 130, 131, 132, and 133.

[Hru02] Ehud Hrushovski, 2002. «Computing the Galois group of a linear differential equation.» In *Differential Galois theory (Będlewo, 2001)*, vol. 58 of *Banach Center Publ.* Polish Acad. Sci., Warsaw, pp. 97–138. Cited twice, on pages 8 and 19.

[HSW68] W. A. Harris Jr., Y. Sibuya and L. Weinberg, 1968. «A reduction algorithm for linear differential systems.» *Funkcialaj Ekvacioj*, 11:59–67. Cited once, on page 32.

[Hum81] James E. Humphreys, 1981. *Linear Algebraic Groups*. No. 21 in Graduate Texts in Mathematics. Springer-Verlag. Cited once, on page 84.

[HvdP95] Peter A. Hendriks and Marius van der Put, 1995. «Galois action on solutions of a differential equation.» *Journal of Symbolic Computation*, 19(6):559–576. Included in [Hen96]. Cited twice, on pages 35 and 120.

[HW75] G. H. Hardy and E. M. Wright, 1975. *An introduction to the theory of numbers*. 4th ed. Oxford University Press. Cited thrice, on pages 80 and 81.

[IEE08] IEEE 754-2008, 2008. *Standard for Floating-Point Arithmetic*. Institute of Electrical and Electronics Engineers. Cited once, on page 62.

[ISO11a] ISO/IEC 14882:2011, 2011. *Programming Languages: C++*. International Organization for Standardization. Cited once, on page 63.

[ISO11b] ISO/IEC 9899:2011, 2011. *Programming Languages: C*. International Organization for Standardization. Cited once, on page 61.

[Jor1877] Camille JORDAN, 1877. «Mémoire sur les équations différentielles linéaires à intégrale algébrique.» *Journal für die reine und angewandte Mathematik*, 84:89–215. Cited once, on page 52.

[JS98] Antoine JOUX and Jacques STERN, 1998. «Lattice reduction: A toolbox for the cryptanalyst.» *Journal of Cryptology*, 11(3):161–185. Cited thrice, on pages 120, 125, and 127.

[Jus90] Bettina JUST, 1990. «Integer relations among algebraic numbers.» *Mathematics of Computation*, 54(189):467–477. Cited once, on page 127.

[Jus92] ———, 1992. «Generalizing the continued fraction algorithm to arbitrary dimensions.» *SIAM Journal on Computing*, 21(5):909–926. Cited once, on page 131.

[Kap76] Irving KAPLANSKY, 1976. *An Introduction to Differential Algebra*. Actualités Scientifiques et Industrielles, 2nd ed. Hermann, Paris. Cited twice, on pages 37 and 51.

[KLL88] R. KANNAN, A. K. LENSTRA and L. LOVÁSZ, 1988. «Polynomial factorization and nonrandomness of bits of algebraic and some transcendental numbers.» *Mathematics of Computation*, 50(181):235–250. Cited 6 times, on pages 120, 125, and 126.

[Klü10] Jürgen KLÜNERS, 2010. «The van Hoeij algorithm for factoring.» In NGUYEN and VALLÉE [NV10], pp. 283–291. Cited once, on page 124.

[Knu81a] Donald E. KNUTH, 1981. *The art of computer programming. Vol. 2: Seminumerical algorithms*. 2nd ed. Addison-Wesley. Cited once, on page 80.

[Knu81b] ———, 1981. «Supernatural numbers.» In David A. KLARNER, ed., *The Mathematical Gardner*. Wadsworth International, pp. 310–325. Cited once, on page 62.

[Kol48] E. R. KOLCHIN, 1948. «Algebraic matric groups and the Picard-Vessiot theory of homogeneous linear ordinary differential equations.» *Annals of Mathematics. Second Series*, 49(1):1–42. The spelling "matric" appears in the original. Cited 4 times, on pages 10 and 20.

[Kov86] Jerald J. KOVACIC, 1986. «An algorithm for solving second order linear homogeneous differential equations.» *Journal of Symbolic Computation*, 2(1):3–43. Cited 6 times, on pages 8, 19, 58, and 97.

[Lan02] Serge Lang, 2002. *Algebra*, vol. 211 of *Graduate Texts in Mathematics*. 3rd ed. Springer-Verlag, New York. Cited once, on page 26.

[Len08] Hendrik W. Lenstra Jr., 2008. «Lattices.» In J. P. Buhler and P. Stevenhagen, eds., *Algorithmic number theory: lattices, number fields, curves and cryptography*, vol. 44 of *Mathematical Sciences Research Institute Publications*. Cambridge University Press, pp. 127–181. Cited once, on page 129.

[Lio1835] Joseph Liouville, 1835. «Mémoire sur l'intégration d'une classe de fonctions transcendantes.» *Journal für die reine und angewandte Mathematik*, 13:93–118. Cited 6 times, on pages 7 and 18.

[Lio1839] ———, 1839. «Mémoire sur l'intégration d'une classe d'équations différentielles du second ordre en quantités finies explicites.» *Journal de mathématiques pures et appliquées*, 4:423–456. Cited 4 times, on pages 9 and 20.

[LLL82] A. K. Lenstra, H. W. Lenstra Jr. and L. Lovász, 1982. «Factoring polynomials with rational coefficients.» *Mathematische Annalen*, 261(4):515–534. Cited 12 times, on pages 120, 124, 125, and 130.

[LR94] Michèle Loday-Richaud, 1994. «Stokes phenomenon, multisummability and differential Galois groups.» *Annales de l'Institut Fourier*, 44(3):849–906. Ref. NUMDAM AIF_1994__44_3_849_0. Cited once, on page 47.

[Luc10] Manuel J. Lucena López, 2010. «Criptografía y seguridad en computadores.» http://wwwdi.ujaen.es/~mlucena, 4th ed., v. 0.8.1. Cited once, on page 61.

[Mag97] Andy R. Magid, 1997. *Lectures on Differential Galois Theory*. No. 7 in University Lecture Series. American Mathematical Society. Cited 5 times, on pages 36, 37, and 38.

[Mei01] Alan Meichsner, 2001. *Integer Relation Algorithms and the Recognition of Numerical Constants*. MSc thesis, Simon Fraser University, Canada. http://www.collectionscanada.ca/obj/s4/f2/dsk3/ftp04/MQ61592.pdf. Cited 5 times, on pages 122, 124, 126, 130, and 132.

[Mei09] ———, 2009. *The Integer Chebyshev Problem: Computational Explorations*. Ph.D. thesis, Simon Fraser University, Canada. http:

//summit.sfu.ca/item/9413. Cited 6 times, on pages 122, 124, 126, 130, 132, and 139.

[MG02] Daniele Micciancio and Shafi Goldwasser, 2002. *Complexity of lattice problems: A cryptographic perspective*. No. 671 in The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Boston. Cited twice, on page 122.

[Mos60] Jürgen Moser, 1960. «The order of a singularity in Fuchs' theory.» *Mathematische Zeitschrift*, 72:379–398. Cited once, on page 75.

[MP12] J. M. McNamee and Victor Y. Pan, 2012. «Efficient polynomial root-refiners: a survey and new record efficiency estimates.» *Computers and Mathematics with Applications*, 63(1):239–254. Cited once, on page 68.

[MR91] Jean Martinet and Jean-Pierre Ramis, 1991. «Elementary acceleration and multisummability I.» *Annales de l'Institut Henri Poincaré* (A) *Physique Théorique*, 54(4):331–401. Cited 7 times, on pages 46, 47, 48, and 50.

[MR92] Bernard Malgrange and Jean-Pierre Ramis, 1992. «Fonctions multisommables.» *Annales de l'Institut Fourier*, 42(1-2):353–368. Ref. NUMDAM AIF_1992__42_1-2_353_0. Cited once, on page 47.

[MR99] Juan J. Morales Ruiz, 1999. *Differential Galois theory and non-integrability of Hamiltonian systems*. No. 179 in Progress in Mathematics. Birkhäuser Verlag, Basel. Cited once, on page 92.

[MRR01] Juan J. Morales-Ruiz and Jean-Pierre Ramis, 2001. «Galoisian obstructions to integrability of Hamiltonian systems I.» *Methods and Applications of Analysis*, 8(1):33–95. Cited once, on page 101.

[MRRS07] Juan J. Morales Ruiz, Jean-Pierre Ramis and Carles Simó, 2007. «Integrability of Hamiltonian systems and differential Galois groups of higher variational equations.» *Annales Scientifiques de l'École Normale Supérieure. Série 4*, 40(6):845–884. Cited once, on page 101.

[Ngu08] K. A. Nguyen [Nguyen An Khuong], 2008. *A modern perspective on Fano's approach to linear differential equations*. Ph.D. thesis, Rijksuniversiteit Groningen, the Netherlands. Http://irs.ub.rug.nl/ppn/314759115. Cited twice, on page 92.

[Ngu10] Phong Q. Nguyen, 2010. «Hermite's constant and lattice algorithms.» In Nguyen and Vallée [NV10], pp. 19–69. Cited once, on page 123.

[Nov08] Andy Novocin, 2008. *Factoring Univariate Polynomials over the Rationals*. Ph.D. thesis, Florida State University. http://diginole.lib.fsu.edu/etd/2515. Cited twice, on page 124.

[NSV11a] Andrew Novocin, Damien Stehlé and Gilles Villard, 2011. «An LLL-reduction algorithm with quasi-linear time complexity.» HAL report ensl-00534899 v. 2, ENS de Lyon, France. Cited once, on page 140.

[NSV11b] ———, 2011. «An LLL-reduction algorithm with quasi-linear time complexity: extended abstract.» In *STOC '11: Proceedings of the 43rd annual ACM symposium on Theory of computing*. ACM, New York, pp. 403–412. Cited once, on page 140.

[NV10] Phong Q. Nguyen and Brigitte Vallée, eds., 2010. *The LLL algorithm: Survey and applications*. Information Security and Cryptography. Springer-Verlag, Berlin. Cited 4 times, on pages 156, 158, 161, and 164.

[NvdPT08] K. A. Nguyen, M. van der Put and J. Top, 2008. «Algebraic subgroups of $\mathrm{GL}_2(\mathbb{C})$.» *Indagationes Mathematicae (New Series)*, 19(2):287–297. Cited twice, on page 92.

[Pad1892] Henri Padé, 1892. «Sur la représentation approchée d'une fonction par des fractions rationnelles.» *Annales scientifiques de l'École Normale Supérieure (Sér. 3)*, 9(suppl.):S3–S93. Cited once, on page 114.

[Pan97] Victor Y. Pan, 1997. «Solving a polynomial equation: some history and recent progress.» *SIAM Review*, 39(2):187–220. Cited once, on page 68.

[Pan00] ———, 2000. «Approximating complex polynomial zeros: modified Weyl's quadtree construction and improved Newton's iteration.» *Journal of Complexity*, 16(1):213–264. Cited once, on page 68.

[Pet09] Karin Peter, 2009. *The LLL-Algorithm and some Applications*. Bachelor thesis, ETH Zurich. Cited twice, on pages 120 and 124.

[PH1876] Pappus and F. Hultsch, 1876. *Pappi Alexandrini Collectionis quae supersunt. Volumen I: librorum II, III, IV et V reliquiae*. Weidmann, Berlin. E libris manu scriptis edidit, latina interpretatione et commentariis. Cited twice, on pages 6 and 17.

[Plo95] Simon Plouffe, 1995. «The inverse symbolic calculator.» http://oldweb.cecm.sfu.ca/projects/ISC/. Cited once, on page 120.

[PT30] Pappus and W. Thomson, 1930. *The Commentary of Pappus on Book X of Euclid's Elements, Arabic text and translation.* Harvard University Press. Cited once, on page 80.

[Ram04] Jean-Pierre Ramis, 2004. «Gevrey asymptotics and applications to holomorphic ordinary differential equations.» In Hua Chen and Roderick S. C. Wong, eds., *Differential Equations and Asymptotic Theory in Mathematical Physics*, vol. 2 of *Series in Analysis*. World Scientific, pp. 44–99. Cited once, on page 47.

[Ris68] Robert H. Risch, 1968. *The problem of integration in finite terms.* Ph.D. thesis, University of California, Berkeley. Cited twice, on pages 7 and 18.

[Ris69] ———, 1969. «The problem of integration in finite terms.» *Transactions of the American Mathematical Society*, 139:167–189. Cited twice, on pages 7 and 18.

[Ris70] ———, 1970. «The solution of the problem of integration in finite terms.» *Bulletin of the American Mathematical Society*, 76:605–608. Cited twice, on pages 7 and 18.

[Rit48] Joseph Fels Ritt, 1948. *Integration in Finite Terms. Liouville's Theory of Elementary Methods.* Columbia University Press, New York. Cited twice, on pages 7 and 18.

[RM90] J.-P. Ramis and J. Martinet, 1990. «Théorie de Galois différentielle et resommation.» In E. Tournier, ed., *Computer algebra and differential equations*, Computational Mathematics and Applications. Academic Press, pp. 117–214. Cited twice, on pages 47 and 49.

[Rog67] Hartley Rogers Jr., 1967. *Theory of recursive functions and effective computability.* McGraw-Hill Book Co., New York. Cited once, on page 141.

[Rom07] Darya Romanova, 2007. «Integer relations among real numbers.» Lecture notes, Joint Advanced Student School, Course 1: Polynomials: Their Power and How to Use Them. http://www14.informatik.tu-muenchen.de/konferenzen/Jass07/courses/1/Romanova/Romanova_Paper.pdf. Cited once, on page 130.

[RS94] Carsten RÖSSNER and Claus P. SCHNORR, 1994. «A stable integer relation algorithm.» Tech. Rep. TR-94-016, International Computer Science Institute, Berkeley, California. http://ftp.icsi.berkeley.edu/ftp/pub/techreports/1994/tr-94-016.pdf. Cited twice, on page 133.

[RS95] Carsten RÖSSNER and Claus-P. SCHNORR, 1995. «Computation of highly regular nearby points.» In *Third Israel Symposium on the Theory of Computing and Systems (Tel Aviv, 1995)*. IEEE Comput. Soc. Press, Los Alamitos, CA, pp. 174–181. Cited once, on page 133.

[Sch1872] H. A. SCHWARZ, 1872. «Ueber diejenigen Fälle, in welchen die Gaussische hypergeometrische Reihe eine algebraische Function ihres vierten Elementes darstellt.» *Journal für die reine und angewandte Mathematik*, 75:292–335. Cited twice, on pages 8 and 18.

[Sch1897] Ludwig SCHLESINGER, 1897. *Handbuch der Theorie der linearen Differentialgleichungen (Band 2, Theil 1)*. Teubner, Leipzig. Cited once, on page 39.

[Sch82] Arnold SCHÖNHAGE, 1982. «The fundamental theorem of algebra in terms of computational complexity.» Preliminary report, Mathematisches Institut der Universität Tübingen. www.iai.uni-bonn.de/~schoe/fdthmrep.ps.gz. Cited once, on page 68.

[Ser92] Jean-Pierre SERRE, 1992. *Lie algebras and Lie groups*, vol. 1500 of *Lecture Notes in Mathematics*. 2nd ed. Springer-Verlag. Cited once, on page 86.

[Sin81] Michael F. SINGER, 1981. «Liouvillian solutions of n-th order homogeneous linear differential equations.» *American Journal of Mathematics*, 103(4):661–682. Cited 7 times, on pages 8, 19, 51, 52, 55, and 59.

[Sma86] Steve SMALE, 1986. «Newton's method estimates from data at one point.» In Richard E. EWING, Kenneth I. GROSS and Clyde F. MARTIN, eds., *The merging of disciplines: new directions in pure, applied, and computational mathematics*. Springer, pp. 185–196. Cited once, on page 69.

[Ste05] Damien STEHLÉ, 2005. *Algorithmique de la réduction de réseaux et application à la recherche de pires cas pour l'arrondi de fonctions mathématiques*. Ph.D. thesis, Université Henri Poincaré, Nancy,

France. Available at http://perso.ens-lyon.fr/damien.stehle/thesis.html together with the source of the examples cited in the thesis. Cited once, on page 128.

[Ste10] ———, 2010. «Floating-point LLL: Theoretical and practical aspects.» In Nguyen and Vallée [NV10], pp. 179–213. Cited twice, on page 128.

[Sto1902] G. G. Stokes, 1902. «On the discontinuity of arbitrary constants that appear as multipliers of semi-convergent series.» *Acta Mathematica*, 26(1):393–397. Cited once, on page 47.

[Str10] Armin Straub, 2010. «A gentle introduction to PSLQ.» http://arminstraub.com/files/pslq.pdf. Cited once, on page 136.

[SU93a] Michael F. Singer and Felix Ulmer, 1993. «Galois group of second and third order linear differential equations.» *Journal of Symbolic Computation*, 16(1):9–36. Cited twice, on pages 8 and 19.

[SU93b] ———, 1993. «Liouvillian and algebraic solutions of second and third order linear differential equations.» *Journal of Symbolic Computation*, 16(1):37–73. Cited twice, on pages 8 and 19.

[SU97] ———, 1997. «Linear differential equations and products of linear forms.» *Journal of Pure and Applied Algebra*, 117/118:549–563. Algorithms for algebra (Eindhoven, 1996). Cited once, on page 111.

[Tou87] Evelyne Tournier, 1987. *Solutions formelles d'équations différentielles*. Thèse d'État, Université Joseph-Fourier, Grenoble, France. Cited once, on page 74.

[Tur55] H. L. Turrittin, 1955. «Convergent solutions of ordinary linear homogeneous differential equations in the neighborhood of an irregular singular point.» *Acta Mathematica*, 93:27–66. Cited once, on page 74.

[Tur75] Paul Turán, 1975. «Power sum method and the approximative solution of algebraic equations.» *Mathematics of Computation*, 29(129):311–318. Cited thrice, on page 69.

[TY05] Patrice Tauvel and Rupert W. T. Yu, 2005. *Lie algebras and algebraic groups*. Springer Monographs in Mathematics. Springer-Verlag, Berlin. Cited 5 times, on pages 84, 86, 93, and 94.

[UW96] Felix Ulmer and Jacques-Arthur Weil, 1996. «Note on Kovacic's algorithm.» *Journal of Symbolic Computation*, 22(2):179–200. Cited thrice, on pages 8, 19, and 59.

[Vaz01] Vijay V. Vazirani, 2001. *Approximation algorithms*. Springer-Verlag, Berlin. Cited 4 times, on pages 123 and 124.

[vdH99] Joris van der Hoeven, 1999. «Fast evaluation of holonomic functions.» *Theoretical Computer Science*, 210(1):199–215. Cited 6 times, on pages 9, 12, 20, 23, and 72.

[vdH01] ———, 2001. «Fast evaluation of holonomic functions near and in regular singularities.» *Journal of Symbolic Computation*, 31(6):717–743. Cited 4 times, on pages 9, 12, 20, and 23.

[vdH05] ———, 2005. «Effective analytic functions.» *Journal of Symbolic Computation*, 39(3-4):433–449. Cited thrice, on pages 9, 20, and 73.

[vdH06a] ———, 2006. «Computations with effective real numbers.» *Theoretical Computer Science*, 351(1):52–60. Cited 5 times, on pages 9, 19, 62, and 64.

[vdH06b] ———, 2006. «Effective real numbers in Mmxlib.» In *Proceedings of ISSAC '06*. Genoa, Italy, pp. 138–145. Cited twice, on page 62.

[vdH07a] ———, 2007. «Around the numeric-symbolic computation of differential Galois groups.» *Journal of Symbolic Computation*, 42(1-2):236–264. Cited 26 times, on pages 8, 9, 12, 19, 20, 23, 38, 47, 82, 87, 88, 89, 100, 106, 112, 118, 119, 120, 141, and 144.

[vdH07b] ———, 2007. «Efficient accelero-summation of holonomic functions.» *Journal of Symbolic Computation*, 42(4):389–428. Cited 5 times, on pages 9, 12, 20, 23, and 33.

[vdPS03] Marius van der Put and Michael F. Singer, 2003. *Galois theory of linear differential equations*. No. 328 in Grundlehren der Mathematischen Wissenschaften. Springer-Verlag. Cited 8 times, on pages 26, 30, 37, 38, 47, and 50.

[Ves1892] Ernest Vessiot, 1892. «Sur l'intégration des quations différentielles linéaires.» *Annales Scientifiques de l'École Normale Supérieure. Série 3*, 9:197–280. Cited twice, on pages 10 and 20.

[vH96] Mark van Hoeij, 1996. *Factorization of Linear Differential Operators*. Ph.D. thesis, Katholieke Universiteit Nijmegen, the Netherlands. Cited once, on page 74.

[vH12] ———, 2012. «PSLQ is no longer state of the art.» Personal communication. Cited thrice, on pages 139, 140, and 146.

[vHW97] Mark van Hoeij and Jacques-Arthur Weil, 1997. «An algorithm for computing invariants of differential Galois groups.» *Journal of Pure and Applied Algebra*, 117-118:353–379. Algorithms for algebra (Eindhoven, 1996). Cited once, on page 118.

[vzGG03] Joachim von zur Gathen and Jürgen Gerhard, 2003. *Modern Computer Algebra*. 2nd ed. Cambridge University Press, New York. Cited 5 times, on pages 117 and 124.

[Was76] Wolfgang Wasow, 1976. *Asymptotic Expansions for Ordinary Differential Equations*. Dover Publications. Cited once, on page 75.

[Wei84] Boris Weisfeiler, 1984. «Post-classification version of Jordan's theorem on finite linear groups.» *Proc. Nat. Acad. Sci. USA*, 81(16):5278–5279. Cited once, on page 52.

[Wei95] Jacques-Arthur Weil, 1995. *Constantes et polynômes de Darboux en algèbre différentielle: applications aux systèmes différentiels linéaires*. Ph.D. thesis, École Polytechnique. Cited twice, on page 108.

[Wel01] Michael Welschenbach, 2001. *Cryptography in C and C++*. 2nd ed. Apress, Berkeley, California. Cited once, on page 61.

[Wey24] Hermann Weyl, 1924. «Randbemerkungen zu Hauptproblemen der Mathematik.» *Mathematische Zeitschrift*, 20(1):131–150. Cited once, on page 68.

[Wey52] ———, 1952. *Symmetry*. Princeton University Press. Cited once, on page 94.

[WS73] W. Wulf and Mary Shaw, 1973. «Global variable considered harmful.» *SIGPLAN Notices*, 8(2):28–34. Cited once, on page 83.

[Żoł06] Henryk Żołądek, 2006. *The monodromy group*. No. 67 in Monografie Matematyczne (New Series) Instytut Matematyczny Polskiej Akademii Nauk. Birkhäuser Verlag, Basel. Cited once, on page 40.