ON ADAPTIVE APPROXIMATION AND D-FINITE FUNCTIONS

Dissertation

zur
Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Ingenieurwissenschaften
der Universität Rostock

vorgelegt von

Olaf Becken, geb. am 6. 5. 1966 in Greifswald

aus Flörsheim am Main

Flörsheim am Main, den 22. Februar 1999

Betreuer und erster Gutachter:
Prof. Dr. rer. nat. habil. K. Hantzschmann

Gutachter:
Prof. Dr. V.P. Gerdt (Nukleares Forschungsinstitut Dubna)
Prof. Dr. rer. nat. W. Koepf (HTWK Leipzig)

Die Dissertation wurde am 9. 3. 2000 verteidigt.

> Unter allen Wissenschaften bauet keine ihre Priester so sehr gegen andere Wissenschaften ein als die sich selbst genügsame Meßkunst, indes die meisten andern die Meßrute selber als eine blühende Aaronsrute entlehnen, die ihnen bei den Priesterwahlen raten helfen soll. Ich kann mir Mathematiker gedenken, die gar nicht gehöret haben, daß ich in der Welt bin, und die also nie diese Zeile zu Gesicht bekommen.
> – Jean Paul

Diese Abhandlung ist Swetlana, Anna und Michael gewidmet. Ihre Ermutigung und Rücksichtnahme gaben mir die notwendige Kraft und Zeit für meine Arbeit. Besonderer Dank gilt Prof. Karl Hantzschmann für seine Geduld, behutsame Führung und Unterstützung. Ich danke ihm, Prof. Vladimir Petrovich Gerdt, Prof. Wolfram Koepf und Dipl.-Inf. Andreas Jung für ihre wertvollen Korrekturhinweise und Anregungen.

Ich versichere: Ich habe die eingereichte Dissertation selbständig und ohne fremde Hilfe verfaßt, andere als die von ihm angebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht.

Contents

List of Notations

a, b, c	sequence of elements a, b, c
$\{a, b, c\}$	set of elements a, b, c
$x \leftarrow y$	term substitution of x by y
$\forall j = a(b)c$	the index j takes values between a and c (including a and c) with step size b
$[x_0, x_1]$	closed interval on the real line between x_0 and x_1
(x_0, x_1)	open interval on the real line between x_0 and x_1
$\hat{C}^{(\nu)}[x_0, x_1]$	space of all functions defined on $[x_0, x_1]$ with continuous derivatives up to the order $\nu - 1$ and a piecewise continuous ν-th derivative
$C_2[x_0, x_1]$	Euclidean space of complex functions, defined on $[x_0, x_1]$, see section 2.3.1
$\mathbb{C}$	field of complex numbers
$\mathrm{condn}(A)$	condition number of the matrix A, see section 6.1
erf	Gaussian error function, Poisson function
$\deg(p, x)$	degree of the polynomial p by the variable x
$\det(A)$	determinant of the square matrix A
$\mathrm{GCRD}(L, M)$	greatest common right divisor of two homogeneous linear differential operators L and M, see definition 8
$\mathbb{K}$	constant field with characteristic 0
$\bar{\mathbb{K}}$	algebraic closure of $\mathbb{K}$
$\mathbb{K}[x]$	ring of polynomials in the variable x with coefficients in $\mathbb{K}$
$\mathbb{K}[[x]]$	ring of Taylor series in the variable x with coefficients in $\mathbb{K}$
$\mathbb{K}(x)$	field of rational functions in the variable x with coefficients in $\mathbb{K}$
$\mathrm{LCLM}(L, M)$	least common left multiple of two homogeneous linear differential operators L and M, see definition 8
l_2	Euclidean space of complex sequences, see section 2.3.2
$\mathbb{N}$	set of natural numbers 0, 1, 2, ...
$\nu_p(q)$	order of the polynomial q at the polynomial p, see section 4.4.2
$\mathrm{order}(L)$	order of the ordinary differential operator L
$\mathrm{pp}(p, x)$	primitive part of the polynomial p by the variable x, see section 4.1
rem	remainder in the commutative ring
$\mathrm{res}(p, q, x)$	resultant of the polynomials p and q taken w.r.t. the variable x
RRem	right remainder in the non-commutative ring, see definition 6
$\mathrm{trace}(A)$	trace of the square matrix A
$V(L)$	set of all solutions of the linear differential operator L
$V_Y(x)$	modified matrix to the sequence of functions Y and differentiation by x, see (2.13)
$W_Y(x)$	Wronskian matrix to the sequence of functions Y and differentiation by x
$\mathbb{Z}$	ring of integers

Chapter 1

Introduction

Let $\mathbb{K}$ be some constant field with characteristic 0 and $\bar{\mathbb{K}}$ its algebraic closure. We consider homogeneous linear ordinary differential equations (DEs) of any order (singular or regular) with polynomial coefficients, i.e. differential equations of the form

$$\begin{aligned} L\,y(x) \;&=\; \sum_{i=0}^{ord} q_i\, D^i\, y(x) \;=\; 0, \qquad (1.1)\\ &D\,y(x) = \frac{\partial}{\partial x}\,y(x),\ \forall i:\ q_i \in \mathbb{K}[x],\\ &ord \geq 1,\ q_{ord} \not\equiv 0, \gcd(q_0, q_1, \ldots, q_{ord}) = 1 \end{aligned}$$

and $\mathbb{K}$ is some constant field with characteristic 0. We denote the set of all solutions of a linear differential operator L as $V(L)$. A function $y : \mathbb{C} \to \mathbb{C}$ is called *differentiably finite* (or D-finite, for short), if there exists some L (1.1) with $y \in V(L)$. In this thesis we give definitions of D-finiteness only for the special case of univariate functions.

Similarly, a sequence $A = \ldots, a_{n-1}, a_n, a_{n+1}, \ldots,\ n \in \mathbb{Z}$, is called polynomially recursive (or P-recursive) when it satisfies a homogeneous linear recurrence equation (RE) with polynomial cofficients

$$\begin{aligned} R\,a_n \;&=\; \sum_{j=0}^{rank} p_j\, F^j\, a_n \;=\; 0, \qquad (1.2)\\ &F\,a_n = a_{n-1},\ \forall j:\ p_j \in \mathbb{K}[n],\\ &p_0 \not\equiv 0,\ p_{rank} \not\equiv 0. \end{aligned}$$

It can be shown, that the generating function $y(x) = \sum_{n\in\mathbb{Z}} a_n\, x^n$ of a P-recursive sequence is D-finite and that reciprocally the sequence of Taylor coefficients of a D-finite function is P-recursive [65].

The class of D-finite functions deserves our interest for at least two reasons. First, it contains the commonly used analytic functions: the algebraic, Bessel, cosine, Gaussian error, exponential, hypergeometric, logarithm, power, rational, sine and many more functions are D-finite. D-finite functions form an algebra which is closed under sum and product, substitution of algebraic functions, differentiation and integration [65, 41].

Second, D-finite functions can be represented by a finite amount of information. Each D-finite function can be uniquely defined by a DE (1.1) and *ord* linearly independent linear boundary constraints

$$U_{x_0}\ Y(x_0) + U_{x_1}\ Y(x_1) = \gamma, \qquad (1.3)$$

$x_0,\ x_1 \in \mathbb{R} \cap \mathbb{K},\ x_0 < x_1,\ Y = (y\ y'\ \ldots\ y^{(ord-1)})^T,\ U_{x_0},\ U_{x_1} \in (\mathbb{C} \cap \mathbb{K})^{ord \times ord},\ \gamma = (\gamma_0\ \gamma_1\ \ldots\ \gamma_{ord-1})^T,\ \gamma_i \in \mathbb{C} \cap \mathbb{K}$, or by a RE (1.2) with similar constraints. There are practicable algorithms, which transform D-finite functions, given in terms of "standard" functions, into this canonical representation (1.1,1.3) [55, 41].

The inverse problem is much more difficult: Let be given a DE (1.1) or a RE (1.2) with the corresponding constraints, find the D-finite function in terms of "elementary" (in some heuristical sense) functions. Algorithms of computer algebra which search for *exact* solutions to that problem can be found in the next section.

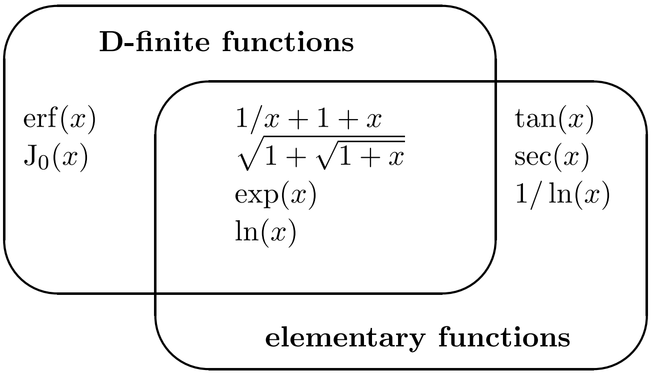
However, there are D-finite functions which are not elementary in the sense of differential algebra.

Definition 1 *Let $\mathbb{K}, k$ be differential fields with derivation D, $k \subset \mathbb{K}$.*

$\mathbb{K}$ is called an elementary extension of k $\overset{\text{def}}{\Longleftrightarrow}$
$\exists$ *tower of fields* $k = k_0 \subset k_1 \subset \ldots \subset k_r = \mathbb{K}\ \forall j = 1(1)r : k_j = k_{j-1}(\theta_j) \land \Big[(\theta_j$ *is algebraic over* $k_{j-1}) \lor (\exists \nu \in k_{j-1} :\ D\,\theta_j = \nu\,\theta_j) \lor (\exists \nu \in k_{j-1} :\ \nu\, D\,\theta_j = D\,\nu)\Big]$

Definition 2 *The function f is called elementary over k* $\overset{\text{def}}{\Longleftrightarrow}$ *f belongs to an elementary extension of k.*

The following picture gives some examples of elementary and D-finite functions:



In this thesis we show how to find simple approximate solutions $\tilde{y}$ for linear ODEs with polynomial coefficients. Beyond their simplicity the solution functions are elementary in the above defined sense. The accuracy of the approximate solutions is controlled by error bounds. Together with the algorithms from [55, 41], this result can be applied for finding elementary approximations to (non-elementary or too complicated) D-finite functions. Because transforming DEs (1.1) into RE's (1.2) and vice versa is trivial (see lemma 2.3), the techniques can also be used for finding approximate REs (1.2) and approximate P-recursive sequences.

1.1 Algorithms of computer algebra for solving ODEs with rational coefficients

We are working on the broad field of ordinary differential equations, whose history goes back at least to Euler and Newton and from this time has a great practical importance and, therefore, has a strong research interest. Many scientific schools have results in this area, but we decided to mention only the algorithms, which have really influenced us. We are aware that this choice will neglect the most well-known part - (pure) numerical algorithms.

Computer algebra (or french *calcul formel*) is a mathematical branch which came up in the 60's with the development of the first computer algebra systems (CAS). A CAS has a built-in mathematical knowledge and allows the user to implement algorithms in a language with mathematical structures.

In the last 30 years numerous algorithms for computer algebra were published [29]. Here we only mention the algorithms for differential algebra which have influenced us. Please note that most of the modern algorithms have their roots in the last century. Many algorithms of the past were not practicable, they can not be performed by hand because of tremendous calculations. Only the fast development of hardware and software technology gives us the possibility to implement (and improve) them. A good overview over the history and the current research give the comprehensive articles [59, 14, 57].

The algorithm of Singer [58, 60, 61] finds all Liouvillian solutions of linear DEs with Liouvillian coefficients.

Definition 3 *Let $\mathbb{K}, k$ be differential fields with derivation $'$, $k \subset \mathbb{K}$.*

$\mathbb{K}$ is called a Liouvillian extension of k $\overset{\text{def}}{\Longleftrightarrow}$

$\exists$ tower of fields $k = k_0 \subset k_1 \subset \ldots \subset k_r = \mathbb{K}$ $\forall j = 1(1)r : k_j = k_{j-1}(\theta_j) \wedge \Big[(\theta_j$ is algebraic over $k_{j-1}) \vee ({\theta_j}'/\theta_j \in k_{j-1}) \vee ({\theta_j}' \in k_{j-1})\Big]$

Definition 4 *The function f is called Liouvillian over k $\overset{\text{def}}{\Longleftrightarrow}$ f belongs to a Liouvillian extension of k.*

Example. $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int \exp(-x^2)\, dx$ is - because of $\pi \in \mathbb{C}$ - Liouvillian over $\mathbb{C}(x)$, the Bessel functions $\mathrm{J}_i(x), \mathrm{Y}_i(x)$ are not. By definition, elementary functions are Liouvillian.

Differential operators (1.1) are a special case of homogeneous linear ordinary differential operators

$$L = \sum_{i=0}^{ord} q_i(x)\, D^i,\ ord \geq 1,\ \forall i:\ q_i \in \hat{C}^{(0)}[x_0, x_1],\ q_{ord} \not\equiv 0 \tag{1.4}$$

and DEs (1.1) are a special case of homogeneous linear ODEs

$$L\ y(x) = 0. \tag{1.5}$$

(In general, $\hat{C}^{(\nu)}[x_0, x_1]$ stands for the space of all functions which are defined on $[x_0, x_1]$ and which have continuous derivatives up to the order $\nu - 1$ and a piecewise continuous ν-th derivative).

Let the coefficients of a homogeneous linear ordinary differential operator (1.4) be elements of the differential field k. Singer showed that, if $L\ y(x) = 0$ has a Liouvillian solution over k, then it has a solution of the form $y(x) = \exp(\int u(x)\, dx)$, where $u(x)$ is algebraic over k. Singer proposes a procedure, which produces an algebraic equation for $u(x)$. The degree of this equation can be bounded by a function $I(ord)$, which depends only on the order of the given differential operator. It is $I(2) = 12$, $I(3) = 36$, but $I(4) \leq 25920$. Singer himself calls his algorithm extremly inefficient and there is no implementation at all.

The algorithm of Kovacic [43, 69] can be treated as a special case of Singer's algorithm and finds all Liouvillian solutions for DEs (1.1) of second order. At this moment, there doesn't exist a complete implementation of Kovacic's algorithm.

Example. Let be given the DE $144\,x^2(x-1)^2\,D^2\,y(x) + (32\,x^2 - 27\,x + 27)\,y(x) = 0$. Then, the algorithm of Kovacic (and the algorithm of Singer, too) computes a solution $y(x) = \exp(\int u(x)\;dx)$ with $0 = 20736\,x^4(x-1)^4\,u^4 - 6912\,x^3(x-1)^3(7\,x-3)\,u^3 + 864\,x^2(x-1)^2(48\,x^2-41\,x+9)\,u^2 - 48\,x(x-1)(320\,x^3-409\,x^2+180\,x-27)\,u + 2048\,x^4 - 3484\,x^3 + 2313\,x^2 - 702\,x + 81$
Further work with such form of solution is quite complicated.

Let L be a linear homogeneous ordinary differential operator with coefficients in $\mathbb{K}(x)$. Then the *associated equations method* [11, 56, 76, 17] returns the right factor L_1 of lowest order over all possible decompositions $L = L_k L_{k-1} \ldots L_1$ into other linear homogeneous ordinary differential operators with coefficients in $\mathbb{K}(x)$. If there is a solution for $L\;y = 0$ of the form $y = e^{\int u}$ where $u \in \mathbb{K}(x)$ then this one can be determined. The algorithm was partially implemented in MACSYMA [56] and Axiom [17], but it suffers for complexity reasons.

In chapter 4 we will describe more in detail other algorithms (finite Laurent series solutions, polynomial solutions, rational solutions, Beke's method, van Hoeij/Singer factorization), for which stable complete implementations exist.

All these methods are pure algebraic methods, which are searching for exact solutions. This is their strength, but also their weakness: exact solutions are rare and often difficult to handle. Let us explain by an example, what this means.

One can compare factoring differential operators with factoring polynomials. If a polynomial equation, e.g.

$$x^7 - 12\,x^6 - 30\,x^5 + 1150\,x^4 - 7449\,x^3 + 21990\,x^2 - 30294\,x + 14796 = 0$$

is to solve, it is easier to apply numerical methods after a symbolic factorization into

$$(x-3)(x^3 + 6\,x^2 - 42\,x + 36)(x^3 - 15\,x^2 + 75\,x - 137) = 0.$$

The same is true for differential operators. We took the example from the thesis of Mark van Hoeij. The problem is, if we change the given problem by changing only one digit, say 14796 to 14797, there is no hope to find a symbolic factorization. An accidentally found symbolic factorization is often very complex. And the same is true for differential operators.

Therefore, in contrast to the above mentioned methods, other authors are interested in series solutions using Frobenius method [26, 23, 24, 68, 21] and Newton polygon techniques. Here especially the solvers DESIR and ISOLDE [4, 52] should be mentioned. Let L be a homogeneous linear differential operator (1.4) and let $\bar{\mathbb{K}}$ be the algebraic closure of $\mathbb{K}$.
If L is regular or regular singular at the point x_0 (see section 2.1 for the definition in the case of differential operators (1.1)), then the Frobenius algorithm computes the complete fundamental system with functions of the form

$$\begin{aligned} y(x) \quad &= \quad (x-x_0)^\lambda \left(t_0 + t_1 \ln(x-x_0) + \ldots + t_{ord-1} \ln(x-x_0)^{ord-1}\right), \\ &\qquad \lambda \in \bar{\mathbb{K}}, \end{aligned}$$

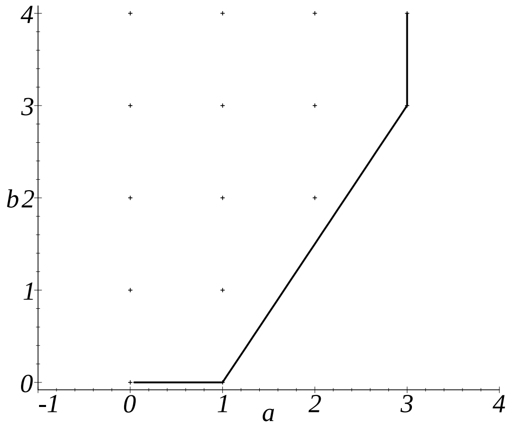
where $\forall i = 0, \ldots, ord{-}1$ the $t_i \in \bar{\mathbb{K}}[[x{-}x_0]]$ are Taylor series at the point x_0. The coefficients of the Taylor series may successively be evaluated by using recurrence equations.

Definition 5 *The Newton polygon* $\mathrm{N}(p^i\,D^j)$ *of a monomial* $p^i\,D^j$ *is defined as the set of two-dimensional points* $\left\{(a,b) \in \mathbb{Z}^2 \mid (0 \le a \le j) \wedge (i \le b)\right\}$. *The Newton polygon of a differential operator L (1.1) is defined as the convex hull of the Newton polygons of the monomials that appear in L. A rational number s is called a slope of L, if s is the slope of one of the edges of the Newton polygon.*

Example. The Newton polygon of $(x-1)^i\,D^j$ for

$$L = 4\,(x-1)^3\,D^3 + 18\,(x-1)^2\,D^2 + (12\,(x-1)-1)\;D$$

is



To become more familiar with Newton polygon techniques the reader is referred to [30, 25, 68, 3, 52, 72]. Most results in chapter 4 can be obtained by Newton polygon techniques.

1.2 Computer analysis

On the other hand, we will follow the main ideas of *computer analysis* [46, 45, 33]. The conception of computer analysis was first formulated by N.J. Lehmann and his school at Dresden University.

Computer analysis concentrates on finding analytical approximate solutions which should reflect inherent properties of the given problem and the possible influence of parameters. All formula expressions should be simple and transparent, while maintaining an appropriate level of precision. The accuracy of approximate solutions is controlled by error bounds, which are computed entirely by the computer program, the same way as approximations are determined without requiring user interaction.

Prevalent algorithms combine the tried methods of numerical computing with symbolic procedures. We use CAS and define fields of "elementary" functions which can be handled by CAS in an effective manner. The author feels the strong need for hybrid systems, which combine fast numerical routines (using floating point hardware) and symbolic capabilities of a CAS.

Since 1980 the following investigations were done (at Dresden, later at Rostock University):

- iteration methods for systems of ODEs [53],
- approximate solutions for boundary value problems and systems of ODEs using projection methods [31] and corresponding error estimation [32],
- a generalized collocation method for ODEs using higher derivatives and perturbation methods for ODEs with parameters [46],
- a Lipschitz-calculus, used for error estimates in the case of non-linear ODEs [46, 45],

- approximate solutions for singular initial value problems for ODEs by adaptive approximation [66, 35],
- a uniform conception of adaptive approximation using projection methods [34].

1.3 Philosophy of the adaptive approximation method

In this section we describe the adaptive approximation method [66, 47] in the general case of a non-linear differential equation. This method is a component of computer analysis.

Let be given a (linear or non-linear) ODE of n-th order

$$G(x, y, y', \ldots, y^{(n)}) = 0 \tag{1.6}$$

with m linearly independent boundary constraints

$$U\ y = \begin{pmatrix} U_1(y(x_0), y'(x_0), \ldots, y^{(n-1)}(x_0), y(x_1), y'(x_1), \ldots, y^{(n-1)}(x_1)) \\ U_2(y(x_0), y'(x_0), \ldots, y^{(n-1)}(x_0), y(x_1), y'(x_1), \ldots, y^{(n-1)}(x_1)) \\ \vdots \\ U_m(y(x_0), y'(x_0), \ldots, y^{(n-1)}(x_0), y(x_1), y'(x_1), \ldots, y^{(n-1)}(x_1)) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{1.7}$$

We are looking for an approximate solution $\tilde{y}$ for (1.6) and (1.7) in $[x_0, x_1]$.

First, we choose the *function* space for the approximate solution function $\tilde{y}$. In our case of D-finite functions we prefer the definition of the solution space by a class of differential operators, because from the algorithmic point of view, dealing with DEs (1.1) is easier then dealing with its solutions (e.g., the solution of an Eulerian DE may contain algebraic numbers while the Eulerian DE does not). We denote this class of differential operators by $\mathbb{S}$. Thus, $\mathbb{S}$ must have at least two properties:

- For all approximate solutions $\tilde{y}$ there exists a differential operator $\tilde{L} \in \mathbb{S}$ such that $\tilde{L}\ \tilde{y} = 0$.
- There is a practicable algorithm to compute the fundamental system for each differential operator in $\mathbb{S}$, respectively.

We select $\mathbb{S}$ not blindly, but pay attention to the physical nature of the problem. Furthermore, the choice of the solution space influences on the complexity of the subsequent computation steps.

Second, we choose *adaption criterions* for determining an element $\tilde{L} \in \mathbb{S}$. We determine q linearly independent functions $\Phi = \phi_1, \phi_2, \ldots, \phi_q$ with

$$\tilde{L}\ \phi_i = 0.$$

We demand $q \geq n$. We call this step *adaption.*

Third, we make a linear ansatz

$$\tilde{y} = \sum_{i=1}^{q} c_i \phi_i.$$

We use the constraints and *approximation criterions* for determining the coefficients c_i. We call this step *approximation.* Adaption and approximation criterions should match the origin of the problem.

The fourth step is an error estimation.

Dividing the computation of approximate solutions into four parts (modules) has several advantages. One can search in parallel for approximate solutions in distinct function spaces (with different complexities). Once found adaptive solutions can be reused with new approximation criterions. For a given class of problems one can heuristically find out the best combination of available adaption and approximation criterions. Last but not least, the modularity of the method eases software reengineering.

1.4 Motivation and schedule

The main concept of this thesis is the adaptive approximation method applied to linear ODEs with polynomial coefficients. We find simple approximate solutions $\tilde{y}$ for (1.1) and (1.3) in form of D-finite functions which are elementary over $\mathbb{K}(x)$ and analytical in $[x_0, x_1]$ (in special cases: analytical only in $(x_0, x_1]$).

Our methodology is inspired by computer algebra and analysis. Our aim is to show that both research areas can benefit from each other and that their synthesis leads to new results. We also hope that this thesis helps to close the gap between pure and applied mathematics.

In chapter 3 we choose the class of adaptive differential operators and define in this way the function space for the approximate solution function $\tilde{y}$. We introduce our class $\mathcal{ELF}$ - the class of *first order decomposable* differential operators with fundamental systems consisting of elementary functions. We show that $\mathcal{ELF}$ is useful to describe the singularities of linear ODEs with polynomial coefficients.

When we wrote chapter 4 our primary intention was to show how to filter out all "simple" *exact* solutions of a given DE (1.1) without much effort before starting any other method. This key-note remains, but additionally the structure theorems in this chapter make it possible to determine candidates for *approximate* solutions in subspaces of $\mathcal{ELF}$.

In the adaption step we obtain the adaptive differential operator $\tilde{L} \in \mathcal{ELF}$. We determine approximate solutions by applying adaption criterions to rest terms. How these rest terms arise and which adaption criterions can be applied will be offered in chapter 5. The advantageous adaption algorithm contained in this chapter is different from those of [66, 47, 34]. We decouple adaption from approximation and simplify the reuse of once found adaptive solutions Φ with new approximation criterions.

In the approximation step at chapter 6 we combine the "best" functions from Φ (in the sense of approximation criterions) to get our approximate fundamental system. We also take care of ill-conditioned solutions. Up to this point we never made use of the constraints, i.e. our approximate fundamental system is reusable for variant constraints (but in the same segment or initial point). Only in the end of the approximation step we use the constraints to build the approximate solution $\tilde{y}$.

The chapters 3, 4, 5 and 6 build up a coordinated adaptive approximation algorithm and the subsequent chapters use results from the previous ones.

The chapter 7 about error estimation was formed in cooperation with Andreas Jung [10]. It is a standalone part of the thesis, though it completes the adaptive approximation algorithm. We propose a method to calculate error estimates for closed-form approximate solutions to boundary and initial value problems in the case of linear differential equations, which covers the case of DEs (1.1).

A lot of examples are embodied in this thesis. Most of these examples originate from the results of CA programs written by the author. A thesis on DEs which claims for practical importance must include a chapter about implementation. Chapter 8 reports on the author's Maple package `DETools` and how it compares to programs of other authors.

At the end we summarize the essence of this work in short sentences (Thesen). By German law this theses should be written in German.

Chapter 2

Preliminaries

This chapter has been added to clarify definitions, notations and prove some basic facts. The literate reader might use it just for reference purposes.

2.1 ODEs with polynomial coefficients

The differential operator L (1.1) is an element of the non-commutative ring $\mathbb{K}(x)[D]$, which is an example of an Ore ring [49, 50, 51, 18, 19].

Definition 6 *Let k be a differential field and L, $l \in k[D]$. The right remainder* $\mathrm{RRem}(L, l)$ *is the (unique) differential operator $r \in k[D]$, for which holds $L = q\,l + r$,* $\mathrm{order}(r) < \mathrm{order}(l)$*, $q \in k[D]$.*

Definition 7 *Let k be a differential field. A homogeneous linear differential operator $L \in k[D]$ is said to be* reducible *over k $\overset{\text{def}}{\Longleftrightarrow}$ $\exists l \in k[D] \setminus k : \mathrm{RRem}(L, l) = 0$. If L is not reducible over k, we say it is* irreducible *over k.*

Definition 8 *Let be given a non-empty set of homogeneous linear differential operators $\{L_1, \ldots, L_k\}$. Then, the* least common left multiple $\mathrm{LCLM}(L_1, \ldots, L_k)$ *is defined as the monic homogeneous linear differential operator with minimal order such that*

$$V(\mathrm{LCLM}(L_1, \ldots, L_k)) = V(L_1) \cup V(L_2) \cup \ldots \cup V(L_k)$$

and the greatest common right divisor $\mathrm{GCRD}(L_1, \ldots, L_k)$ *is defined as the monic homogeneous linear differential operator with maximal order such that*

$$V(\mathrm{GCRD}(L_1, \ldots, L_k)) = V(L_1) \cap V(L_2) \cap \ldots \cap V(L_k).$$

Each DE (1.1) at any point $x_0 \in \bar{\mathbb{K}}$ can be written in a normalized standard form

$$\begin{aligned} 0 \;=\; & L\,y(x) \;=\; \sum_{i=0}^{ord} \psi_i\,(x - x_0)^i\,y^{(i)}(x), \qquad (2.1)\\ & \psi_i = \sum_{j=0}^{rank} c_{i,j}\,(x - x_0)^j,\\ & \gcd(\psi_0, \psi_1, \ldots, \psi_{ord}) = 1,\; x_0, c_{i,j} \in \bar{\mathbb{K}},\; \exists i :\; c_{i,rank} \neq 0,\; ord \geq 1. \end{aligned}$$

We use this standard form for classifying the singular points:

Definition 9 *The DE (1.1) and the corresponding differential operator L are said to be regular at the point $x_0 \overset{\text{def}}{\Longleftrightarrow} q_{ord}(x_0) \neq 0$.*
The DE (1.1) and the corresponding differential operator L are said to be singular at the point $x_0 \overset{\text{def}}{\Longleftrightarrow} q_{ord}(x_0) = 0$.
The DE (1.1) and the corresponding differential operator L are said to be regular singular at the point $x_0 \overset{\text{def}}{\Longleftrightarrow}$ the DE is singular and $\psi_{ord}(x_0) \neq 0$.
The DE (1.1) and the corresponding differential operator L are said to be irregular singular at the point $x_0 \overset{\text{def}}{\Longleftrightarrow}$ the DE is singular and $\psi_{ord}(x_0) = 0$.

A DE (1.1) belongs to the *Fuchsian* class if it has no irregular singularities.

Theorem 2.1 *If the DE (1.1) is regular or regular singular at the point x_0, then there exist ord linearly independent solutions of the form*

$$\begin{aligned} y(x) &= (x-x_0)^{\lambda}\left(t_0 + t_1\ln(x-x_0) + \ldots + t_{ord-1}\ln(x-x_0)^{ord-1}\right), \\ &\quad \lambda \in \bar{\mathbb{K}}, \end{aligned} \tag{2.2}$$

where $\forall i = 0,\ldots,ord-1$ the $t_i \in \bar{\mathbb{K}}[[x-x_0]]$ are formal Taylor series at the point x_0.

If the DE (1.1) is irregular singular at the point x_0, then there exist ord linearly independent solutions of the form

$$\begin{aligned} y(x) &= (x-x_0)^{\lambda}\left(t_0 + t_1\ln(x-x_0) + \ldots + t_{ord-1}\ln(x-x_0)^{ord-1}\right)\exp(q), \\ &\quad \lambda \in \bar{\mathbb{K}},\ ri \in \mathbb{N}\setminus\{0\},\ w:\ \mathbb{C}\to\mathbb{C},\ w(x)^{ri} = x-x_0,\ q \in \bar{\mathbb{K}}[1/w(x)], \end{aligned} \tag{2.3}$$

where $\forall i = 0,\ldots,ord-1$ the $t_i \in \bar{\mathbb{K}}[[w(x)]]$ are formal Puiseux series

$$\begin{aligned} t_i(x) &= \sum_{n=0}^{\infty} a_n\, w(x)^n, \\ &\quad a_n \in \bar{\mathbb{K}} \end{aligned}$$

at the point x_0.

Proof. The regular or regular singular case goes back to [27, 28, 67, 26]. As an example of later publications, see [20, 21]. For the more complicated irregular case, see [22, chapter 4]. Note, w is an algebraic function, defined by $w(x)^{ri} = x - x_0$. □

The minimal ri such that one solution y of (1.1) can be written in the form (2.3), is called the *ramification index for y* (in point x_0). If we have a sequence of functions $Y = y_1, y_2, \ldots, y_k$ and each of them can be written in form (2.3) then the *ramification index for Y* is the gcd of the ramification indices of the y_i.

Lemma 2.2 *For each DE of the form (1.1) at any regular or regular singular point $x_0 \in \bar{\mathbb{K}}$ there exists at least one extended formal Laurent series (FLS) solution of the form*

$$\begin{aligned} y(x) &= (x-x_0)^{\lambda}\sum_{n=-\infty}^{\infty} a_n\,(x-x_0)^n, \\ &\quad a_n, \lambda \in \bar{\mathbb{K}}. \end{aligned} \tag{2.4}$$

Proof. The proposition follows directly from theorem 2.1. □

2.2 D-finite functions and recurrence equations

We use recurrence (difference) equations (REs) as a tool for solving differential equations. The reader is referred to [20, 21, 39, 40] for a detailed consideration of the behaviour of such REs.

Lemma 2.3 *In each regular or regular singular point a DE of the form (2.1) corresponds to a RE at the point x_0 of the form*

$$0 \;=\; R_{x_0}\, a_n \;=\; \sum_{j=0}^{rank} p_j\, F^j\, a_n \;=\; \sum_{j=0}^{rank}\left[\sum_{i=0}^{ord} c_{i,j}\, i!\binom{r-j}{i}\right] a_{n-j}, \tag{2.5}$$

$$F\, a_n = a_{n-1},\ n\in\mathbb{Z},\ \ a_n,\lambda\in\bar{\mathbb{K}},\ \ r=n+\lambda,\ \forall j:\ p_j\in\mathbb{K}[r],\ p_0\not\equiv 0,\ \ p_{rank}\not\equiv 0.$$

Proof. In each regular or regular singular point exists at least one solution (2.4). By substituting (2.4) into (2.1) we get

$$\begin{aligned}
0 \;&=\; \sum_{i=0}^{ord}\left[\sum_{j=0}^{rank} c_{i,j}\,(x-x_0)^j\right]\,(x-x_0)^i\, y^{(i)}(x),\\
&=\; \sum_{j=0}^{rank}\sum_{i=0}^{ord} c_{i,j}\,(x-x_0)^{j+i}\left[\sum_{n=-\infty}^{\infty} a_n\,(x-x_0)^{n+\lambda}\right]^{(i)}\\
&\quad\;\{\text{differentiate in the radius of convergence}\}\\
&=\; \sum_{j=0}^{rank}\sum_{i=0}^{ord} c_{i,j}\,(x-x_0)^{j+i}\left[\sum_{n=-\infty}^{\infty} a_n\, i!\binom{n+\lambda}{i}(x-x_0)^{n+\lambda-i}\right]\\
&=\; \sum_{j=0}^{rank}\sum_{i=0}^{ord} c_{i,j}\,(x-x_0)^j\, a_n\, i!\binom{n+\lambda}{i}(x-x_0)^{n+\lambda}\\
&\quad\;\{\text{substitute } n\leftarrow n-j\}\\
&=\; \sum_{j=0}^{rank}\sum_{i=0}^{ord} c_{i,j}\, a_{n-j}\, i!\binom{r-j}{i}(x-x_0)^{\lambda}
\end{aligned}$$

Dividing the last equation by $(x-x_0)^\lambda$, we get (2.5). $p_{rank}\not\equiv 0$, because $p_{rank}\equiv 0 \leftrightarrow \forall j:\ c_{rank,j}=0$ and this is a contradiction to (2.1). A similar argument holds for $p_0\not\equiv 0$ $(\gcd(\psi_0,\psi_1,\ldots,\psi_{ord})=1)$. □

Comparing (1.2) with (2.5) we recognize that REs (1.2) are special cases of REs (2.5) for $\lambda=0$, or rephrased, REs (2.5) are constructed by an extension of formal Laurent series (strictly speaking, by the multiplication with $(x-x_0)^\lambda$), whereas REs (1.2) may be constructed using formal Laurent series. If a D-finite function has a series expansion (2.4), then a RE (2.5) may be used to compute its coefficients. In regular and regular singular points the existence of at least one such D-finite function is guaranteed by lemma 2.2. Unfortunately, there are D-finite functions (e.g., $\exp(1/\sqrt{x})$), which don't have series expansions (2.4) in the irregular singular points of the corresponding DEs.

The natural number *rank* in (2.5) is called the *rank of recursion.* Let h be the greatest common divisor of all j in (2.5) for which holds $p_j(r)\not\equiv 0$. Then, h is called the *symmetry number* of the RE. E.g., the symmetry number of an Eulerian DE is 1. If $h = rank$ then the given DE is a (generalized) hypergeometric DE in x_0, and formal hypergeometric solutions can be immediately found.

Example. Let me illustrate the relations between linear ODEs with polynomial coefficients, D-finite functions, RE's (1.2) and P-recursive sequences. Using lemma 2.3, we want to determine the generating function $y(x) = 1 + x + 2x^2 + 3x^3 + 5x^4 + 8x^5 + \dots$ of the Fibonacci sequence $\dots,\ a_{-2} = 0,\ a_{-1} = 0,\ a_0 = 1,\ a_1 = 1,\ a_2 = 2, \dots$. This sequence is completely determined by the RE $n\,a_n - n\,a_{n-1} - n\,a_{n-2} = 0$ with the constraints $a_{-2} = 0$, $a_{-1} = 0$, $a_0 = 1$. By lemma 2.3 we construct in point $x_0 = 0$ the corresponding DE $(x^2 + x - 1)\,y' + (2x + 1)\,y = 0$. We solve the DE with the constraint and get $y(x) = \dfrac{-1}{x^2 + x - 1}$.

The polynomial p_0 is known as the *indicial polynomial*, the roots of the indicial polynomial are called *indices*. We know that the indices in regular points are always $0, 1, \dots, ord - 1$, as the following lemma shows:

Lemma 2.4 *The indicial polynomial of a DE (1.1) of order ord in a regular point is*

$$p_0(r) = c_{ord,0}\; r\,(r-1)\,(r-2)\dots(r - ord + 1).$$

Proof. By lemma 2.3

$$p_0 = \sum_{i=0}^{ord} c_{i,0}\, i! \binom{r}{i}$$

and if the DE is regular in x_0, then $\forall i = 0, \dots, ord - 1 :\ c_{i,0} = 0$. □

Note that rank and symmetry number are in fact functions of the chosen evaluation point x_0. Nevertheless, the following property of p_{rank} holds:

Lemma 2.5 $p_{rank(x_0)}(r{+}rank(x_0))$ *doesn't depend on the chosen regular or regular singular evaluation point* x_0.

Proof. By lemma 2.3 for all evaluation points x_0 holds

$$p_{rank(x_0)}(r + rank(x_0)) = \sum_{i=0}^{ord} c_{i,rank(x_0)}\, i! \binom{r}{i}$$

and the coefficients $c_{i,rank(x_0)}$ of the highest power of x are invariant to linear substitutions $x \leftarrow x + x_0$ in (2.1), even if the rank changes in some new evaluation point. □

If we substitute $x = \tilde{x} + x_0$ into a DO L of the form (1.1), where $\tilde{x}$ is the new indeterminate and x_0 is a free parameter, we also get a differential operator $\tilde{L}$ of the form (1.1). We construct the corresponding RE $\tilde{R}_0\ a_n = 0$ for $\tilde{L}$ in point 0. $\tilde{x} = 0 \leftrightarrow x = x_0$ implies that $\tilde{R}_0\ a_n = 0$ for $\tilde{L}$ and $R_{x_0}\ a_n = 0$ for L are equivalent. Finally, with

$$0 = \tilde{R}_0\ a_n\ \bmod(A(x_0)),\ A(x_0) \in \mathbb{K}[x_0], \tag{2.6}$$

we are able to compute the corresponding RE for (1.1) at $\{x_0 \mid A(x_0) = 0\}$. (If now $\tilde{L}$ is irregular singular at $\{x_0 \mid A(x_0) = 0\}$, then (2.6) describes only the Laurent series coefficients of solutions (2.4).) This procedure is advantageous, since we stay in the given coefficient field but nevertheless are able to predict the behaviour of the series solutions in algebraic evaluation points.

Example. Consider the DO

$$L = x\,(x^2+1)^2\,D^2 - (x^2+1)^2\,D + x^3,$$

taken from [62]. We are interested in the RE at $\{x_0 \mid {x_0}^2 + 1 = 0\}$. First, we substitute $x = \tilde{x} + x_0$ into L and get a DO

$$\tilde{L} = (\tilde{x} + x_0)\left((\tilde{x} + x_0)^2 + 1\right)^2 D^2 - \left((\tilde{x} + x_0)^2 + 1\right)^2 D + (\tilde{x} + x_0)^3$$

In practice, it is advisable to compute as early as possible modulo $A(x_0)$ to keep the expressions small. Thus, we compute $\tilde{L}$ modulo ${x_0}^2 + 1$ and write the result in the standard form (2.1) in point 0:

$$\tilde{L} = \tilde{x}^2(\tilde{x}^3 + 5\,x_0\,\tilde{x}^2 - 8\,\tilde{x} - 4\,x_0)D^2 + \tilde{x}(-\tilde{x}^3 - 4\,x_0\,\tilde{x}^2 + 4\,\tilde{x})\,D + \tilde{x}^3 + 3\,x_0\,\tilde{x}^2 - 3\,\tilde{x} - x_0$$

Now, we might compute the whole RE to L at $\{x_0 \mid {x_0}^2 + 1 = 0\}$ by constructing the RE to $\tilde{L}$ in point 0 and using lemma 2.3. We illustrate the computation of the index polynomial p_0. By formula (2.5) we know $p_0 = \sum\limits_{i=0}^{ord} c_{i,0}\, i!\binom{r}{i}$. Substituting $c_{0,0} = -x_0$, $c_{1,0} = 0$ and $c_{2,0} = -4\,x_0$, we get $p_0 = -x_0 - 4\,x_0 r(r-1) = -x_0(2\,r-1)^2$. The indices are $\frac{1}{2}$ and $\frac{1}{2}$. In section 4.1 it will be shown that the indices define the lowest powers of Laurent series solutions of DEs (1.1). Indeed, one exact solution of $L\; y(x) = 0$ is $y(x) = \sqrt{x^2+1}$.

2.3 Complex Euclidean space

Definition 10 *Let be given a complex linear space E. A function $f : E \times E \to \mathbb{C}$ is called* scalar product $\overset{\text{def}}{\Longleftrightarrow}$ *it fulfils the following properties:*

- $\forall a, b, c \in E : \; f(a + b, c) = f(a, c) + f(b, c)$
- $\forall \lambda \in \mathbb{C}\; \forall a, b \in E : \; f(\lambda a, b) = \lambda f(a, b)$
- $\forall a, b \in E : \; f(a, b) = \overline{f(b, a)}$
- $\forall a \in E : \; f(a, a) \geq 0$
- $(f(a, a) = 0) \to a$ *is the zero element of* E

The scalar product $f(a, b)$ is usually noted by $\langle a, b\rangle$.

A linear space with a fixed scalar product is called *Euclidean space*. The norm in Euclidean space is canonically defined by

$$\|f\| := \sqrt{\langle f, f\rangle}. \tag{2.7}$$

Note that in our notation, the scalar product and the norm are taken with respect to the first variable. If this is not the case, a dot denotes the appropriate position as in $\langle G(x,\cdot), f(\cdot)\rangle$ and $\|\Gamma(x,\cdot)\|$.

2.3.1 The space $C_2[x_0, x_1]$

We deal with the space $C_2[x_0, x_1]$ [42] of functions $f : [x_0, x_1] \to \mathbb{C}$, for which holds

$$\int_{x_0}^{x_1} |f(x)|^2\; dx < \infty.$$

These form a complex Euclidean space with the scalar product

$$\langle f(x), g(x)\rangle := \int_{x_0}^{x_1} f(x)\overline{g(x)}\; dx. \tag{2.8}$$

The norm in this special case is

$$\|f(x)\| = \sqrt{\int_{x_0}^{x_1} |f(x)|^2 \, Dx}.$$

If the integration interval is different from $[x_0, x_1]$, the integration limits are specified as in $\langle f(\cdot), g(\cdot)\rangle_x^{x_1}$ and $\|f(x)\|_{x_0}^{\xi}$.

2.3.2 The space l_2

l_2 is the space of complex sequences $A = a_0, a_1, \ldots, a_{n-1}, a_n, a_{n+1}, \ldots$, $n \in \mathbb{N}$, for which holds $\sum_{n=0}^{\infty} |a_n|^2 < \infty$ [42, 34]. These form a complex Euclidean space with the scalar product

$$\langle a_n, b_n\rangle := \sum_{n=0}^{\infty} a_n \overline{b_n}. \tag{2.9}$$

2.4 Principal solutions and Green functions

This section shall give the reader a short overview over the basic facts we need for the error estimation. For more details the reader is referred to [37, 38] or [48, 12].

We consider boundary value problems consisting of a linear ordinary differential equation

$$L\ y(x) = f(x) \tag{2.10}$$

and the boundary conditions (1.3). Here L is a homogeneous linear differential operator (1.4). The right hand side of (2.10) may be piecewise continuous, what we denote by $f \in \hat{C}^{(0)}[x_0, x_1]$.

We will restrict our view on those boundary value problems (2.10,1.3) for which the corresponding homogeneous problem consisting of the homogeneous linear DE (1.5) and homogeneous linear boundary constraints

$$\begin{aligned} 0 &= U_{x_0}\ Y(x_0) + U_{x_1}\ Y(x_1) \qquad (2.11)\\ & \quad x_0, x_1 \in \mathbb{R} \cap \mathbb{K},\ x_0 < x_1,\ Y = (y\ y'\ \ldots\ y^{(ord-1)})^T, U_{x_0}, U_{x_1} \in (\mathbb{C} \cap \mathbb{K})^{ord \times ord} \end{aligned}$$

has only the trivial solution $y(x) \equiv 0$.

By $f^{(\nu)}(u, s)$, we denote the ν-th partial derivative of $f(x, s)$ with respect to its *first* argument, taken at $x = u$, i.e. $D^{\nu} f(x, s)|_{x=u}$.

2.4.1 Principal solutions

Definition 11 *A function $\gamma(x, \xi)$ which is defined for $x_0 \leq x, \xi \leq x_1$ is called a* principal solution (French: solution principale, German: Grundlösung) *of (1.4) in $[x_0, x_1]$, iff*

1. *$\gamma(x, \xi)$ has partial derivatives in the square $x_0 \leq x, \xi \leq x_1$ with respect to x up to the order $ord - 2$ which are continuous in x and ξ (this part is of no significance for $ord = 1$),*

2. *$\gamma(x, \xi)$ has partial derivatives in both triangles $x_0 \leq x < \xi \leq x_1$ and $x_0 \leq \xi < x \leq x_1$ with respect to x up to the order ord which are continuous in x and ξ,*

3. *$\gamma(x, \xi)$ is, in both triangles, a solution to the homogeneous DE (1.5),*

4. for $x_0 < \xi < x_1$,

$$\lim_{\substack{\varepsilon\to 0\\ \varepsilon>0}} \gamma^{(ord-1)}(\xi+\varepsilon,\xi) - \lim_{\substack{\varepsilon\to 0\\ \varepsilon>0}} \gamma^{(ord-1)}(\xi-\varepsilon,\xi) = \frac{1}{q_{ord}(\xi)} \tag{2.12}$$

holds.

Definition 12 *Let be given $Y = y_1(x), y_2(x), \ldots, y_k(x)$, a non-empty sequence of functions. Then, we define*

$$W_Y(x) = \begin{pmatrix} y_1(x) & y_2(x) & \cdots & y_k(x) \\ y_1'(x) & y_2'(x) & \cdots & y_k'(x) \\ y_1''(x) & y_2''(x) & \cdots & y_k''(x) \\ \vdots & \vdots & \vdots & \vdots \\ y_1^{(k-1)}(x) & y_2^{(k-1)}(x) & \cdots & y_k^{(k-1)}(x) \end{pmatrix}.$$

$W_Y(x)$ is called the Wronskian matrix and $\det(W_Y(x))$ the Wronskian determinant for Y.

We define

$$V_Y(x,\xi) = \begin{pmatrix} y_1(\xi) & y_2(\xi) & \cdots & y_k(\xi) \\ y_1'(\xi) & y_2'(\xi) & \cdots & y_k'(\xi) \\ y_1''(\xi) & y_2''(\xi) & \cdots & y_k''(\xi) \\ \vdots & \vdots & \vdots & \vdots \\ y_1^{(k-2)}(\xi) & y_2^{(k-2)}(\xi) & \cdots & y_k^{(k-2)}(\xi) \\ y_1(x) & y_2(x) & \cdots & y_k(x) \end{pmatrix} \tag{2.13}$$

If the homogeneous linear differential operator (1.4) is regular in $[x_0, x_1]$, then it always has principal solutions. One of these is

$$\gamma_0(x,\xi) := \begin{cases} -\dfrac{\det(V_Y(x,\xi))}{2\,q_{ord}(\xi)\det(W_Y(\xi))} & : \quad x < \xi \\ 0 & : \quad x = \xi \\ \dfrac{\det(V_Y(x,\xi))}{2\,q_{ord}(\xi)\det(W_Y(\xi))} & : \quad x > \xi, \end{cases} \tag{2.14}$$

where Y is a solution basis $Y = y_1(x), \ldots, y_{ord}(x)$ of (1.5). $\gamma_0(x,\xi)$ doesn't depend on the choice of basis functions $y_1(x), \ldots, y_n(x)$, as long as they span the solution space of (1.5). The set of all principal solutions of (1.4) can be obtained by

$$\gamma(x,\xi) = \gamma_0(x,\xi) + \sum_{\nu=1}^{n} h_\nu(\xi)\, y_\nu(x) \tag{2.15}$$

with arbitrary continuous functions $h_1(\xi), \ldots, h_n(\xi)$. Principal solutions depend only on the differential operator, they do not depend on the choice of a fundamental system.

Principal solutions have a specific property which makes them very useful for the analysis of linear differential equations:

Theorem 2.6 *Let $\gamma(x,\xi)$ be a principal solution of (1.4) and $f \in \hat{C}^{(0)}[x_0, x_1]$. Then*

$$y(x) = \int_{x_0}^{x_1} \gamma(x,\xi) f(\xi)\, d\xi \tag{2.16}$$

is a solution to the non-homogeneous differential equation (2.10).

The proof is given, for example, in [37] for continuous functions $f(x)$, but it can easily be extended to the case where $f(x)$ is only piecewise continuous.

2.4.2 Green functions

Definition 13 *A function $\Gamma(x,\xi)$ is called a* Green function *of (1.4,2.11), iff it is a principal solution of the homogeneous linear ODE (1.4), and for each fixed ξ with $x_0 < \xi < x_1$, it fulfils the homogeneous linear boundary conditions (2.11) as a function in the variable x.*

Throughout this section, we generally assume (1.5,2.11) to have only the trivial solution $y(x) \equiv 0$. The following theorem gives the motivation for this assumption.

Theorem 2.7 *If the homogeneous linear differential operator (1.4) is regular in $[x_0, x_1]$ and (1.5,2.11) only has the trivial solution $y(0) \equiv 0$, then (1.5,2.11) has exactly one Green function $\Gamma(x,\xi)$.*

Proof. See [37, pp 248–249]. □

Since the Green function of (1.4, 2.11) is a special principal solution, it has the form (2.15). So it can be computed by substituting $y(x) = \gamma_0(x,\xi) + \sum_{\nu=1}^{n} h_\nu(\xi)\, y_\nu(x)$ (the right hand side of (2.15)) into (2.11), assuming $x_0 < \xi < x_1$, and by solving this linear equation system for the unknown functions $h_1(\xi), \ldots, h_n(\xi)$. Theorem 2.7 assures that this system is uniquely solvable.

Example. Consider the differential operator $L = (x-1)\,D^2 - x\,D + 1$, together with the boundary conditions $y(0) = 0$, $y'(0) + y'(1) = 0$. The solution space is spanned by $y_1(x) = x$, $y_2(x) = \exp(x)$, so with (2.14) we get

$$\gamma_0(x,\xi) = \begin{cases} -\dfrac{\xi\exp(x) - \exp(\xi)x}{2(\xi-1)^2\exp(\xi)} & : \quad x < \xi \\ 0 & : \quad x = \xi \\ \dfrac{\xi\exp(x) - \exp(\xi)x}{2(\xi-1)^2\exp(\xi)} & : \quad x > \xi. \end{cases}$$

Assuming $x_0 < \xi < x_1$, we substitute $y(x)$ by $\gamma_0(x,\xi) + h_1(\xi)\,x + h_2(\xi)\,\exp(x)$ in the boundary conditions, obtaining the equations

$$\begin{aligned} -\frac{\xi}{2(\xi-1)^2\exp(\xi)} + h_2(\xi) &= 0 \\ \frac{\xi(\exp(1)-1)}{2(\xi-1)^2\exp(\xi)} + 2\,h_1(\xi) + (1+\exp(1))\,h_2(\xi) &= 0 \end{aligned}$$

which is uniquely solved by $h_1(\xi) = -\dfrac{\xi\,\exp(1-\xi)}{2(\xi-1)^2}$, $h_2(\xi) = \dfrac{\xi}{2(\xi-1)^2\exp(\xi)}$, thus

$$\Gamma(x,\xi) = -\frac{x\,\xi\,\exp(1-\xi)}{2(\xi-1)^2} + \frac{\xi\exp(x-\xi)}{2(\xi-1)^2} + \gamma_0(x,\xi).$$

Corollary 2.8 *Let $\Gamma(x,\xi)$ be the Green function of some homogeneous linear differential operator L (1.4) and some homogeneous linear boundary constraints (2.11). Then, the Green function of $a(x)\,L$ and the same boundary constraints is $\dfrac{\Gamma(x,\xi)}{a(\xi)}$.*

The notion of principal solutions and Green functions was originally introduced for the analysis of non-homogeneous boundary value problems. The next theorem is an application of theorem 2.6 to those problems.

Theorem 2.9 *Let $\Gamma(x,\xi)$ be the Green function of (1.4,2.11), and let $f \in \hat{C}^{(0)}[x_0, x_1]$. Then*

$$y(x) = \int_{x_0}^{x_1} \Gamma(x,\xi) f(\xi) d\xi \tag{2.17}$$

is the unique solution to the non-homogeneous linear ODE (2.10) with homogeneous linear boundary constraints (2.11).

For a proof, the reader is referred to [37] or [48, pp 311–313], for example.

Corollary 2.10 *Let $\Gamma(x,\xi)$ be the Green function of (1.4,2.11), and let $f \in \hat{C}^{(ord)}[x_0, x_1]$ fulfil the same homogeneous linear boundary constraints (2.11). Then*

$$\begin{aligned} f(x) &= \int_{x_0}^{x_1} \Gamma(x,\xi) L_{(\xi)} f(\xi)\ d\xi \qquad (2.18)\\ &= \int_{x_0}^{x} \Gamma(x,\xi) L_{(\xi)} f(\xi)\ d\xi + \int_{x}^{x_1} \Gamma(x,\xi) L_{(\xi)} f(\xi)\ d\xi. \end{aligned}$$

holds for $x_0 \le x \le x_1$.

Proof. Because of $L\, f \in \hat{C}^{(0)}[x_0, x_1]$ and theorem 2.9, the integral on the right hand side of (2.18) is a solution to $L\, y(x) = L\, f(x)$ which fulfils (2.11), as well as $f(x)$ itself. Since (1.5,2.11) is uniquely solvable, (2.18) holds. □

One remark about notation. In the literature one finds the equation (2.18), although for the proof the integral is decomposed. We always keep in mind that for $x \in [x_0, x_1]$

$$\begin{aligned} \int_{x_0}^{x_1} \left\{ \begin{array}{ccc} -h(\xi) & : & x < \xi \\ 0 & : & x = \xi \\ h(\xi) & : & x > \xi \end{array} \right\} d\xi &= \int_{x_0}^{x} h(\xi)\ d\xi + \int_{x}^{x_1} -h(\xi)\ d\xi \\ &= 2\, H(x) - H(x_0) - H(x_1), \qquad H' = h. \end{aligned}$$

Now let's have a closer look on (2.14). It can be regarded as the sum of two separate functions. This motivates the following definition:

Definition 14 *We define*

$$\Gamma_l(x,\xi) = \begin{cases} -\dfrac{\det(V_Y(x,\xi))}{q_{ord}(\xi)\det(W_Y(\xi))} & : \ x < \xi \\ 0 & : \ x \ge \xi \end{cases} \tag{2.19}$$

as the left Green function *of (1.4) and*

$$\Gamma_r(x,\xi) = \begin{cases} 0 & : \ x \le \xi \\ \dfrac{\det(V_Y(x,\xi))}{q_{ord}(\xi)\det(W_Y(\xi))} & : \ x > \xi \end{cases} \tag{2.20}$$

as the right Green function *of (1.4).*

Note that

$$\Gamma_l(x,\xi) + \Gamma_r(x,\xi) = 2\gamma_0(x,\xi). \tag{2.21}$$

The name "left and right Green function" is indeed justified, as the following theorem states:

Theorem 2.11 *Let (1.4) be regular in* $[x_0, x_1]$. *Then (2.19) is the unique Green function of (1.4) and the boundary conditions*

$$y(x_1) = y'(x_1) = \cdots = y^{(n-1)}(x_1) = 0, \tag{2.22}$$

whilst (2.20) is the unique Green function of (1.4) and the boundary conditions

$$y(x_0) = y'(x_0) = \cdots = y^{(n-1)}(x_0) = 0. \tag{2.23}$$

Proof. By comparing (2.19) and (2.20) with (2.14) and by observing that $D^\nu \det(V_Y(x, \xi))|_{x=\xi} = 0$ for $\nu = 0, \ldots, n-2$ and $D^{n-1} \det(V_Y(x, \xi))|_{x=\xi} = \det(W_Y(\xi))$, we state that they fulfil the properties of definition 11. Obviously, for each fixed ξ with $x_0 < \xi < x_1$, (2.19) and (2.20) fulfil (2.22) and (2.23), respectively. Since (1.5,2.22) and (1.5, 2.23) are indeed uniquely solvable initial value problems, the theorem holds by definition 13 and theorem 2.7. □

Theorem 2.12 *Let (1.4) be regular in* $(x_0, x_1]$*, (2.19) its left Green function and* $f \in \hat{C}^{(ord)}(x_0, x_1]$ *fulfil (2.22). Then for* $x_0 < x \leq x_1$,

$$f(x) = \int_x^{x_1} \Gamma_l(x, \xi) L_{(\xi)} f(\xi) d\xi \tag{2.24}$$

holds.

Proof. Let $x_0 < x \leq x_1$. Then there exists a c with $x_0 < c \leq x \leq x_1$, and with Theorem 2.11, (2.19) is the Green function of (1.4, 2.22) on the interval $[c, x_1]$. With corollary 2.10 and the unique solvability of (1.5,2.22), we get

$$f(x) = \int_c^b \Gamma_l(x, \xi) L_{(\xi)} f(\xi) d\xi = \int_x^{x_1} \Gamma_l(x, \xi) L_{(\xi)} f(\xi) d\xi,$$

where the second equal sign holds because of $\Gamma_l(x, \xi) = 0$ for $x > \xi$, see (2.19). □

The next corollary shows how to combine the fundamental systems of the factors of a linear differential operator.

Corollary 2.13 *d'Alembert reduction, [38, section 17.2]*

Let the homogeneous linear differential operator L *be regular in* $(x_0, x_1]$. *Let* $L = MN$ *be a given factorization of* L *into homogeneous linear differential operators* M *and* N *of order* $ord - n$ *and* n*, respectively. Let* $f_1(x), \ldots, f_{ord-n}(x)$ *be a fundamental system of* M *and* $g_1(x), \ldots, g_n(x)$ *be a fundamental system of* N. *Let* Γ_l *be the left Green function of* N *and homogeneous initial value conditions (2.22). Assign*

$$\forall j = 1(1)n : \; y_j(x) := g_j(x)$$

and

$$\forall j = 1(1)ord - n : \; y_{n+j}(x) := \int_x^{x_1} \Gamma_l(x, \xi) f_j(\xi)\, d\xi.$$

Then, $\forall x \in (x_0, x_1]$ $y_1(x), \ldots, y_{ord}(x)$ *is a fundamental system of* L.

Proof. First, we observe $\forall j = 1(1)n : \; L\ y_j(x) = MN\ y_j(x) = M\ 0 = 0$. The left Green function Γ_l is a special principal solution. Thus, by theorem 2.6 $\forall j = 1(1)ord - n :$ $y_{n+j}(x) = \int_{x_0}^{x_1} \Gamma_l(x, \xi) f_j(\xi)\, d\xi$ is a solution to the non-homogeneous differential equation

$N\,y(x) = f_j(x)$ $(\forall x \in (x_0, x_1])$. Therefore, $\forall j = 1(1)ord - n:\ L\ y_{n+j}(x) = MN\ y_{n+j}(x) = M\ f_j(x) = 0$, i.e. all functions in $y_1(x), \dots, y_{ord}(x)$ are solutions of $L\ y = 0$.

They are linearly independent, too. We assumed $y_1(x), \dots, y_n(x)$ to be linearly independent. If $y_{n+j}(x) = \sum_{i=1}^{n} a_i\, g_j(x)$ for at least one nonzero constant a_i, then $L\ y_{n+j}(x) = L\ \sum_{i=1}^{n} a_i\, g_j(x) = 0$, but $L\ y_{n+j}(x) = f_j(x)$, which leads to a contradiction. If $y_{n+j}(x) = \sum\limits_{\substack{i=1\\ i\neq j}}^{ord-n} a_i\, y_{n+i}(x)$ for at least one nonzero constant a_i, then

$$f_j(x) = L\ y_{n+j}(x) = \sum_{\substack{i=1\\ i\neq j}}^{ord-n} a_i\, L\ y_{n+i}(x) = \sum_{\substack{i=1\\ i\neq j}}^{ord-n} a_i\, f_i(x),$$

which contradicts the linear independence of the fundamental system of M. □

Chapter 3

Choosing the class of adaptive differential operators

In this chapter we have to choose the function space for the approximate solution $\tilde{y}$ to a DE (1.1) and *ord* linearly independent linear boundary constraints. We prefer the definition of the solution space by a class $\mathbb{S}$ of adaptive differential operators, because the exact solution in form of a D-finite function can also be described by a DO with constraints. Our concept of the thesis defines the properties of $\mathbb{S}$:

- There should be a practicable algorithm to compute the fundamental system for each differential operator in $\mathbb{S}$ respectively.
- The fundamental system of each differential operator in $\mathbb{S}$ should consist of functions which are elementary over $\mathbb{K}(x)$. From our point of view, non-elementary Liouvillian solutions are not practicable enough. Additionally, we would have more computational problems in chapter 7.
- $\mathbb{S}$ should be a subset of the set of DEs (1.1).
- In the case of boundary value problems in $[x_0, x_1]$, $\tilde{y}$ should be analytical in $[x_0, x_1]$.

In the past, the following classes (which own the above properties) were used [66, 47, 34]:

- $\mathcal{POL} = \left\{ L \,|\, V(L) \subset \bar{\mathbb{K}}[x] \right\}$
- $\mathcal{RAT} = \left\{ L \,|\, V(L) \subset \bar{\mathbb{K}}(x) \right\}$
- $\mathcal{CON} = \left\{ L \,|\, \exists c_1, \ldots c_{ord} \in \bar{\mathbb{K}} : \; L = \sum\limits_{i=0}^{ord} c_i \, D^i \right\}$
- $\mathcal{EUL} = \left\{ L \,|\, \exists c_1, \ldots c_{ord}, x_0 \in \bar{\mathbb{K}} : \; L = \sum\limits_{i=0}^{ord} c_i \, (x - x_0)^i \, D^i \right\}$

We note an important special case: the given DE (1.1) is singular in x_0 and a initial value problem is given in point x_1. We are interested in the behaviour of the solution function near x_0. Then, $\mathbb{S}$ must also be able to describe all kinds of singularities given as from theorem 2.1. None of the classes $\mathcal{POL}$, $\mathcal{RAT}$, $\mathcal{CON}$ and $\mathcal{EUL}$ has the capability to do this. First steps in this direction were made by Thinh [66], who used special cases of extended Taylor series (2.2), which were able to describe singularities of Fuchsian type.

Each Eulerian differential operator $E \in \mathcal{EUL}$ at the point x_0 factors into

$$E = ((x - x_0)\, D + e_{ord}) \cdots ((x - x_0)\, D + e_2)\, ((x - x_0)\, D + e_1)\,,$$

and each differential operator $C \in \mathcal{CON}$ with constant coefficients factors into

$$C = (D + e_{ord}) \cdots (D + e_2)\, (D + e_1)$$

where the $e_i \in \bar{\mathbb{K}}$. There exist similar factorizations into first order differential operators for the differential operators in $\mathcal{POL}$ and in $\mathcal{RAT}$. We generalize this idea and derive

Definition 15 *(cf. definition 7)*
Let $\mathbb{K}$ be a constant field. A differential operator L (1.1) is said to be first order decomposable *over $\bar{\mathbb{K}} \overset{\text{def}}{\Longleftrightarrow}$ L can be written as $L = L_{ord} \dots L_2\, L_1$ where L_i are linear homogeneous differential operators of first order with coefficients in $\bar{\mathbb{K}}(x)$.*

We denote by $\mathcal{ELF}$ the class of differential operators (1.1), which

- have only elementary solutions over $\mathbb{K}(x)$ and
- are first order decomposable over $\bar{\mathbb{K}}(x)$.

The question, whether a given first order decomposable differential operator has only elementary solutions over $\mathbb{K}(x)$, can algorithmically be decided by Risch integration [54, 29].

$\mathcal{ELF}$ is our biggest class of adaptive differential operators. $\mathcal{POL}$, $\mathcal{RAT}$, $\mathcal{CON}$ and $\mathcal{EUL}$ are proper subclasses of $\mathcal{ELF}$. If we compare $\mathcal{ELF}$ with the solution structure in theorem 2.1, we see that series expansions of solutions for DEs $\in \mathcal{ELF}$ have the required form, with only one (rare) exception: We are only able to produce solutions with ramification index 1. This will suffice for

- boundary value problems with DEs (1.1) or
- initial value problems with Fuchsian DEs (1.1).

Solutions with ramification index greater than 1 may only appear when we consider the behaviour of the solution of some initial value problem near some irregular singularity point. With the method shown in the next section, we are able to express asymptotically the singularities of DEs (1.1) in *all* cases.

3.1 The ramification index is greater than 1

We want to transform a DE (1.1) with a ramification index greater than 1 in point $x_0 \in \bar{\mathbb{K}}$ into a DE (1.1) with the ramification index 1 in point x_0.

Suppose, we have already computed the ramification index of the original DE. Then, by a substitution

$$x \leftarrow (x - x_0)^{ri} + x_0 \tag{3.1}$$

we transform the set of solutions (2.3) into a set of functions with ramification index 1 (in point x_0). Therefore, the substitution also transforms (1.1) into another DE (1.1), but with ramification index 1.

Thus, a solution to the problem for DEs (1.1) is to try out substitutions $x \leftarrow (x-x_0)^2 + x_0$, $x \leftarrow (x - x_0)^3 + x_0$, ... as long as the number of linearly independent solutions in form of extended Puiseux series (2.3) is less than the order of the DE.

Suppose the DEs we got by the substitutions are L_2, L_3, ... Then solutions to L_2, L_3, ... of the form (4.13) can be computed by a DESIR-like algorithm [68, 4, 52], which combines Frobenius with Beke's method. If y_j is a solution of $L_j\ y(x) = 0$ and y_k is a solution of $L_k\ y(x) = 0$, then y_j and y_k can only be linearly dependent in the case, if the result of the substitution $x \leftarrow (x - x_0)^{k/j} + x_0$ into $y_j(x)$ is $y_k(x)$. Finally, we get the solutions (2.3) of the original DE by inverse substitutions.

Chapter 4

Adaption step - restricting the coefficient space

In this chapter we describe, how to find

- exact solutions,
- candidates for approximate solutions

of DEs (1.1). After defining the class of adaptive differential operators $\mathcal{ELF}$ in the previous chapter, the structure theorems in this chapter make it possible to determine candidates for approximate solutions in subclasses of $\mathcal{ELF}$. We demonstrate the search for solutions in those subclasses of $\mathcal{ELF}$ which are small enough to guarantee practicable algorithms even in the case of exact solutions.

4.1 Finite Laurent series solutions

Lemma 4.1 *[6, lemma 2] If there exists a solution of the form*

$$\begin{aligned} y(x) &= (x-x_0)^{\lambda} \sum_{n=\alpha}^{\omega} a_n \, (x-x_0)^n, \qquad (4.1) \\ & \quad \alpha, \omega \in \mathbb{Z}, \;\; x_0, a_n, \lambda \in \bar{\mathbb{K}}, \;\; a_\alpha \neq 0, \;\; a_\omega \neq 0, \end{aligned}$$

for a DE of the form (1.1), then the following condition holds for the corresponding RE (2.5) at the point x_0:

$$p_0(\lambda+\alpha) = 0 \textit{ and } p_{rank}(\lambda+\omega+rank) = 0. \qquad (4.2)$$

Proof. With (4.1), (2.5) and $n = \alpha$ we conclude that

$$\begin{aligned} 0 &= p_0(\lambda+\alpha)\, a_\alpha + p_h(\lambda+\alpha)\, a_{\alpha-h} \cdots + p_{rank}(\lambda+\alpha)\, a_{\alpha-rank} \\ &= p_0(\lambda+\alpha)\, a_\alpha + p_h(\lambda+\alpha)\, 0 + \cdots + p_{rank}(\lambda+\alpha)\, 0 \\ &= p_0(\lambda+\alpha)\, a_\alpha, \end{aligned}$$

where h is the symmetry number of (2.5). $a_\alpha \neq 0$ implies that $\lambda+\alpha$ is a root of the indicial polynomial p_0. With $n = \omega + rank$ we conclude analogously that

$$\begin{aligned} 0 &= p_0(\lambda+\omega+rank)\, a_{\omega+rank} + \cdots + p_{rank}(\lambda+\omega+rank)\, a_\omega \\ &= p_0(\lambda+\omega+rank)\, 0 + \cdots + p_{rank}(\lambda+\omega+rank)\, a_\omega \\ &= p_{rank}(\lambda+\omega+rank)\, a_\omega \end{aligned}$$

$a_\omega \neq 0$ implies that $\lambda + \omega + rank$ is a root of p_{rank}. □

In [56, bound 1], [2, page 267], [1, theorem 1] the reader will find similar observations, sometimes extended to the inhomogeneous case, but always restricted to the case $y \in \mathbb{K}[x]$.

Definition 16 *We introduce the notation* pp *for the primitive part of a nonzero polynomial p. If $p = a_0 + a_1\, r + \cdots + a_m\, r^m \in \mathbb{K}[r] \setminus \{0\}$, then* $\mathrm{pp}(p, r) = p / \gcd(a_0, a_1, \ldots, a_m)$.

We use lemma 4.1 to determine all finite Laurent series solutions at any given point x_0 for any given DE (1.1). Each such solution must satisfy (4.2). Besides, the step size between the powers of two distinct nonzero terms of the series (4.1) must be a multiple of the symmetry number of the corresponding RE (2.5). Let h be the symmetry number of (2.5) and let $\beta = \lambda + \alpha$. Then, we solve $\{\mathrm{pp}(p_0(\beta), \beta) = 0,\ \mathrm{pp}(p_{rank}(\beta + n\, h + rank), \beta) = 0\}$ for $\{\beta, n\}$ with the condition $n \in \mathbb{N}$. Considering the fact that $\deg(p_0, r) \leq ord$ and $\deg(p_{rank}, r) \leq ord$, we conclude that the number of possible candidates is at most ord^2.

Our implementation *always* uses existing Gröbner basis solvers for solving systems of polynomial equations. In this way we compute the smallest univariate polynomial over n. We should not factorize p_0 and p_{rank} over $\bar{\mathbb{K}}$ (which could be troublesome for DEs with higher order), but only have to find whether they have roots that differ by an integer. This can also be done in another way. Let $P, Q \in \mathbb{K}[x]$. We introduce the notation $\mathrm{res}(P, Q, x)$ for the resultant of P and Q (taken w.r.t. x) [29]. Then, the natural-number roots for n are in the roots of $\mathrm{res}(\mathrm{pp}(p_0(\beta), \beta), \mathrm{pp}(p_{rank}(\beta + n\, h + rank), \beta), \beta)$.

By substituting the candidate pairs into the given DE, we get linear equation systems for the coefficients. Our implementation solves them, using existing linear solvers, because we prefer the reduction to this well-known problem. One could also compute the coefficients in a forward procedure, using the recurrence as done in [1, section 3.1]. This means that for each given candidate pair $(\lambda + \alpha, \lambda + \omega)$ we set $a_{\lambda+\alpha} := 1$ and use the equation

$$a_n = -\sum_{j=1}^{rank} p_j(r)\, a_{n-j} / p_0(r)$$

by substituting one after another $n = \alpha + h$, $n = \alpha + 2h$, ..., $n = \omega$. But then one has to struggle with the difficulties arising from the fact that the numerator or the denominator of the right hand side of the equation can be 0.

Algorithm 1 Finite_Laurent_Series_Solutions(L, x_0)
(The algorithm computes all solutions of the form (4.1) at the point x_0 for the differential operator L (1.1).)

```
1. Compute the symmetry number $h$, the indicial polynomial $p_0$, the rank and $p_{rank}$ at
   the point $x_0$.

2. Define $\beta = \lambda + \alpha$. Solve the system of two polynomial equations $\{\mathrm{pp}(p_0(\beta), \beta) = 0,$
   $\mathrm{pp}(p_{rank}(\beta + n\, h + rank), \beta) = 0\}$ for $\{\beta, n\}$ with the condition $n \in \mathbb{N}$. Construct the
   set of all solution pairs $\mathcal{P} = \{(\beta_i, \beta_i + n_i\, h)\}$.

3. $\{y_k\} := \emptyset$
   for each $(\beta, \gamma) \in \mathcal{P}$

      (a) Substitute $y(x) = a_\beta\, (x - x_0)^\beta + a_{\beta+h}\, (x - x_0)^{\beta+h} + \cdots + a_\gamma\, (x - x_0)^\gamma$ with free
          coefficients $a_\beta$, $a_{\beta+h}$, ..., $a_\gamma$ into $L\ y(x) = 0$. Solve the linear equation system
          for the coefficients. Let $\{\bar{\mathbf{a}}_k\}$ be a set of nontrivial linearly independent solution
          vectors.
```

(b) *for each* $\bar{\mathbf{a}}_k = (\hat{a}_\beta, \hat{a}_{\beta+h}, \ldots, \hat{a}_\gamma)^T$

$$\{y_k\} := \{y_k\} \cup \{\hat{a}_\beta\,(x-x_0)^\beta + \hat{a}_{\beta+h}\,(x-x_0)^{\beta+h} + \cdots + \hat{a}_\gamma\,(x-x_0)^\gamma\}$$

4. *return* $\{y_k\}$.

Example. Consider the DO

$$L = x\,(x^2+1)^2\,D^2 - (x^2+1)^2\,D + x^3,$$

taken from [62]. In section 2.2 we have already showed, how to compute the indicial polynomial at $\{x_0 \mid {x_0}^2+1=0\}$. Using lemma 2.3 we now compute the whole RE at $\{x_0 \mid {x_0}^2+1=0\}$:

$$-x_0(2\,r-1)^2\,a_n + (-8\,r^2+28\,r-23)\,a_{n-1} + x_0(5\,r^2-29\,r+41)\,a_{n-2} + (r-4)^2\,a_{n-3} = 0$$

The RE has symmetry number 1 and rank 3. We solve $\{(2\,\beta-1)^2=0,\,(\beta+n+3-4)^2=0\}$. There is no solution with a natural number n. Therefore, the DO has no solutions of the form (4.1) at $\{x_0 \mid {x_0}^2+1=0\}$.

Example. Consider the DE

$$3\,x^2\,D^2\,y(x) + 2\,x\,(x^2+5)\,D\,y(x) + (2\,x+2)\,y(x) = 0$$

taken from [66]. The corresponding RE at point 0 is

$$(r+2)\,(3\,r+1)\,a_n + 2\,a_{n-1} + (2\,r-4)\,a_{n-2} = 0.$$

The RE has symmetry number 1 and rank 2. We solve $\{(\beta+2)\,(3\,\beta+1)=0, (2\,(\beta+n+2)-4)=0\}$. There is only one solution with a natural n: $\beta=-2$ and $n=2$. We substitute $y(x) = a_{-2}\,x^{-2} + a_{-1}\,x^{-1} + a_0$ into the DE. The linear system for the coefficients has one nonzero solution vector and one solution of the DE is

$$y(x) = \frac{x^2+x+1}{x^2}.$$

Example. $0 = L\,y(x) = 3\,(x-1)^2x^4(6-12\,x-x^2+8\,x^3-7\,x^4)\,D^2\,y(x) + 2\,x\,(-18+72\,x-45\,x^2-180\,x^3+332\,x^4-134\,x^5-81\,x^6+100\,x^7-28\,x^8)\,D\,y(x) + 2\,(x-1)(-6-18\,x+37\,x^2+87\,x^3-16\,x^4-28\,x^5+9\,x^6+7\,x^7)\,y(x)$

The corresponding RE at any point $x_0 \in \bar{\mathbb{K}}$ is

$$\begin{aligned} 0 \;&=\; 3\,r\,(r-1)\,{x_0}^4\,(x_0-1)^2\,(7\,{x_0}^4 - 8\,{x_0}^3 + {x_0}^2 + 12\,x_0 - 6)\,a_n + \\ &\;\;\vdots \\ &\;\;7\,(3\,r^2 - 55\,r + 248)a_{n-10}. \end{aligned}$$

Reducing this RE mod$(A(x_0))$, where $A(x_0) = 7\,{x_0}^4 - 8\,{x_0}^3 + {x_0}^2 + 12\,x_0 - 6$, we construct the RE at $\{x_0 \mid A(x_0)=0\}$:

$36\,r\,(r-2)\,(83744\,{x_0}^3 - 139415\,{x_0}^2 + 73002\,x_0 - 11742)\,a_n + \cdots + 33614\,(r^2\,(105\,x_0-33) - r\,(1533\,x_0+461) + 5488\,x_0 - 1578)\,a_{n-8} + 117649\,(3\,r-28)(r-7)\,a_{n-9} \;=\; 0.$

The RE has symmetry number 1 and rank 9. We solve $\{\beta\,(\beta-2)=0, (3\,(\beta+n+9)-28)\,(\beta+n+9-7)=0\}$. There is no solution with a natural number n. Therefore, the DE has no solutions of the form (4.1) at $\{x_0 \mid A(x_0)=0\}$.

4.2 Polynomial solutions

The search for polynomial solutions $\in \mathbb{K}[x]$ is a special case of the search for finite Laurent series solutions with $\lambda = 0$ and $\alpha \in \mathbb{N}$. The reader is referred to lemma 4.1. Polynomial solutions are needed in many applications and, therefore, are the subject of current research [1]. The best solution at this moment is an algorithm of Barkatou [5], which especially shows his advantages for polynomial solutions of high degree, although with some programming effort.

For our purposes the following algorithm is efficient enough (and is much faster than the standard procedures in Maple, see chapter 8):

Algorithm 2 Polynomial_Solutions*(L)*
(The algorithm computes all polynomial solutions for the differential operator L (1.1).)

```
1. Compute the symmetry number $h$, the indicial polynomial $p_0$, the rank and $p_{rank}$ at the point $x_0$.
2. Solve the system of two polynomial equations $\{\mathrm{pp}(p_0(\alpha), r) = 0,\ \mathrm{pp}(p_{rank}(\alpha + n\,h + rank), r) = 0\}$ for $\{\beta, n\}$ with the conditions $\alpha, n \in \mathbb{N}$. Construct the set of all solution pairs $\mathcal{P} = \{(\alpha_i, \alpha_i + n_i\,h)\}$.
3. $\{y_k\} := \emptyset$
   for each $(\alpha, \omega) \in \mathcal{P}$
      (a) Substitute $y(x) = a_\alpha\,x^\alpha + a_{\alpha+h}\,x^{\alpha+h} + \cdots + a_\omega\,x^\omega$ with free coefficients $a_\alpha, a_{\alpha+h}, \ldots, a_\omega$ into $L\ y(x) = 0$. Solve the linear equation system for the coefficients. Let $\{\bar{\mathbf{a}}_k\}$ be a set of nontrivial linearly independent solution vectors.
      (b) for each $\bar{\mathbf{a}}_k = (\hat{a}_\alpha, \hat{a}_{\alpha+h}, \ldots, \hat{a}_\omega)^T$
             $\{y_k\} := \{y_k\} \cup \{\hat{a}_\alpha\,x^\alpha + \hat{a}_{\alpha+h}\,x^{\alpha+h} + \cdots + \hat{a}_\omega\,x^\omega\}$
4. return $\{y_k\}$.
```

4.3 Rational solutions

In this section we search for rational solutions $y(x) \in \mathbb{K}(x)$ of DEs (1.1). Remembering lemma 4.1 we conclude that, if x_0 is a root of the denominator of a rational solution with multiplicity k, then $-k$ is a root of the indicial polynomial in point x_0. From lemma 2.4 we know that the indicial polynomial has only natural roots, if x_0 is a regular point. Therefore, if x_0 is a root of the denominator of a rational solution, then x_0 is a singular point of (1.1).

We can restrict our search for negative indices to irreducible over $\mathbb{K}$ factors of q_{ord}, as if we have such factor s and decompose it into linear factors $s = (x - x_1)(x - x_2)\ldots(x - x_k)$, the multiplicity of each root x_i, $i = 1\ldots k$ in the denominator is the same. Let us remark that we even might use only balanced factorization (definition 18) as done in [2].

Thus, the idea of the algorithm is: We factor q_{ord} over $\mathbb{K}$. If we have an irreducible factor $s(x)$, we construct the indicial polynomial at $\{x_0 \mid s(x_0) = 0\}$. From the smallest negative integer index we construct the denominator of a candidate for rational solutions. Then, we transport the problem of finding rational solutions to the problem of finding polynomial solutions.

Algorithm 3 Rational_Solutions*(L)*
(The algorithm computes all rational solutions for the differential operator L (1.1).)

```
1. Make $q_{ord}$ squarefree and factor the result over the coefficient field $\mathbb{K}$. Let $\mathcal{S}$ be the set of factors.
2. $d := 1$
   for each $s \in \mathcal{S}$
      (a) Compute the indicial polynomial $p_0$ at $\{x_0 \mid s(x_0) = 0\}$.
      (b) Find the smallest negative integer root of $p_0(r)$.
          if such root $r$ exists then
              $d := d\, s^r$
3. Substitute $y(x) = y_1(x)\, d$ in $L$. Let $M$ be the resulting DO with $M\, y_1(x) = 0$.
4. $\{f_k\} :=$ Polynomial_Solutions($M$);
5. $\{y_k\} := \emptyset$
   for each $f_k \in \{f_k\}$
       $\{y_k\} := \{y_k\} \cup \{f_k\, d\}$
6. return $\{y_k\}$.
```

Practical experience shows, the algorithm of Barkatou [5] is more efficient then the above algorithm or [2]. Especially in the cases, when the step size between two nonzero terms of the Laurent series solution is large (as, e.g., in $\frac{1-x^{13}+x^{145}}{x}$), the algorithm of Barkatou shows its advantages, because it needs to compute only the coefficients of the nonzero terms. Arguments in favor for **Rational_Solutions** are its simplicity, robustness and satisfactory efficiency. Finding rational solutions has its own worth, but rational solutions are also useful for bigger classes of solutions.

Definition 17 *[62, page 59], [63] Let be given a homogeneous linear differential operator L (1.4). Then, for any positive integer m, the m-th symmetric power of L, denoted $L^{Ⓢm}$ is defined to be the monic homogeneous linear differential operator of the type (1.4) with minimal order whose solution space is spanned by y^m, where y is an arbitrary solution of L:*

$$\forall y: \ (L\ y = 0) \to (L^{Ⓢm}\ y^m = 0).$$

The rational solutions of $L^{Ⓢm}$ provide explicitely the Liouvillian solutions of second order equations [69], and are needed as a first step in computing the Liouvillian solutions of higher-order DEs [64].

4.4 Beke's algorithm

The algorithm of Beke [11] finds all exact solutions y of the form

$$(\ln(y))' = y'/y = u \in \bar{\mathbb{K}}(x) \tag{4.3}$$

i.e.

$$y = \exp\left(\int u\ dx\right) \tag{4.4}$$

for any given DE (1.1). We call functions of this form *functions with logarithmic derivative* in $\bar{\mathbb{K}}(x)$ or *exponential functions* over $\bar{\mathbb{K}}$. If the algorithm finds an exponential solution, then

$$L = Q\ (D - u)\,,$$

where $Q \in \bar{\mathbb{K}}(x)[D]$, i.e. L is reducible over $\bar{\mathbb{K}}(x)$.

Exponential solutions are needed for the algorithms of Kovacic and Singer. The special case of finding solutions y whose logarithmic derivatives y'/y are in $\mathbb{K}(x)$ is one of the essential parts of the *associated equations method* [11, 56, 76, 17].

Let $y(x) = \exp(\int u\ dx)$, $u \in \bar{\mathbb{K}}(x)$, be an exponential solution of (1.1). Let $\mathbb{K}_b(x) \subset \bar{\mathbb{K}}(x)$ be a differential field with the minimal number of algebraic extensions $\theta_1, \theta_2, \cdots, \theta_s \in \bar{\mathbb{K}}$, such that $u \in \mathbb{K}_b(x)$. Using Hermite reduction and the Rothstein/Trager method [29] we can conclude

$$\begin{aligned} y(x) &= s \exp\Big(p + \frac{c}{d}\Big) \prod_i {r_i}^{c_i}, \\ & \qquad p, c, d \in \mathbb{K}_b[x],\ c_i \in \bar{\mathbb{K}},\ s, r_i \in \mathbb{K}_b(c_i)[x]. \end{aligned}$$

The quantity p is called the *polynomial part.* The partial fractions of $\frac{c}{d}$ are said to be the *irregular singular parts* because each root of d is an irregular singular point of (1.1). We define

$$t = s \prod_{i\,:\ c_i \in \mathbb{N}} {r_i}^{c_i} \tag{4.5}$$

and obtain

$$\begin{aligned} y(x) &= t \exp\Big(p + \frac{c}{d}\Big) \prod_i {r_i}^{c_i}, \\ & \qquad p, c, d \in \mathbb{K}_b[x],\ c_i \in \bar{\mathbb{K}} \setminus \mathbb{N},\ t, r_i \in \mathbb{K}_b(c_i)[x]. \end{aligned} \tag{4.6}$$

The ${r_i}^{c_i}$ are said to be the *singular parts* because each root of any r_i is a singular point of (1.1). In order to minimize the number of singular parts we excluded the case $c_i \in \mathbb{N}$.

Singer presented a procedure which decides if a given DE (1.1) has exponential solutions (4.4) and if so, finds such a solution [58, proposition 3.5]. Other authors had similar ideas [43, section 3], [56, section 3]. Bronstein proposed an algorithm, which introduces algebraic extensions of $\mathbb{K}$ only if they appear in the potential solutions, and not in the singularities of the DE as was previously required. Instead of complete factorization of q_{ord} or factorization over $\mathbb{K}$ he used balanced factorization [2]. As van Hoeij mentioned in his thesis [70, section 3.4], the differences between these algorithms are small enough to call it Beke's method.

Our algorithm is based on the original algorithm [11], too. We include results from Abramov, Bronstein, Schwarz and Singer. The algorithm has the following basic strategy:

Step 1 Bound the degree of the polynomial part.

Step 2 Determine the denominators of the irregular singular parts.

Step 3 Determine the singular parts. We need the singular parts in order to be able to bound the degree of t.

Step 4 For all members of the set of possible combinations of singular parts determine the corresponding t and the coefficients in p and c simultaneously.

Our algorithm differs from those of other authors in the following points:

We mainly handle corresponding recurrence equations (REs) for the given DEs. This was done, because our main interest in Beke's algorithm came from finding approximate exponential solutions and instead of the search of an exact polynomial t in step 4 we now can apply, e.g., Frobenius method, which needs the recurrence equation.

In contrast to [16, 15], we don't evaluate candidates for the polynomial part in step 1. We made this decision because it is not always possible to get the coefficients in t as radicals (consider, e.g., $y^{(5)}(x) + 2\,y(x) = 0$). In general one has to describe the coefficients in t, p, c (and the c_i) as algebraic extensions of $\mathbb{K}$. Any implementation of an algorithm, which computes (exact or approximate) exponential solutions of DEs (1.1), should be able to calculate in algebraic number fields. If $\mathbb{K}_b = \mathbb{Q}$, our algorithm should do it only in the last step 4.

We excluded the case $c_i \in \mathbb{N}$. If we did not, then in step 4 we would find any possible exponential solution (4.4) at least twice: once in the case when indeed the corresponding ${r_i}^{c_i}$ is a part of a candidate in the set of possible combinations of singular parts and once when it is not and hides in t. This would double the computation.

4.4.1 Bounding the degree of the polynomial part

Lemma 4.2 *[56, bound 3], [15, definition 3, theorem 1] Let (4.4) be a solution of (1.1) and*

$$\mathcal{S} \;:=\; \left\{ s_{j,k} = \frac{\deg(q_j, x) - \deg(q_k, x)}{k - j} \,\middle|\, j,k \in \mathbb{N} \wedge 0 \le j < k \le ord \wedge \forall l \in \mathbb{N},\right.$$
$$\left. 0 \le l \le ord : \deg(q_l, x) - \deg(q_k, x) \le (k - l)\, s_{j,k} \right\} \bigcap \mathbb{N}.$$

If $\mathcal{S} = \emptyset$ *then* $p = 0$ *else* $\deg(p, x) \le \max(\mathcal{S}) + 1$.

Continued example. $L = 3\,(x-1)^2 x^4 (6 - 12\,x - x^2 + 8\,x^3 - 7\,x^4)\,D^2 + 2\,x\,(-18 + 72\,x - 45\,x^2 - 180\,x^3 + 332\,x^4 - 134\,x^5 - 81\,x^6 + 100\,x^7 - 28\,x^8)\,D + 2\,(x-1)(-6 - 18\,x + 37\,x^2 + 87\,x^3 - 16\,x^4 - 28\,x^5 + 9\,x^6 + 7\,x^7)$

We get $\mathcal{S} = \emptyset$ and $p = 0$.

We could also get this bound using Newton polygon techniques. Then, it comes out, that $\mathcal{S}$ is the set of natural-number slopes of the Newton polygon of the differential operator at the point infinity.

4.4.2 Determining the denominators of the irregular singular parts

In this section we benefit from the results of Abramov and Bronstein [2, 16, 15].

Let $P, Q \in \mathbb{K}[x]$, $\deg(P, x) > 0$. The *order* of Q at P, denoted by $\nu_P(Q)$, is the natural number m for which $P^m \mid Q$ and $P^{m+1} \not| \, Q$.

Definition 18 *Let* $A, B \in \mathbb{K}[x]$.
A *is balanced w.r.t.* $B \overset{\text{def}}{\Longleftrightarrow} \nu_P(B) = \nu_Q(B)$ *for any two irreducible factors* $P, Q \in \mathbb{K}[x]$ *of* A.

Definition 19 *Let* $A \in \mathbb{K}[x]$ *and* $\mathcal{Q} \subseteq \mathbb{K}[x]$.
$A = A_1^{l_1} \cdots A_m^{l_m}$ *is a balanced factorization w.r.t.* $\mathcal{Q} \overset{\text{def}}{\Longleftrightarrow}$ *each* A_i *is squarefree and balanced w.r.t.* B *for any* $B \in \mathcal{Q}$ *and* $\forall i \neq j : \gcd(A_i, A_j) = 1$.

An algorithm for balanced factorization was given in [13]. It only requires gcd computations in $\mathbb{K}[x]$ and is at most as expensive as computing a square-free-gcd-basis for $\{A\} \cup \mathcal{Q}$.

Theorem 4.3 [15, theorem 1] *Let q_{ord} be monic and $q_{ord} = A_1^{l_1} \cdots A_m^{l_m}$ be a balanced factorization with respect to $\mathcal{Q} = \{q_0, \ldots, q_{ord}\}$. Let (4.4) be a solution of (1.1). Then (4.3) can be written as*

$$u = \frac{B_1}{A_1^{\delta(\mathcal{S}_{A_1})}} + \cdots + \frac{B_m}{A_m^{\delta(\mathcal{S}_{A_m})}} + \frac{t'}{t} + p',$$

where the conditions from lemma 4.2 holds for p, $B_1, \ldots, B_m$, $t \in \bar{\mathbb{K}}[x]$, $\forall i : \deg(B_i, x) < \delta(\mathcal{S}_{A_i}) \deg(A_i, x)$ and

$$\mathcal{S}_{A_i} \quad := \quad \left\{ \frac{\nu_{A_i}(q_k) - \nu_{A_i}(q_j)}{k - j} \,\middle|\, \forall (j,k) : 0 \leq j < k \leq ord \right\} \bigcap \left(\mathbb{N} \setminus \{0\} \right).$$

If $\mathcal{S}_{A_i} = \emptyset$ then $\delta(\mathcal{S}_{A_i}) = 1$ else $\delta(\mathcal{S}_{A_i}) = \max(\mathcal{S}_{A_i})$.

Balanced factorization not only reduces the cost of factoring q_{ord}, but also reduces the number of singular parts to be checked.

We could also get this bound using Newton polygon techniques. Then, it comes out, that the $\delta(S_{A_i})$ are the maximal positive natural-number slopes in *p-adic Newton polygons* (i.e. modulo A_i).

Theorem 4.3 points out that each exponential solution (4.4) can be written as

$$y = \exp \int \frac{B_1}{A_1^{\delta(\mathcal{S}_{A_1})}} + \cdots + \frac{B_m}{A_m^{\delta(\mathcal{S}_{A_m})}} + \frac{t'}{t} + p'.$$

We apply Hermite reduction and get

$$\begin{aligned} y \quad &= \quad t \exp \left(p + \frac{C_1}{A_1^{\delta(\mathcal{S}_{A_1})-1}} + \cdots + \frac{C_m}{A_m^{\delta(\mathcal{S}_{A_m})-1}} + \int \frac{D_1}{A_1} + \cdots + \int \frac{D_m}{A_m} \right), \qquad (4.7) \\ & t, p, C_1, \ldots, C_m, D_1, \ldots, D_m \in \bar{\mathbb{K}}[x], \ \deg(p, x) \leq \max(\mathcal{S}) + 1, \\ & \forall i : \deg(D_i, x) < \deg(A_i, x), \ \forall i : \deg(C_i, x) < (\delta(\mathcal{S}_{A_i}) - 1) \deg(A_i, x). \end{aligned}$$

Continued example. The balanced factorization of q_{ord} is

$$\begin{aligned} \frac{1}{21} q_2 \quad &= \quad A_1 \, A_2{}^2 \, A_3{}^4, \\ A_1 \quad &= \quad -\frac{6}{7} + \frac{12}{7} x + \frac{1}{7} x^2 - \frac{8}{7} x^3 + x^4, \\ A_2 \quad &= \quad x - 1, \\ A_3 \quad &= \quad x. \end{aligned}$$

We compute

i	$\nu_{A_i}(q_0)$	$\nu_{A_i}(q_1)$	$\nu_{A_i}(q_2)$	$\mathcal{S}_{A_i}$	$\delta(\mathcal{S}_{A_i})$
1	0	0	1	$\{1\}$	1
2	1	0	2	$\{2\}$	2
3	0	1	4	$\{1, 3\}$	3

Thus, after formal Hermite reduction, each exponential solution can be written as

$$\begin{aligned} y(x) &= t \exp\left(\frac{\zeta_1}{x-1} + \frac{\zeta_2 x + \zeta_3}{x^2} + \int \frac{\zeta_4}{x-1} + \int \frac{\zeta_5}{x} + \int \frac{B_1}{6-12x-x^2+8x^3-7x^4}\right), \\ & \quad \zeta_i \in \bar{\mathbb{K}},\ t, B_1 \in \bar{\mathbb{K}}[x]. \end{aligned}$$

4.4.3 Determining the singular parts

In this section we determine all singular parts and rules for the irregular singular parts. We know from theorem 4.3 that the singular parts of an exponential solution for L (1.1) originate from the $\exp\left(\int D_j/A_j\right)$ where the A_j are factors from the balanced factorization of q_{ord}, $j = 1(1)m$, m is the number of balanced factors and $D_j \in \bar{\mathbb{K}}[x]$, $\deg(D_j, x) < \deg(A_j, x)$. We can distinguish two cases:

- All roots of A_j are regular singularities of (1.1).
- All roots of A_j are irregular singularities of (1.1).

First, let us consider the regular singular case. Then, at $\{x_0 \mid A_j(x_0) = 0\}$ each exponential solution can be written in the form of a Laurent series

$$\begin{aligned} y(x) &= (x-x_0)^{\lambda} \sum_{n=\alpha}^{\infty} a_n\,(x-x_0)^n, \qquad (4.8) \\ & \alpha, \omega \in \mathbb{Z},\ x_0, a_n, \lambda \in \bar{\mathbb{K}},\ a_\alpha \neq 0,\ a_\omega \neq 0. \end{aligned}$$

By lemma 4.1 we can build up the singular parts from the indices of L at $\{x_0 \mid A_j(x_0) = 0\}$.

Now let us consider the more complicated irregular case. We choose an index j for which L is irregular at $\{x_0 \mid A_j(x_0) = 0\}$ and denote $C_j = \zeta_0 + \zeta_1\, x + \cdots + \zeta_m\, x^m$. We transform (4.7) into

$$y(x) = \varphi(x) \exp\left(\frac{C_j}{A_j^{\delta(\mathcal{S}_{A_j})-1}}\right). \qquad (4.9)$$

If we compare (4.7) with (4.9), we note that $\varphi(x)$ can be written at $\{x_0 \mid A_j(x_0) = 0\}$ in the form of a Laurent series (4.8). We substitute (4.9) into the original DE (1.1). We get a DE

$$\tilde{L}\ \varphi(x) = 0. \qquad (4.10)$$

Note that $\tilde{L}$ contains parameters $\zeta_0, \zeta_1, \ldots, \zeta_m$. If we knew the values of the parameters, by lemma 4.1 we could build up the singular parts from the indices of $\tilde{L}$ at $\{x_0 \mid A_j(x_0) = 0\}$.

Thus, the goal of our algorithm **Singular_Parts** it to produce all sets of rules for the $\zeta_0, \zeta_1, \ldots, \zeta_m$, for which the indicial polynomial p_0 of $\tilde{L}$ has roots (for which $\deg(p_0, r) > 0$). If we find one solution for the rules, we construct the singular part from the corresponding indicial polynomial.

For the present we have decided to implement the algorithm given in [15, section 4.3], but with some modifications. We observed that occasionally the Singer/Bronstein procedure computes too many useless candidates and if $\deg(A_j(x), x) > 3$ then the procedure sometimes produces a system of polynomial equations, which existing solvers cannot solve in reasonable time. As far as possible we do not apply the Singer/Bronstein procedure and substitute it by "simpler" methods.

Let $P, Q \in \mathbb{K}[x]$. We introduce the notation $\operatorname{rem}(P, Q)$ for the remainder of P modulo Q.

Algorithm 4 Singular_Parts$(\tilde{L}, A_j, rules)$
(The algorithm computes recursively all possible pairs of singular parts and rules for the coefficients $\zeta_0, \zeta_1, \dots, \zeta_m$ of the irregular singular parts for any DE of the form (1.1) at $\{x_0 \mid A_j(x_0) = 0\}$. $\hat{L}$ is derived from the substitution of (4.9) into (1.1) and may contain parameters $\zeta_0, \zeta_1, \dots, \zeta_m$. Start the recursion with rules $= \emptyset$.)

```
1. $\mathcal{M}_j := \emptyset$; $g := \deg(A_j(x), x)$; Let $\tilde{L} = \sum_{i=0}^{ord} \tilde{q}_i\, y^{(i)}(x)$. Construct the indicial polynomial
   $p_0 = a_0 + a_1\, r + \cdots + a_{ord}\, r^{ord}$ of $\tilde{L}$ at $\{x_0 \mid A_j(x_0) = 0\}$.
2. if $p_0(r)$ has at least one root in $\bar{\mathbb{K}} \setminus \mathbb{N}$ then
   (a) if $\forall (c_i, c_j) : p_0(c_i) = p_0(c_j) = 0 \wedge c_i - c_j \in \mathbb{Z}$ then
           Compute the index $c$ with the minimal real part.
           $\mathcal{M}_j := \mathcal{M}_j \cup \left(\left\{A_j(x)^c\right\}, rules\right)$
   (b) else if $g = 1$ or $\deg(\mathrm{pp}(p_0, r), x_0) = 0$ and the set of indices contains only one
       element then
           for each index $c \in \bar{\mathbb{K}} \setminus \mathbb{N} :\ p_0(c) = 0$
               $\mathcal{M}_j := \mathcal{M}_j \cup \left(\left\{A_j(x)^c\right\}, rules\right)$
   (c) else (This is the special case $\delta = 1$ of the procedure in [15].) Compute
               $n_1 := \max_{0 \le k \le ord,\ \tilde{q}_j \neq 0} k - \nu_{A_j}(\tilde{q}_k);$
       $D_j := \sum_{k=0}^{g-1} \xi_k\, x^k$ where the $\xi_k$ are free coefficients,
               $H_1 := \sum_{k=0}^{ord} \mathrm{rem}\left(\frac{\tilde{q}_k}{A_j^{\nu_{A_j}(\tilde{q}_k)}}, A_j\right) \prod_{l=0}^{k-1} D_j - l\, A_j{}'$
       It follows that $H_1 \in \mathbb{K}(\xi_0, \dots, \xi_{g-1})[x]$. Compute the coefficients of $H_1 \bmod(A_j)$.
       Equating the coefficients to 0, we get a system of polynomial equations for the
       $\xi_0, \dots, \xi_{g-1}$. Solve this system. Let $\mathcal{L}$ be the set of solutions.
       for each $(\tilde{\xi}_0, \dots, \tilde{\xi}_{g-1}) \in \mathcal{L}$
           $\tilde{D}_j := \sum_{k=0}^{g-1} \tilde{\xi}_k\, x^k$;
           if all roots of $p_0(r)$ are in $\bar{\mathbb{K}} \setminus \mathbb{N}$ or $\mathrm{res}(\tilde{D}_j(x) - z\, A_j{}'(x), A_j(x), x) \in \mathbb{K}(\tilde{\xi}_0, \dots,$
           $\tilde{\xi}_{g-1})[z]$ has at least one root in $\bar{\mathbb{K}} \setminus \mathbb{N}$ then
               $\mathcal{M}_j := \mathcal{M}_j \cup \left(\left\{\exp \int \frac{\tilde{D}_j}{A_j}\, dx\right\}, rules\right)$
3. $rules := rules \cup \{a_0 = 0, a_1 = 0, \dots, a_{ord} = 0\}$;
   if rules is solvable for $\zeta_0, \zeta_1, \dots, \zeta_m$ then
       Apply rules to $\tilde{L}$ and recompute it.
       $\mathcal{M}_j := \mathcal{M}_j \cup$ Singular_Parts$(\tilde{L}, A_j, rules)$
4. return $\mathcal{M}_j$.
```

In the case of irregular singularities, the recursion is started with $\tilde{L}$ from (4.10) and $rules = \emptyset$. **Singular_Parts** can be applied to regular singular parts A_j too. In this case, start the recursion with **Singular_Parts**$(L, A_j, \emptyset)$. Then, L has no parameters and the recursion stops after one step. Let us now discuss the algorithm.

In step 1 we compute the indicial polynomial. We only search for those singular parts

$$\exp\left(\int D_j/A_j\right) = \prod_i {r_i}^{c_i},$$

for which at least one index c_i is not a natural number. Any DE (1.1) of order ord has at any regular point the indices $0, 1, \ldots, ord - 1$. Only if p_0 has roots in $\bar{\mathbb{K}} \setminus \mathbb{N}$ it is worth searching for singular parts. This way the test in the beginning of step 2 lessens the number of calls to the Singer/Bronstein procedure and, consequently, lessens the number of singular parts.

If all roots of the indicial polynomial p_0 at $\{x_0 \mid A_j(x_0) = 0\}$ differ by integers (an important subcase is: if all roots *are* integers), then we take in step 2a only that singular part ${A_j}^c$, for which c is the index with the minimal real part. This can be done because each other singular part $\exp\left(\int D_j/A_j\right)$ is the product of a polynomial and ${A_j}^c$.

Roughly speaking, the singular parts are built by combining the roots of the indicial polynomial with the roots of A_j in all possible ways. If $\deg(A_j(x), x) = 1$ or $\deg(\mathrm{pp}(p_0, r), x_0) = 0$ and the set of indices contains only one element then the way to combine them in step 2b is predetermined.

In the steps 2a and 2b we also don't really solve $p_0(r) = 0$ in terms of radicals, but add the indicial polynomial to the rules if it has roots in $\bar{\mathbb{K}} \setminus \mathbb{K}$. This way we avoid heavy computation at a moment, where we don't know whether the candidate for the singular part is in fact needed in the solution, or not.

If the reader is familar with [15, section 4.3] then he will notice that in step 2c we only use the Singer/Bronstein procedure in the special case $\delta = 1$. Proofs of the correctness of the Singer/Bronstein procedure are given in [60, proposition 2.3], [16, theorem 8.4].

We added a test in step 2c, where we check, if our singular part really produces non-natural indices. If p_0 has roots in $\bar{\mathbb{K}} \setminus \mathbb{N}$ and roots in $\mathbb{N}$ too, the singular parts given by the Singer/Bronstein procedure sometimes have the property that all the c_i are natural numbers. These cases are excluded by a test, taken from the Rothstein/Trager method [29]. This test lessens again the number of singular parts.

Unfortunately, it is not always possible to solve the system of polynomial equations for the coefficients $\xi_0, \ldots, \xi_{g-1}$ in terms of radicals. In general one has to describe the $\xi_0, \ldots, \xi_{g-1}$ as algebraic extensions of $\mathbb{K}$. If such a case appears, we leave the coefficients undetermined and add the system of polynomial equations, which describes the coefficients, to the rules. This is another part of our "lazy evaluation" strategy.

In step 3 we add the coefficients of the indicial polynomial to the rules for the parameters $\zeta_0, \zeta_1, \ldots, \zeta_m$. Our Maple implementation stores the rules in form of pure lexicographic Gröbner bases, but other forms are possible. We don't "solve" rules explicitly, but we have to decide whether rules is algebraically consistent, i.e. has at least one solution. If this is the case, then the parameters $\zeta_0, \zeta_1, \ldots, \zeta_m$ take values such that all a_i are 0, the "old" indicial polynomial vanishes and hence the "new" indicial polynomial needs to be recomputed. Before we call **Singular_Parts** again, we apply $rules$ to $\tilde{L}$. In terms of our Maple implementation, we compute the reduced form of the coefficients of $\tilde{L}$ with respect to the ideal basis $rules$ and pure lexicographic term order.

The algorithm **Singular_Parts** is called recursively while rules is solvable.

We apply to each A_j from the balanced factorization of q_{ord}. Then we combine the $\mathcal{M}_j$ in all possible ways. This is done in algorithm **Exponential_Solutions**, step 3. We denote the set of all possible combinations by $\mathcal{M}$.

Continued example. We have already noted that the DE is regular singular at $\{x_0 \mid 6 - 12\,x_0 - {x_0}^2 + 8\,{x_0}^3 - 7\,{x_0}^4 = 0\}$ with $\mathrm{pp}(p_0, r) = r\,(r-2)$. The indices are natural numbers that one can ignore. This is one reason why our algorithm is sometimes more efficient than the algorithm proposed in [15], which does not check whether the special case $\delta = 1$ of the procedure in [15] applies or not. If we omit the check then we get a system of polynomial equations, which is hard to solve and gives a lot of unnecessary singular parts with algebraic extensions of degree up to 6. (This also seems to be the reason why the Maple V Release 4 procedure 'dsolve/diffeq/expsols' fails on this example.)

The DE is irregular singular at $\{x_0 \mid 1 - x_0 = 0\}$. After substitution of

$$y(x) = \varphi(x)\ \exp\left(\frac{\zeta_1}{x-1}\right)$$

we get the DE $\tilde{L}\ \varphi(x) = 0$. The corresponding RE at $\{x_0 \mid x_0 - 1 = 0\}$ is

$$0 = 18\,\zeta_1(\zeta_1+2)\,a_n + \cdots + 7\,(3\,r-31)(r-8)\,a_{n-10}.$$

If $\zeta_1(\zeta_1+2) \neq 0$ then $\deg(p_0, r) = 0$. If $\zeta_1(\zeta_1+2) = 0$, we get

$$0 = -36\,(\zeta_1+1)\,r\,a_n + \cdots + 7\,(3\,r-28)(r-7)\,a_{n-9}.$$

The root of $\mathrm{pp}(p_0, r)$ is a natural number. The ruleset for the next recursion

$$rules := \{\zeta_1(\zeta_1+2) = 0\} \cup \{\zeta_1 + 1 = 0\}$$

is not solvable and the recursion is finished. Consequently, each exponential solution of the DE can be written as

$$\begin{aligned} y(x) \quad &= \quad t\ \exp\left(\frac{\zeta_1}{x-1} + \frac{\zeta_2 x + \zeta_3}{x^2} + \int \frac{\zeta_5}{x}\right), \\ & \qquad \zeta_i \in \bar{\mathbb{K}},\ t \in \bar{\mathbb{K}}[x]. \end{aligned}$$

4.4.4 Determining t and the coefficients in p and c

In the previous section we defined by $\mathcal{M}$ the set of all pairs (combination of singular parts, rules for the irregular singular part) to a given DE (1.1). We were able to compute $\mathcal{M}$. Now, we choose a point $x_0 \in \mathbb{K}$ and note that each exponential solution $y(x)$ (4.6) of an ODE with polynomial coefficients (1.1) can be written as

$$y(x) = \varsigma(x)\,f\ \exp(p + c/d), \tag{4.11}$$

where there exists a combination of singular parts f in $\mathcal{M}$ such that $\varsigma(x)$ can be expanded into the form (4.1) at the point x_0. In other words, if we compare (4.6) with (4.11), we get

$$t\prod_i {r_i}^{c_i} = \varsigma(x)\,f,$$

i.e. $\varsigma(x)$ is one singular part, which corresponds to the chosen point x_0, multiplied by the polynomial t. We substitute (4.11) into (1.1) and obtain a DE

$$\hat{L}\ \varsigma(x) = 0. \tag{4.12}$$

$\hat{L}$ can be transformed again to the form (1.1), because $f\ \exp(p + c/d)$ is an exponential function and substitutions in (1.1) of the dependent variable y by a new dependent variable multiplied with an exponential function give again linear ODEs with rational coefficients.

Thus, we have reduced the question of finding exponential solutions to the less complex question of finding finite Laurent series solutions $\varsigma(x)$ at the point x_0. But $\hat{L}$ contains coefficients ζ_l which come from the polynomials p and c. Therefore, we have to modify algorithm **Finite_Laurent_Series_Solutions**.

If $\deg(p_0, r) = 0$ or $\deg(p_{rank}, r) = 0$ then solutions $\varsigma(x)$ for (4.12) cannot exist. We use this property to determine the ζ_l. If p_0 contains free ζ_l then we determine rules for the ζ_l, for which p_0 vanishes. Then we reduce the old RE modulo the rules and get a new RE. We do the same for p_{ord}. We get a tree with nodes of pairs (RE, rules). The rank of the new RE is less than the rank of the old RE. This guarantees that the recursion always stops.

Algorithm 5 Tree*($\hat{L}$,rules,$p + c/d$,x_0)*
(The algorithm computes the set of functions $\varsigma(x)\ \exp(p + c/d)$ for any DE of the form (1.1) at the point $x_0 \in \mathbb{K}$. $\hat{L}$ is derived from the substitution of (4.11) into (1.1) and may contain parameters $\zeta_1, \ldots, \zeta_j$. Start the recursion with known rules for the parameters.)

```
1. $\{y_k\} := \emptyset$. Apply rules. Compute the RE at the point $x_0$. Denote the symmetry
   number by $h$ and the rank by $rank$. Suppose $p_0 = a_0 + a_1\, r + \cdots + a_{ord}\, r^{ord}$ and
   $p_{rank} = b_0 + b_1\, r + \cdots + b_{ord}\, r^{ord}$.
2. Define $\beta = \lambda + \alpha$. Solve $\{\mathrm{pp}(p_0(\beta), r) = 0,\ \mathrm{pp}(p_{rank}(\beta + n\,h + rank), r) = 0\} \cup rules$ for
   $\{\beta, n\}$ with the condition $n \in \mathbb{N}$. Construct the set of all solutions $\mathcal{M} = \{\beta_i, \beta_i + n_i\, h)\}$.
3. for each $(\beta, \gamma) \in \mathcal{M}$
   (a) Substitute $y(x) = a_\beta\,(x - x_0)^\beta + a_{\beta+h}\,(x - x_0)^{\beta+h} + \cdots + a_\gamma\,(x - x_0)^\gamma$ with free
       coefficients $a_\beta$, $a_{\beta+h}$, ..., $a_\gamma$ into $L\ y(x) = 0$. Solve the system of polynomial
       equations for the coefficients $\zeta_1, \ldots, \zeta_j, a_\beta, a_{\beta+h}, \ldots, a_\gamma$. Let $\mathcal{L}$ be a set of non-
       trivial solutions.
   (b) for each $(\hat{\zeta}_1, \ldots, \hat{\zeta}_j, \hat{a}_\beta, \hat{a}_{\beta+h}, \ldots, \hat{a}_\gamma) \in \mathcal{L}$
       i. $\varphi := \hat{a}_\beta\,(x - x_0)^\beta + \cdots + \hat{a}_\gamma\,(x - x_0)^\gamma$
       ii. Denote by $z$ the result of the substitutions $\zeta_1 \leftarrow \hat{\zeta}_1$, ..., $\zeta_j \leftarrow \hat{\zeta}_j$ in $\exp(p +$
           $c/d)$).
       iii. $\{y_k\} := \{y_k\} \cup \{\varphi\, z\}$
4. $r := rules \cup \{a_0 = 0, a_1 = 0, \ldots, a_{ord} = 0\}$;
   if $r$ is solvable then $\{y_k\} := \{y_k\} \cup \mathbf{Tree}(\hat{L}, r, p + c/d, x_0)$
5. $r := rules \cup \{b_0 = 0, b_1 = 0, \ldots, b_{ord} = 0\}$;
   if $r$ is solvable then $\{y_k\} := \{y_k\} \cup \mathbf{Tree}(\hat{L}, r, p + c/d, x_0)$
6. return $\{y_k\}$.
```

We have already mentioned that we always use Gröbner basis solvers for solving systems of polynomial equations. In fact, we don't really need to "solve" $\mathcal{L}$, whatever this means, but we have to decide whether $\mathcal{L}$ is algebraically consistent, i.e. has at least one solution. If $\mathcal{L}$ is algebraically consistent then (4.11) with parameters in ς, p and c satisfying $\mathcal{L}$, is a solution of (1.1).

One can show that step 5 gives the rules for the coefficients in the candidates for the polynomial part.

We put a summary of the information obtained into the main algorithm:

Algorithm 6 Exponential_Solutions*(L)*
(The algorithm determines all exponential solutions of any given differential operator L (1.1).)

```
1. Compute $\max(\mathcal{S})$. $p := \sum_{j=0}^{\max(\mathcal{S})+1} \zeta_j\, x^j$, where the $\zeta_j$ are free coefficients.

2. Compute the balanced factorization $A_1^{l_1} \cdots A_m^{l_m}$ of $q_{ord}$ w.r.t. $\{q_0, \ldots, q_{ord}\}$.
   $\mathcal{A} := \{A_1, \ldots, A_m\}$.
   for each $A_j \in \mathcal{A}$
       Compute $\delta(\mathcal{S}_{A_j})$.
   $c/d := \dfrac{C_1}{{A_1}^{\delta(\mathcal{S}_{A_1}) - 1}} + \cdots + \dfrac{C_m}{{A_m}^{\delta(\mathcal{S}_{A_m}) - 1}}$ with
   $C_1, \ldots, C_m \in \bar{\mathbb{K}}[x]$, $\forall i : \deg(C_i, x) < (\delta(\mathcal{S}_{A_i}) - 1)\, \deg(A_i, x)$ and the $C_i$ contain free
   coefficients $\zeta_j$.

3. Choose a point $x_0 \in \mathbb{K}$.
   $\mathcal{M} := \{1\}$; $\mathcal{A} := \mathcal{A} \setminus \{x - x_0\}$;
   for each $A_j \in \mathcal{A}$
     (a) if L is irregular singular at $\{x \mid A_j(x) = 0\}$ then
           i. Derive $\tilde{L}$ with $\tilde{L}\ \varphi(x) = 0$ by substitution of
                $y(x) = \varphi(x)\ \exp\left(\dfrac{C_j(x)}{{A_j(x)}^{\delta(\mathcal{S}_{A_j}) - 1}}\right)$
              into L.
          ii. $\mathcal{M}_j :=$ Singular_Parts$(\tilde{L}, A_j, \emptyset)$
         else $\mathcal{M}_j :=$ Singular_Parts$(L, A_j, \emptyset)$
     (b) $\mathcal{F} := \mathcal{M}$;
         for each $((f, rules_f), (g, rules_g)) \in \mathcal{F} \times \mathcal{M}_j$
             if $rules_f \cup rules_g$ is solvable then
                 if $\mathcal{M}_j = \{{A_j}^c\}$, $c \in \mathbb{Z} \setminus \mathbb{N}$ then $\mathcal{M} := \left\{(f\, {A_j}^c, rules_f \cup rules_g)\right\}$
                 else $\mathcal{M} := \mathcal{M} \cup \{(f\, g, rules_f \cup rules_g)\}$

4. $\{y_k\} := \emptyset$;
   for each $(f, rules) \in \mathcal{M}$
     (a) Derive $\hat{L}$ with $\hat{L}\ \varsigma(x) = 0$ by substitution of $y(x) = \varsigma(x)\, f\, \exp(p + c/d)$ into L.
     (b) $\{z_k\} :=$ Tree$(\hat{L}, rules, p + c/d, x_0)$
     (c) for each $z_k \in \{z_k\}$
             $\{y_k\} := \{y_k\} \cup \{z_k\, f\}$

5. return $\{y_k\}$.
```

In step 3 we have to choose a point $x_0 \in \mathbb{K}$. A good choice for x_0 would be a singular point for which the number of elements in the set of singular parts $|\mathcal{M}|$ becomes minimal. Unfortunately, we are not able to predict the best choice without testing all singular points, which would be too expensive. At present in our implementation we fix $x_0 = 0$. This choice is based on the simple observation that many authors like to have a singularity at 0.

Continued example. If we choose the point $x_0 = 0$, step 3a in algorithm **Exponential_Solutions** gives $\mathcal{M} = \{1\}$. We continue and call **Tree**$(\hat{L}, \emptyset, p{+}c/d, 0)$ with substitution of

$$\begin{aligned} y(x) &= \varsigma(x)\, \exp(p + c/d) \\ &= \varsigma(x)\, \exp\left(\frac{\zeta_1}{x-1} + \frac{\zeta_2 x + \zeta_3}{x^2}\right) \end{aligned}$$

into L and $\hat{L}\, \varsigma(x) = 0$. We try now to find a finite Laurent series solution $\varsigma(x)$ at the point 0. At $x_0 = 0$ we construct the corresponding RE for $\hat{L}$, which is

$$0 = -72\, \zeta_3\, (\zeta_3 + 1)\, a_n + \cdots + 7\, (3\, r - 37)(r - 10)\, a_{n-12}.$$

In the next step with $rules = \{\zeta_3\, (\zeta_3 + 1) = 0\}$ we get

$$0 = -36\, \zeta_2\, (2\, \zeta_3 + 1)\, a_n + \cdots + 7\, (3\, r - 34)(r - 9)\, a_{n-11}$$

and in the last step with $rules = \{\zeta_3\, (\zeta_3 + 1) = 0, \zeta_2\, (2\, \zeta_3 + 1) = 0\}$ the RE becomes

$$\begin{aligned} 0 \;=\;& 6\, (12\, r\, \zeta_3 + 6\, r - 3\, {\zeta_2}^2 + 10\, \zeta_3 - 2)\, a_n + \\ & \vdots \\ & 2\, (-54\, r^2 + (-21\, r + 182)(\zeta_1 + \zeta_2) + \\ & 870\, r - 3444)\, a_{n-9} + \\ & 7\, (3\, r - 31)\, (r - 8)\, a_{n-10}. \end{aligned}$$

The symmetry number is 1 and the rank is 10. We solve $\{12\, \beta\, \zeta_3 + 6\, \beta - 3\, {\zeta_2}^2 + 10\, \zeta_3 - 2 = 0, (3\, (\beta + n + 10) - 31)\, (\beta + n + 10 - 8) = 0\} \cup rules$. The solutions with $n \in \mathbb{N}$ are $\mathcal{L} = \{\{\zeta_3 = 0, \zeta_2 = 0, \beta = 1/3, n = 0\}, \{\zeta_2 = 0, \zeta_3 = -1, \beta = -2, n = 0\}\}$.

We take the first solution and substitute $\varsigma(x) = a_{1/3}\, x^{1/3}$ into $\hat{L}\, \varsigma(x) = 0$. We solve the system of polynomial equations and find $\zeta_1 = -2$. Therefore,

$$\begin{aligned} y_1(x) &= \varsigma(x)\, \exp\left(\frac{\zeta_1}{x-1} + \frac{\zeta_2\, x + \zeta_3}{x^2}\right) \\ &= a_{1/3}\, x^{1/3}\, \exp\left(\frac{-2}{x-1}\right) \end{aligned}$$

is a solution of the DE.

We take the second solution, substitute $\varsigma(x) = a_{-2}\, x^{-2}$ into $\hat{L}\, \varsigma(x) = 0$ and find $\zeta_1 = 0$. Therefore,

$$y_2(x) = a_{-2}\, x^{-2}\, \exp\left(\frac{-1}{x^2}\right)$$

is a solution of the DE too. We have now determined the complete fundamental system of the given DE.

4.4.5 The application of Beke's method to approximate solutions

We have mainly handled REs for the given DEs. This allows us to extend the algorithm in a natural way to find formal exact solutions of the form

$$\begin{aligned} y(x) &= \Big(t_0 + t_1 \ln(x - x_0) + \ldots t_{ord-1} \ln(x - x_0)^{ord-1}\Big) \exp\Big(p + \frac{c}{d}\Big) \prod_i {r_i}^{c_i}, \quad (4.13) \\ & \quad c_i \in \bar{\mathbb{K}},\ p, c, d, r_i \in \bar{\mathbb{K}}[x], \end{aligned}$$

where the t_i are formal Laurent series (2.4). To do this, the subroutine in **Exponential_Solutions** which searches for finite Laurent series solutions could be replaced by the Frobenius method [26, 21]. We obtain DESIR-like solutions [4, 52] around regular and regular singular points. Together with the simple test in section 3.1, we are also able to find all series solutions (2.3) around irregular singular points. The solutions are exact in the sense that the REs describe exactly the coefficients of the series. The solutions are only formal solutions because ad hoc nothing is known about convergence. If we truncate the Laurent series t_i, we have good candidates for approximate solutions in closed form.

Finally, Beke's method can also be used for finding approximate exponential solutions (4.6) with free parameters. Those solutions are needed in the **Adaption** algorithm of the adaptive approximation method, see next chapter. As done in [16, 15], we determine the set $\mathcal{N}$ of candidates for polynomial parts p and the set $\mathcal{M}$ of candidates for the singular parts $r_i^{c_i}$ with their corresponding irregular singular parts. Then, for each function $f \in \mathcal{N} \times \mathcal{M}$ we multiply f with a monic polynomial t with free coefficients. Our candidates for approximate exponential solutions (4.6) with free parameters are the functions $f\,t$.

Of course, the number of free coefficients in t is important for the accuracy of the approximate solution. By our practical experience we know: in most cases, the adaptive approximation algorithm in this thesis needs only $\deg(t, x) = 2$ or $\deg(t, x) = 3$. With higher degrees, our solutions will be even too complicated.

Then, we apply adaption criterions to the parameters in t. But this is an anticipation of the next chapter and will be explained there.

4.5 An abridged version of Beke's method

While our search for exponential solutions can be expensive when it uses algebraic extensions, we have good experience with an abridged version of Beke's algorithm, which omits the program code of the special case $\delta = 1$ of the Singer/Bronstein procedure in **Singular_Parts**. Under the assumption that computations in the given field $\mathbb{K}$ are fast, e.g., if $\mathbb{K} = \mathbb{Q}$, it quickly filters out at least all exact solutions of the form

$$\begin{aligned} y(x) &= t \exp\Big(p + \frac{c}{d}\Big) \prod_i {r_i}^{c_i}, \\ & \quad p \in \bar{\mathbb{K}}[x],\ c, d \in \mathbb{K}[x],\ -c_i \in \mathbb{N},\ t, r_i \in \mathbb{K}[x] \end{aligned}$$

and it even covers the search for rational solutions. Nevertheless, we don't use this idea for finding approximate solutions, because if we did, then we would not be able to find good approximations to functions with non-integer indices c_i.

4.6 The factorization algorithm of van Hoeij/Singer

Recently a new efficient algorithm has been found [61, section 6.1],[70, 71, 72].

Theorem 4.4 *Let $\mathbb{K}$ be a field of constants with characteristic* 0*. Let $L, r \in \mathbb{K}(x)[D]$ be homogeneous linear differential operators. Let L be monic. Let r be a solution of* $\mathrm{RRem}(L\,r, L) = 0$ *with* $1 \leq \mathrm{order}(r) < \mathrm{order}(L)$*. Let $x_0 \in \mathbb{K}$ be a regular point of L and let $b_1, \ldots, b_{\mathrm{order}(L)} \in \mathbb{K}[[x - x_0]]$ be a basis of formal Taylor series solutions of L in point x_0. Let $z \in \bar{\mathbb{K}}$ be an eigenvalue of the matrix of the linear map $r : \mathbb{K}[[x - x_0]] \to \mathbb{K}[[x - x_0]]$ in the basis $b_1, \ldots, b_{\mathrm{order}(L)}$.*

Then, $\mathrm{GCRD}(L, r - z)$ *is a right hand factor of L, i.e.*

$$\mathrm{RRem}(L, \mathrm{GCRD}(L, r - z)) = 0.$$

Proof. We have that $L\,r$ is divisible on the right by L, i.e. there exist a l that $L\,r + l\,L = 0$. Therefore, if $\hat{y}$ is a solution of $L\,y = 0$, then $r\,\hat{y}$ is again a solution of $L\,y = 0$. This implies that $\hat{y} \to r\,\hat{y}$ is a linear map of the solution space of $L\,y = 0$ into itself. Computing a basis of formal Taylor series in a regular point, we get an easy way to compute the matrix of this map. If z is an eigenvalue of the matrix, then $(r - z)\,y = 0$ and $L\,y = 0$ have a common solution (the eigenvector of r corresponding to z). Since $1 \leq \mathrm{order}(r - z) < \mathrm{order}(L)$, $\mathrm{GCRD}(L, r - z)$ will be a nontrivial factor of L. □

In this way, the problem of finding a right hand factor of L is reduced to the problem of finding a differential operator r with the above mentioned properties.

Let $R, c \in \mathbb{K}(x)[D]$, R be monic and let the order of c be less than the order of L. r is a solution of $\mathrm{RRem}(R\,r, L) = c$ if and only if there exist a $l \in \mathbb{K}(x)[D]$ which is a solution of the *mixed equation* (called *gemischte Gleichung* in [50]) $R\,r + l\,L = c$. In our application, $c = 0$. It was also shown by van Hoeij how to solve the mixed equation.

Note that the only algebraic extension over $\mathbb{K}$ that is used to factor L is the eigenvalue z. This is the main reason, why the factorization algorithm of van Hoeij/Singer is more effective than Beke's method and why it should be preferred for finding exact solutions. On the other hand, the algorithm of van Hoeij/Singer seems to be purely algebraic and the author has no idea how to use the algorithm for finding approximate solutions.

Up to now there is only one implementation of this algorithm - a Maple-Package *diffop* from Mark van Hoeij. At present it cannot solve DEs involving parameters.

Chapter 5

Adaption step - getting candidates

In this chapter we determine the adaptive differential operator $\tilde{L} \in \mathcal{ELF}$ to a given DO L (1.1). In the previous chapter we have shown, how structure theorems of computer algebra can be used to find candidates for approximate exponential solutions $\tilde{y}$ (4.4) for differential operators of the form (1.1). One important property of exponential solutions is that they are solutions of DOs $\tilde{l}$ (1.1) of first order. Now we use this property of the candidates in a recursive procedure to split up approximate right first-order factors of L. We choose *adaption criterions* and apply them for minimizing rest terms which appear in the right remainders $\mathrm{RRem}(L, \tilde{l})$.

Algorithm 7 Adaption*(L,R)*
(Let L,R be DOs (1.1). The algorithm computes linearly independent approximate solutions $\Phi = \phi_1, \phi_2, \ldots, \phi_q$ for $L\,R$. The fundamental system of R is known.)

```
1. Use structure theorems of computer algebra (e.g., those of chapter 4) for determining candidates $\tilde{y}_j$ for possible approximate exponential solutions of $L$. Each $\tilde{y}_j$ contains free coefficients. Assign $\Phi$ the empty sequence.
2. for each $\tilde{y}_j \in \{\tilde{y}_j\}$
    (a) Compute a DO $\tilde{l}$ (1.1) of first order with free parameters $c_1, \ldots, c_m$ and $\tilde{l}\,\tilde{y}_j = 0$.
    (b) Compute the right quotient $Q$ of the form (1.1) and the right remainder $r$ such that $p\,L = Q\,\tilde{l} + r$ and $p, r \in \bar{\mathbb{K}}[c_1, \ldots, c_m][x]$.
    (c) Determine the coefficients $c_1, \ldots, c_m$ in $\tilde{l}$ and $\tilde{y}_j$ by applying adaption criterions (least square method, smooth adaption, collocation, ...) to $r$.
    (d) Determine a fundamental system of $\tilde{l}\,R$ by d'Alembert reduction 2.13. Denote by $\phi$ the function, which is in the fundamental system of $\tilde{l}\,R$ but not in the fundamental system of $R$.
    (e) if $\phi$ is elementary and $\det(W_{\Phi,\phi}) \neq 0$ then $\Phi := \Phi, \phi$
    (f) $\Psi := \{$Adaption$(Q, \tilde{l}\,R)\}$
    (g) for each $\psi \in \Psi$
            if $\det(W_{\Phi,\psi}) \neq 0$ then $\Phi := \Phi, \psi$
3. return $\Phi$.
```

The algorithm should be started with **Adaption**(L,1), where 1 is the identity DO.

The result of the whole adaption step is a fundamental system of $\tilde{L}$ in form of a sequence of q linearly independent functions $\Phi = \phi_1, \phi_2, \ldots, \phi_q$. Our concept doesn't need the adaptive differential operator $\tilde{L}$ in an explicit form. Nevertheless, $\tilde{L}$ can easily be computed by

$$\tilde{L}\, y = \frac{\det(W_{\phi_1,\phi_2,\ldots,\phi_q,y})}{\det(W_{\phi_1,\phi_2,\ldots,\phi_q})}. \tag{5.1}$$

Note, (5.1) is monic.

The fundamental system of $\tilde{L}$ was computed using d'Alembert reduction. The integration routine (hopefully an implementation of the entire Risch algorithm [54, 29]) decides whether the integrals are elementary functions or not.

The search for possible approximate exponential solutions in step 1 can be restricted to subclasses of exponential functions as polynomials, finite Laurent series solutions or rational functions. This makes the computation less complex, but we lose the ability of describing all kinds of singularities.

In step 2c a question arises: Why minimize the right remainder r and not the defect function $L\ \tilde{y}$ (as usually done)? The answer is: From $p\,L\ \tilde{y} = Q\,\tilde{l}\ \tilde{y} + r\tilde{y} = r\tilde{y}$ we know that the defect function is the remainder r multiplied by $\tilde{y}/p$. Thus, minimizing the defect means minimizing $\tilde{y}$, too. This is not our goal. Additionally, this adaption algorithm has the great advantage that the adaption criterions should be applied to polynomials - one of the most simple algebraic structures. Our adaption algorithm is different to those of [66, 47, 34] for this reason and for another reason, too.

Unlike them, in the adaption step we don't use boundary conditions nor start solutions. (Only the question, whether we have a BVP or an IVP becomes important for choosing the criterions.) Thereby we decouple completely adaption from approximation and simplify the reuse of once found adaptive solutions with new approximation criterions.

We have to determine the coefficients $c_1, \ldots, c_m$ in r in such a manner that r becomes minimal in the sense of the adaption criterions. E.g., if a boundary value problem is given, then we are interested in a good approximation (and a minimal r) in the whole segment; if an initial value problem is given, then we are mostly interested in a good approximation near the initial point. We now indicate possible adaption criterions, cf. [34].

5.1 Least square method

5.1.1 Least square method with integral norms

The least square method is suitable for boundary value problems in $[x_0, x_1]$. In the adaption step, the right remainder $r \in \bar{\mathbb{K}}[c_1, \ldots, c_m][x]$ is contained in $C_2[x_0, x_1]$. Therefore, we can apply the scalar product (2.8). We have to minimize $\|r(x)\|$, which is the same as minimizing

$$F(c_1, \ldots, c_m) = \|r(x)\|^2 = \int_{x_0}^{x_1} |r(x)|^2\, dx.$$

One necessary condition for the minimum is

$$\begin{array}{rcl} F_{c_1}(c_1, \ldots, c_m) & = & 0 \\ F_{c_2}(c_1, \ldots, c_m) & = & 0 \\ & \vdots & \\ F_{c_m}(c_1, \ldots, c_m) & = & 0 \end{array} \tag{5.2}$$

We get a system of m non-linear polynomial equations in m unkowns, which we can solve

- symbolically, if m is small (in practice: less than 4),
- numerically.

The mimimal r can be computed by the coefficients of the minimal $F(\hat{c}_1, \ldots, \hat{c}_m)$, where $\hat{c}_1$, …, $\hat{c}_m$ is one solution of the system (5.2). Although in general we get a system of non-linear equations, the algorithms in chapter 4 for finding solution candidates in subclasses of $\mathcal{ELF}$ produce candidates which cause linear systems (5.2). This is one argument in favor for the whole concept of the thesis.

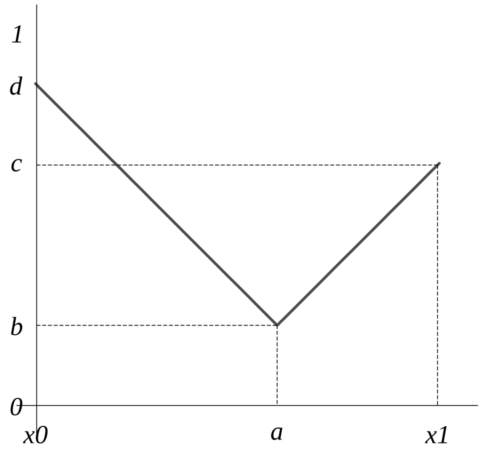
The scalar product (2.8) is suitable for those boundary value problems, where our interest in accuracy is the same at each point in $[x_0, x_1]$. If our interest varies on the segment, we adapt the method.

The space of complex polynomials r is a subspace of the space of functions $f : [x_0, x_1] \to \mathbb{C}$, for which holds $\int_{x_0}^{x_1} |f(x)|^2 \, dx < \infty$. The space of complex polynomials r forms a complex Euclidean space with the scalar product

$$\begin{aligned} \langle f(x), g(x) \rangle \ &:= \ \int_{x_0}^{x_1} f(x)\overline{g(x)}\, w(x) \ dx, \\ & \quad w \in \hat{C}^{(0)}[x_0, x_1],\ 0 \le w(x) \le 1,\ w(x) \not\equiv 0, \end{aligned} \tag{5.3}$$

i.e., the scalar product (2.8) is generalized by using some weight function w, which expresses our interest in accuracy: $w(\xi) = 0$ means that we have no interest at point ξ, $w(\xi) = 1$ means that we have maximal interest at point ξ. The weight function of (2.8) is $w = 1$, other practicable weight functions are, e.g.,

$$w(x) = \begin{cases} \dfrac{(x - x_0)b + (a - x)d}{a - x_0} & : \ x_0 \le x \le a \\ \dfrac{(x_1 - x)b + (x - a)c}{x_1 - a} & : \ a < x \le x_1 \end{cases}$$



or polynomials $w \in \mathbb{K}[x]$.

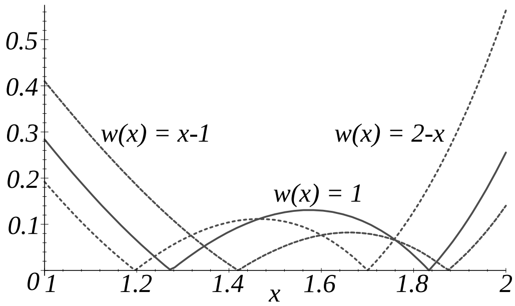
Example. Consider the DE

$$L \ y(x) = 3\,x^2\, D^2\, y(x) + 2\,x\,(x^2 + 5)\, D\, y(x) + (2\,x + 2)\, y(x) \ = \ 0$$

taken from [66]. We try to find an approximate solution in $[1, 2]$ in form of a truncated Taylor series $\tilde{y} = 1 + c_1 x + c_2 x^2$. We construct $\tilde{l} = (1 + c_1 x + c_2 x^2)\, D - (c_1 + 2 c_2 x)$ and define $p = 1 + c_1 x + c_2 x^2$. Then, the right quotient Q and the right remainder r for $p\, L = Q\, \tilde{l} + r$ are

$$\begin{aligned} Q &= 3\,x^2\, D + 2\,x\,(x^2 + 5), \\ r &= 4\,c_2 x^4 + (2\,c_1 + 2\,c_2) x^3 + (2\,c_1 + 28\,c_2) x^2 + (12\,c_1 + 2) x + 2. \end{aligned}$$

The figure shows the absolute value of the minimal r for three different weight functions.



5.1.2 Least square method for sequences

This adaption criterion is suitable for initial value problems at some point ξ, but can also be used for boundary value problems if one of the boundaries is “near” to some singular point of the considered DE. Each right remainder $r \in \bar{\mathbb{K}}[c_1, \ldots, c_m][x]$ can be represented as a Taylor series $r = \sum_{n=0}^{\infty} a_n\,(x-\xi)^n$ at the point ξ. Therefore, we can apply the scalar product (2.9) to the sequence of series coefficients $a_0, a_1, \ldots, a_{n-1}, a_n, a_{n+1}, \ldots$. We have to minimize

$$F(c_1, \ldots, c_m) = \|a_n\|^2 = \sum_{n=0}^{\deg(r,x)} |a_n|^2.$$

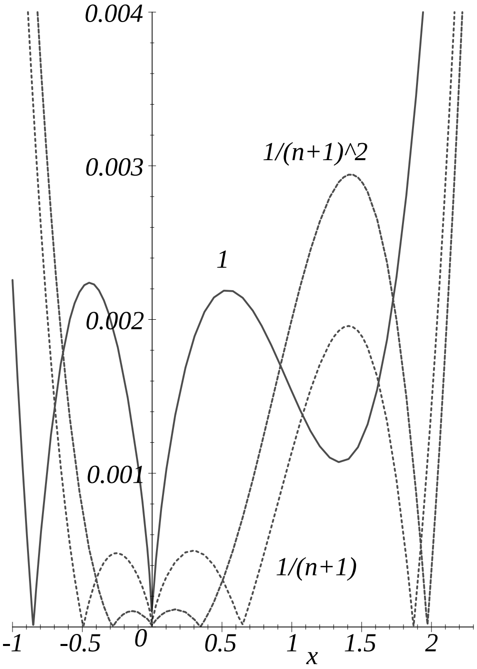
As in the previous section, we use the necessary condition for the minimum und get the system (5.2). Again, the algorithms in chapter 4 for finding solution candidates in subclasses of $\mathcal{ELF}$ produce candidates which cause linear systems (5.2).

The scalar product (2.9) can also be generalized using some sequence $W \in l_2$,

$$\begin{aligned} \langle a_n, b_n \rangle \quad &:= \quad \sum_{n=0}^{\infty} a_n \overline{b_n}\, w_n, \\ & \qquad w_n \in [0,1],\ w_n \not\equiv 0, \end{aligned} \tag{5.4}$$

The weight w_n expresses the kind of interest in accuracy. An increasing weight $w_n < w_{n+1}$ makes the higher coefficients more important, a decreasing weight has the opposite effect.
Continued example. Consider the DE $3\,x^2\,D^2\,y(x)+2\,x\,(x^2+5)\,D\,y(x)+(2\,x+2)\,y(x) = 0$. We find an approximate solution in form of a truncated Laurent series $\tilde{y} = x^{-1/3}(1 + c_1 x + c_2 x^2)$ in point 0.

The figure shows the absolute value of the difference between the exact solution (with the same index $-1/3$) and $\tilde{y}$ for three different weights w_n.



5.1.3 Least square method applied to collocation

Let be given a sequence of points $\Xi = \xi_1, \xi_2, \dots, \xi_m$ with $\forall j = 1(1)m : \ (x_0 \le \xi_j) \wedge (\xi_j \le x_1)$ where m is the number of coefficients in $r(x)$. Let $i(j)$ be a natural-valued function of the indices $j = 1(1)m$ with the property $\forall \xi_j, \xi_k \in \Xi : \ (\xi_j = \xi_k) \to (i(j) \ne i(k))$. For each point $\xi_j \in \Xi$ we demand that the derivative $D^{i(j)}\ r(x)$ should be near 0 at the point ξ_j. We use the scalar product (5.4) and change the sense of the sequence whose norm has to be minimized: If $\xi \in \Xi$ and

$$r(x) = \sum_{n=0}^{\infty} t_n (x - \xi)^n \tag{5.5}$$

and we want to minimize $D^i\ r(x)$ at the point ξ, then we append $i!\, t_i$ to the sequence a_n (with the weight w_n). In our case of coefficients $\hat{c}_1, \dots, \hat{c}_m$ coming from candidates in $\mathcal{ELF}$ we get a linear system with m equations for m unknowns.

Continued example. We find an approximate solution in form of a truncated Laurent series $\tilde{y} = x^{-1/3}(1 + c_1 x + c_2 x^2)$, which fulfils $r(0) \approx r(1) \approx 0$. The right remander r is

$$5x^3 c_2 + (3c_2 + 2c_1)x^2 + (33c_2 + 3c_1 - 1)x + 3 + 12c_1.$$

We define $a_1 = 3 + 12c_1$ as the coefficient at power 0 of the Taylor series in point 0, $a_2 = 41c_2 + 17c_1 + 2$ as the coefficient at power 0 of the Taylor series in point 1. If $w_n = 1$, then $\|a_n\|^2 = {a_1}^2 + {a_2}^2$. We get a linear system for the minimum which gives the same solution as in the next section.

5.2 Collocation

This is the usual kind of collocation, where we demand that $r(x) = 0$ at a sequence of points ξ_n, $n = 1(1)m$, $x_0 \le \xi_1 < \xi_2 < \dots < \xi_m \le x_1$. We call it *pure*, because the criterion doesn't deal with the derivatives of r. Like in the previous section, we can expand this method and use derivatives of r.

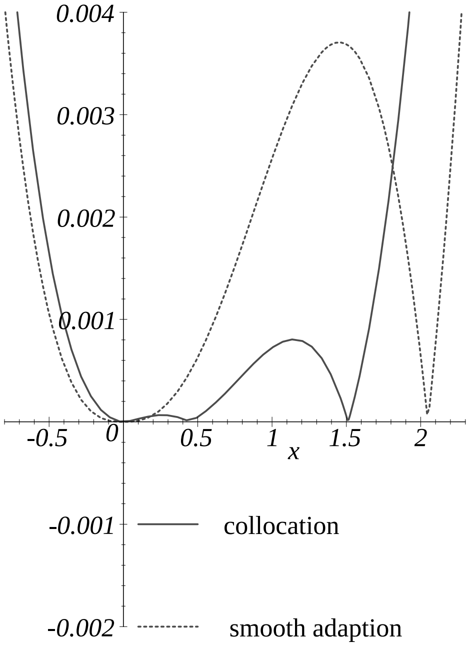
Continued example. We find an approximate solution in form of a truncated Laurent series $\tilde{y} = x^{-1/3}(1 + c_1 x + c_2 x^2)$, which fulfils $r(0) = r(1) = 0$.

5.3 Smooth adaption

Smooth or (complete) Taylor adaption is suitable for initial value problems at some (regular) point ξ. We demand $(D^j\ r)(\xi) = 0$ for $j = 1(1)m$, m is the number of coefficients $\hat{c}_1, \ldots, \hat{c}_m$ in r. Again, the algorithms in chapter 4 for finding solution candidates in subclasses of $\mathcal{ELF}$ cause linear systems. From (5.5) we get at system of m equations $t_1 = 0$, $t_2 = 0$, ..., $t_m = 0$ in m unknowns $\hat{c}_1, \ldots, \hat{c}_m$.

This usual smooth adaption method can be generalized and used for problems where one of the boundaries is "near" to some singular point ξ of the considered DE, because r is a polynomial even in a singular point (by choosing an appropriate p in algorithm 7).

Continued example. We find an approximate solution in form of a truncated Laurent series $\tilde{y} = x^{-1/3}(1 + c_1 x + c_2 x^2)$, which fulfils $r(0) = (D\ r)(0) = 0$. The figure shows the absolute value of the difference between the exact solution and the approximate solutions from collocation and smooth adaption, respectively.



Summarizing, each adaption criterion has its own advantages.

Chapter 6

Approximation step

When the approximation step starts, we have already determined q linearly independent functions $\Phi = \phi_1, \phi_2, \ldots, \phi_q$ with $\tilde{L}\ \phi_i = 0$, where $\tilde{L}$ is an adaptive differential operator in $\mathcal{ELF}$. We know that $q \geq ord$, because if $\tilde{L} \in \mathcal{ELF}$, then we are able to approximate all *ord* formal solutions of the fundamental system of the given DO L in theorem 2.1. We have now to combine the "best" functions from Φ (in the sense of approximation criterions) to get our approximate solution $\tilde{y}$.

The usually proposed procedure [35, 34] for the approximation consists of two steps: First, we make a linear ansatz $\tilde{y} = \sum_{i=1}^{q} c_i \phi_i$ and substitute $\tilde{y}$ into the constraints. From the constraints we find *ord* of the q coefficients. Then, the other $q - ord$ coefficients are determined by applying approximation criterions to the whole $\tilde{y}$. In most cases, none of the coefficients will be 0. This method works fine as long as q is not much greater than *ord* and each function in Φ has an utmost simple structure. But, our algorithm may give a large number of functions in Φ, and most of them will be quite complex. A linear combination of all functions in Φ with a lot of nonzero coefficients c_i will result in an even more complex expression. Because it would contradict our goal of finding simple and lucid approximate solutions, we will not follow this way.

Each function in Φ we got from our adaption step is the result of some optimization process (they minimize norms of rest terms) and, therefore, is an optimal approximation (in the sense of adaption criterions) to some function of the fundamental system of the given differential operator L. Now, we sort the functions in Φ by applying an approximation criterion to each of them. Note, the approximation criterions are similar to the adaption criterions listed in section 5.1. It is even possible, that the same criterions are used in the adaption and the approximation step.

Algorithm 8 Approximation*(L,U,Φ)*

(Let L be a monic homogeneous linear differential operator (1.4) with coefficients $q_i \in \mathbb{K}(x)$ and let U be linear boundary constraints (1.3). Let ord be the order of L. Let $\Phi = \phi_1, \phi_2, \ldots, \phi_q$ be a sequence of linearly independent functions with $q \geq ord$. The algorithm computes a function $\tilde{y}$ which is an approximate solution for L and an exact solution of the boundary constraints.)

```
1. for each $\phi \in \Phi$
    (a) Compute $\tilde{l}$, the linear monic DO of first order and $\tilde{l}\ \phi = 0$.
    (b) Compute the right remainder $r$ such that $L = Q\,\tilde{l} + r$, $r \in \bar{\mathbb{K}}[\ln(x)](x)$.
    (c) Apply approximation criterions (least square methods) to $r$ and compute its
        norm.
```

```
2. Sort $\Phi$ by increasing norms of the corresponding $r$.
3. $i := 2$; $\Psi := \phi_1$;
   while $i \leq q$ and the lenght of $\Psi$ is less than ord do
      (a) if $\Psi, \phi_i$ is not ill-conditioned then $\Psi := \Psi, \phi_i$;
      (b) $i := i + 1$
   if the lenght of $\Psi$ is less than ord then return FAIL else $\Phi := \Psi$
4. Make a linear ansatz $\tilde{y} = \sum_{i=1}^{ord} c_i\, \phi_i$ and substitute it into the constraints $U$. Determine the coefficients from the linear system.
5. return $\tilde{y}$.
```

The set of possible approximation criterions is described in section 5.1. As shown there, for computing the norms of r we use the scalar products (5.3) and (5.4). Now we compute the norms not for minimizing them, but for comparing one with another. Because r may be more complicated ($r \in \bar{\mathbb{K}}[\ln(x)](x)$) as in the adaption step and because we don't need the norms in high precision, it is advisable to compute the norms numerically.

Up to step 4 we never made use of the boundary value conditions, i.e. our approximate fundamental system is reusable for variant boundary value conditions.

Step 3 has been added for practical reasons. By our adaption algorithm, the functions in Φ are linearly independent. Still it may happen that they are numerically very close to be linear dependent. In this case, the coefficients c_i in the linear ansatz at step 4 would behave very unstable, that is, a small change in the boundary conditions may lead to a large change for the c_i. To prevent this shortcoming, we included a condition check for Φ. The next section describes, how this can be done.

6.1 Ill-conditioned problems

Formally, the *condition number* $\operatorname{condn}(M)$ of a matrix M is defined as the absolute value of the ratio of the largest of the eigenvalues to the smallest of the eigenvalues. If a matrix is singular then its condition number is defined to be infinite, and a matrix is *ill-conditioned* if its condition number is "too large", e.g., if its reciprocal approaches the machine's floating-point precision. We define a sequence Y of functions to be ill-conditioned, if the Wronskian W_Y is ill-conditioned.

If Y is a sequence of functions depending on x, then the eigenvalues of the Wronskian W_Y and its condition number are functions of x. In the following sections we are able to compute them as defined, but, as the dimension of the Wronskian grows, this approach is too expensive.

Therefore, our program uses another procedure. Let $Y = y_1(x), y_2(x), \ldots, y_k(x)$ and let $z_1(x), \ldots, z_k(x)$ be its eigenvalues. We know $\det(W_Y) = \prod\limits_{i=1}^{k} z_i(x)$. Let $z_1(x)$ and $z_k(x)$ be the eigenvalues with respectively the smallest and the largest absolute value. Then,

$$\operatorname{condn}(W_Y) = \left| \frac{z_k(x)}{z_1(x)} \right| = \left| \frac{z_k(x)}{\det(W_Y)} \prod_{i=2}^{k} z_i(x) \right| \leq \frac{\mid z_k(x) \mid^k}{\mid \det(W_Y) \mid},$$

or, in other words, if the determinant is not too small and the biggest absolute value of the eigenvalues is not too big, then the matrix is not ill-conditioned.

The determinant of the Wronskian can much easier be computed than its smallest eigenvalue. If we store the determinant of the Wronskian, it can also be reused in chapter 7, where we need it for the Green function (in form of (2.14)) of the adaptive differential operator. Regarding (5.1), we mention that the singularities of the differential operator are given by the roots of $\det(W_Y)$. Thus, controlling $\det(W_Y) > 0$ has also the welcome side-effect of proving non-singularity of the corresponding adaptive differential operator.

A complete linear space with a norm is called *Banach space*. From the theory of linear operators [42, theorem 7, §5, chapter 4] we know that the absolute value of the eigenvalues of the bounded linear operator M in some Banach space is bounded by the operator norm. In the general case, W_Y cannot be bounded (e.g., if $Y = 1/x^2$, then $\lim_{x\to 0} z_k(x) = \infty$). But, for each regular point ξ of the adaptive differential operator constructed from W_Y by formula (5.1), the functions in Y are analytical functions and the linear operator $b = W_Y(\xi)\, a$ for $a, b \in \mathbb{C}^k$ is bounded. Additionally, $\mathbb{C}^k$ with the norms $\|a\| = \sqrt{\sum_{i=1(1)k} |a_i|^2}$, $\|a\| = \sum_{i=1(1)k} |a_i|$ or $\|a\| = \max_{i=1(1)k} |a_i|$ are Banach spaces. Therefore, we can apply the theorem.

The operator norm of $W_Y(\xi)$ can be computed by $\|W_Y(\xi)\| = \sup_{\|a\|\le 1} \|W_Y(\xi)\, a\|$, cf. [42, theorem 1, §5, chapter 4]. We know

$$W_Y(\xi)\, a = \begin{pmatrix} \sum_{j=1(1)k} W_Y(\xi)_{1,j}\ a_j \\ \sum_{j=1(1)k} W_Y(\xi)_{2,j}\ a_j \\ \vdots \\ \sum_{j=1(1)k} W_Y(\xi)_{k,j}\ a_j \end{pmatrix}. \tag{6.1}$$

By applying the norm $\|a\| = \sqrt{\sum_{i=1(1)k} |a_i|^2}$ to (6.1) we get

$$|\ z_k(\xi)\ | \le \sup_{\|a\|\le 1} \|W_Y(\xi)\, a\| \ = \ \sup_{\|a\|\le 1} \sqrt{\sum_{i=1(1)k} \left| \sum_{j=1(1)k} W_Y(\xi)_{i,j}\ a_j \right|^2}$$

By applying the inequality of Schwarz we further obtain

$$|\ z_k(\xi)\ | \le \sup_{\|a\|\le 1} \sqrt{\sum_{i=1(1)k} \left(\sum_{j=1(1)k} |W_Y(\xi)_{i,j}|^2 \right) \left(\sum_{j=1(1)k} |a_j|^2 \right)}$$

Thus, in the regular case the biggest absolute value can be bounded by

$$|\ z_k(\xi)\ | \le \sqrt{\sum_{i=1(1)k} \sum_{j=1(1)k} |\ W_Y(\xi)_{i,j}\ |^2}.$$

By applying the norm $\|a\| = \sqrt{\sum_{i=1(1)k} |a_i|}$ to (6.1) we get also the bound

$$|\ z_k(\xi)\ | \le \sum_{i=1(1)k} \sqrt{\sum_{j=1(1)k} |\ W_Y(\xi)_{i,j}\ |^2},$$

using the norm $\|a\| = \max_{i=1(1)k} |a_i|$ we get the bound

$$|\ z_k(\xi)\ | \le \max_{i=1(1)k} \sum_{j=1(1)k} |\ W_Y(\xi)_{i,j}\ |\ .$$

All these bounds are well suited for symbolic calculations.

The biggest absolute value of the eigenvalues can also be approximated by numerical standard methods. We only mention the simple iteration method with

$$| z_k | = \lim_{n\to\infty} \|{W_Y}^n\|^{1/n}.$$

Here we use pointwise computation for avoiding difficult powers of symbolic matrices.

6.1.1 Ill-conditioned sequences of functions and boundary value problems

If we have a boundary problem on $[x_0, x_1]$ and Y is the sequence of candidates for the approximate fundamental system, we should be sure that the condition number $\mathrm{condn}(W_Y)(\xi)$ is "small" for all $\xi \in [x_0, x_1]$.

Example. Consider a BVP in $[\frac{1}{2}, 1]$ and

$$L\, y(x) = 3\,x^2\, D^2\, y(x) + 2\,x\,(x^2+5)\,D\,y(x) + (2\,x+2)\,y(x) = 0.$$

We search for approximate exponential solutions. As described in chapter 4 we compute the polynomial parts $\{\exp(-1/3x^2)\}$, singular parts $\{x^{-1/3}, x^{-2}\}$ and irregular singular parts $\emptyset$. In the adaption step we apply the collocation method and demand $r(0) = r(1/2) = r(1) = 0$. With the candidates

$$\begin{aligned}
\phi_1 &= (1 - 1/4x + 1171/20516x^2 - 25/20516x^3)x^{-1/3}\\
\phi_2 &= (1 + x + x^2)x^{-2}\\
\phi_3 &= (1 - 1/4x + 3/11x^2 + 1/11x^3)x^{-1/3}\exp(-1/3x^2)\\
\phi_4 &= (1 + x - 127/121x^2 + 161/121x^3)x^{-2}\exp(-1/3x^2)
\end{aligned}$$

we go into the approximation step. Now, we choose the integral norm (2.7) in $[\frac{1}{2}, 1]$ for the approximation criterion. As an example, the linear monic DO of first order and $\tilde{l}\ \phi_1 = 0$ is

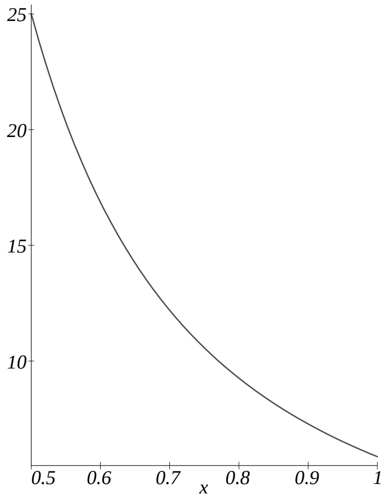
$$\tilde{l} = D + \frac{-10258x + 5855x^2 - 200x^3 - 20516}{3x(-20516 + 5129x - 1171x^2 + 25x^3)},$$

the right remainder is

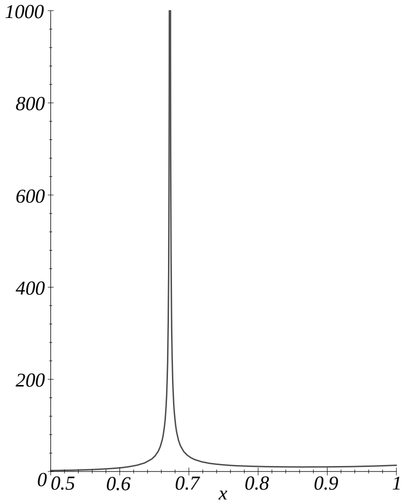
$$r_1 = 40/9\frac{10x^3 - 289x^2 + 416x - 137}{-20516 + 5129x - 1171x^2 + 25x^3}$$

and $\|r_1\| = 0.002207$. In the same way, we compute $\|r_2\| = 0$, $\|r_3\| = 0.06354$ and $\|r_4\| = 0.1252$. The best candidates in the sense of the approximation criterion are ϕ_1, ϕ_2. We verify whether they are ill-conditioned.

The figure shows the condition number of the Wronskian W_{ϕ_1,ϕ_2}:



The ratio of the eigenvalues is bounded by a relatively small number. Thus, our approximate fundamental system for *all* boundary value problems in $[\frac{1}{2}, 1]$ is ϕ_1, ϕ_2. For comparison, this



is the condition number of the Wronskian W_{ϕ_1,ϕ_3} which has a pole in the segment.

6.1.2 Ill-conditioned sequences of functions and initial value problems

If we have an initial value problem in x_1 and Y is the sequence of candidates for the approximate fundamental system, we should be sure that the condition number $\operatorname{condn}(W_Y)(\xi)$ is "small" for all ξ in the "neighbourhood" of x_1. On the other hand, we should expect an unlimited growth of the condition number "near" the next singular point of the given DO on the real line, as the functions in Y typically will have singular parts.
Example. Let be given the initial value problem

$$L\,y(x) = x^2\,D^3\,y(x) + x(-3\,x+1)\,D^2\,y(x) + 2\,(x^2-x-2)\,D\,y(x) + (-2\,x^2+x+4)\,y(x) = 0$$

and $y(1) = 1, y'(1) = 0, y''(1) = \frac{3}{2}$. The exact solution

$$y(x) \;=\; \left(1 + \int_1^x \left[\frac{4\,Y_1(1) - Y_0(1)}{Y_1(1)J_0(1) - \,Y_0(1)J_1(1)}\frac{J_1(u)}{u} + \frac{Y_0(1) - 4\,Y_1(1)}{2\,Y_1(1)J_0(1) - 2\,Y_0(1)J_1(1)}J_0(u) + \right.\right.$$

$$\left.\left.\frac{J_0(1)-4\,J_1(1)}{Y_1(1)J_0(1)-Y_0(1)J_1(1)}\frac{Y_1(u)}{u}+\frac{4\,J_1(1)-J_0(1)}{2\,Y_1(1)J_0(1)-2\,Y_0(1)J_1(1)}Y_0(u)\right]du\right)e^{x-1}$$

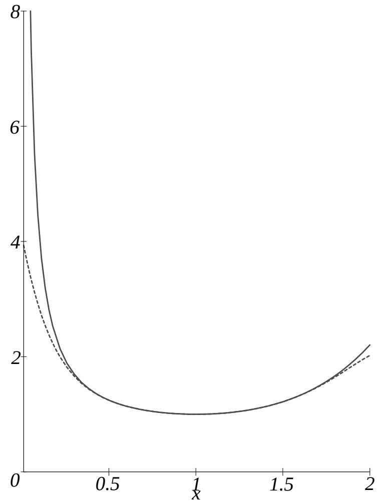
is even after a numerical evaluation of the constants

$$\begin{aligned} y(x) \quad &\approx \quad e^{x-1}\Big(1+5.0471386\int_1^x \frac{J_1(u)}{u}du-2.5235693\int_1^x J_0(u)du+ \\ &\qquad 1.5629496\int_1^x \frac{Y_1(u)}{u}du-0.78147483\int_1^x Y_0(u)du\Big)\end{aligned}$$

quite complicated. The numerical evaluation of the integrals by quadratures makes problems, too. Of course, one can always construct an approximate solution in form of a truncated Taylor series

$$\begin{aligned} y_{series}(x) \quad &= \quad 1+\frac{3}{4}(x-1)^2+\frac{9}{16}(x-1)^4-\frac{73}{240}(x-1)^5+\frac{17}{45}(x-1)^6- \\ &\quad \frac{3679}{10080}(x-1)^7+\frac{4919}{13440}(x-1)^8-\frac{33211}{90720}(x-1)^9,\end{aligned}$$

but the figure



shows, the series solution is bad near the singular point 0. Therefore, we apply the adaptive approximation method. First, we search for an adaptive differential operator $\tilde{L} \in \mathcal{ELF}$ and start up the **Adaption** algorithm 7. We compute the polynomial parts $\{\exp(x)\}$, singular parts $\{x^{-1}\}$ and irregular singular parts $\emptyset$ of L. Our adaption criterion is smooth adaption in point 0. The adaption algorithm computes the candidates recursively and among the candidates in the highest level of the tree are

$$\begin{aligned} \phi_1 \quad &= \quad \frac{1+3/4x-5/24x^3}{x} \\ \phi_2 \quad &= \quad 1+x+1/2x^2+4/27x^3+5/216x^4 \\ \phi_3 \quad &= \quad (1+x+9/20x^2+7/60x^3+239/13440x^4)x^3 \\ \phi_4 \quad &= \quad \frac{1-1/4x^2}{x}\exp(x) \\ \phi_5 \quad &= \quad \exp(x) \\ \phi_6 \quad &= \quad (1/896x^4-1/20x^2+1)x^3\exp(x).\end{aligned}$$

With the candidates of the highest level we recursively compute other candidates, e.g., $\exp(x)$ allows a factorization of $L=[x^2\,D^2+(x-2x^2)D+2x^2-x-4][D-1]$. From this factorization we get another candidate

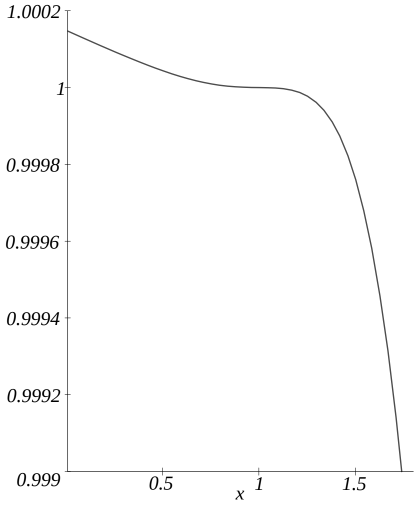
$$\phi_7 = \left(\frac{1}{1800}x^5 - \frac{1}{4}x + x^{-1} + \left(\frac{1}{43008}x^7 - \frac{1}{960}x^5 + \frac{1}{48}x^3\right)\ln(x)\right)\exp(x).$$

Other candidates are non-elementary, too complicated or to bad in the sense of the approximation criterion, for which we choose the integral norm (2.7) in $[\frac{1}{2}, 1]$. The best three candidates are ϕ_5, ϕ_6 and ϕ_7. The condition number of $W_{\phi_5,\phi_6,\phi_7}(1)$ is less than 15, so our fundamental system is stable near 1. On the other hand, as should be expected, $\lim_{x\to 0+}\operatorname{condn}(W\phi_5, \phi_6, \phi_7)(x) = \infty$, e.g., $\operatorname{condn}(W_{\phi_5,\phi_6,\phi_7})(0.1) \approx 39000$. For comparison, the condition number for $W_{\phi_3,\phi_6}(1)$ is about 22000.

We now substitute our fundamental system in the initial conditions and get our approximate solution

$$\begin{aligned}\tilde{y} \;=\; & -\frac{1}{37460980963350}\Big(x^4\left(-866560275x^8 + 38821900320x^6 - 776438006400\right)\ln(x) \\ & +43456525x^{10} - 3352360500x^8 + 129480736896x^6 - 3003715008000x^4 \\ & +9317256076800x^2 - 6631669557871x - 37269024307200\Big)/x\exp(x-1)\end{aligned}$$

The next figure shows the ratio of the exact solution and $\tilde{y}$.



Even near the singularity the ratio is less than 1.0002, i.e., $\tilde{y}$ mimics perfectly the behaviour of the exact solution.

Chapter 7

Error estimation

Approximate solutions are of no practical use without proper error estimates. These are often obtained by estimating rest terms in series (which is sometimes impossible, especially when the convergence radius is unknown).

Lehmann [44] describes methods using *Green functions*. Thinh [66] adapts these methods to singular initial value problems. We enriched their concept by some details to make it implementable in a computer algebra system.

After the construction of an approximate solution $\tilde{y}$ we estimate the error function

$$\text{error} = |y - \tilde{y}| = \sqrt{(y-\tilde{y})\overline{(y-\tilde{y})}} = \sqrt{\Re(y-\tilde{y})^2 + \Im(y-\tilde{y})^2}.$$

We work in the *complex* Euclidean space, because even for the simple case $\mathbb{K} = \mathbb{Q}$ we get complex solution functions (with coefficients $\in \bar{\mathbb{Q}}$).

7.1 Error estimation for boundary value problems

This section contains results in analogy to [44, page 270].

Let $\tilde{y} \in \hat{C}^{(ord)}[x_0, x_1]$ be an approximate solution of the some linear ODE (2.10) and an exact solution of linear boundary constraints (1.3). We are interested in the behaviour of the error function $|y(x) - \tilde{y}(x)|$ in $[x_0, x_1]$, where $y(x)$ is the exact solution of (2.10,1.3).

Theorem 7.1 *Let $\tilde{L}$ be a homogeneous linear differential operator such that $\Delta := L - \tilde{L}$ is of an order less than L. Let L and $\tilde{L}$ be regular in $[x_0, x_1]$ and let $\tilde{\Gamma}(x, \xi)$ be the Green function of $\tilde{L}$ and (2.11). If*

$$h = \sqrt{\int_{x_0}^{x_1} \int_{x_0}^{x_1} |\Delta \tilde{\Gamma}(x,\xi)|^2 \, dx \, d\xi} < 1, \tag{7.1}$$

(i.e. we demand the approximation should be reasonably good), then

$$\forall x \in [a,b]: \; |y(x) - \tilde{y}(x)| \leq \frac{\|\tilde{\Gamma}(x,\cdot)\|}{1-h} \, \|L\, \tilde{y}(x) - f(x)\|. \tag{7.2}$$

Proof. On the one hand, $L\,[y(x) - \tilde{y}(x)] = f(x) - L\,\tilde{y}(x)$. Since we demanded L to be regular in $[x_0, x_1]$, $y - \tilde{y} \in \hat{C}^{(ord)}[x_0, x_1]$. Since both $y(x)$ and $\tilde{y}(x)$ fulfil (1.3), $y(x) - \tilde{y}(x)$ fulfils (2.11). With corollary 2.10 we get

$$y(x) - \tilde{y}(x) = \int_{x_0}^{x_1} \Gamma(x,\xi) \left(f(\xi) - L_{(\xi)}\, \tilde{y}(\xi) \right) \, d\xi.$$

Applying the inequality of Schwarz we get

$$|y(x) - \tilde{y}(x)| \leq \|\Gamma(x,\cdot)\| \; \|L\,\tilde{y}(x) - f(x)\|. \tag{7.3}$$

On the other hand, let $g \in \hat{C}^{(0)}[x_0, x_1]$ be an arbitrary function. Because of theorem 2.9,

$$u(x) := \int_{x_0}^{x_1} \tilde{\Gamma}(x,\xi) g(\xi)\, d\xi \tag{7.4}$$

is the unique solution of $\tilde{L}\,u = g$ and (2.11). Because of $L\,u(x) = \Delta\,u(x) + g(x)$ and corollary 2.10 we conclude

$$\begin{aligned} u(x) &= \int_{x_0}^{x_1} \Gamma(x,t) \left[\Delta_{(t)}\,u(t) + g(t)\right] dt \\ &\overset{(7.4)}{=} \int_{x_0}^{x_1} \Gamma(x,t) \left[\Delta_{(t)} \int_{x_0}^{x_1} \tilde{\Gamma}(t,\xi) g(\xi)\, d\xi + g(t)\right] dt \\ &= \int_{x_0}^{x_1} \Gamma(x,t)\, \Delta_{(t)} \int_{x_0}^{x_1} \tilde{\Gamma}(t,\xi) g(\xi)\, d\xi\, dt + \int_{x_0}^{x_1} \Gamma(x,\xi) g(\xi)\, d\xi \end{aligned}$$

Here we can interchange $n-1$ times differentiation of Δ with the integral sign and obtain

$$u(x) = \int_{x_0}^{x_1} \left[\int_{x_0}^{x_1} \Gamma(x,t)\, \Delta_{(t)}\tilde{\Gamma}(t,\xi)\, dt\right] g(\xi)\, d\xi + \int_{x_0}^{x_1} \Gamma(x,\xi) g(\xi)\, d\xi \tag{7.5}$$

Since $g(x)$ was arbitrarily chosen and regarding (7.4,7.5) we conclude

$$\Gamma(x,\xi) - \tilde{\Gamma}(x,\xi) = -\int_{x_0}^{x_1} \Gamma(x,t)\, \Delta_{(t)}\tilde{\Gamma}(t,\xi)\, dt. \tag{7.6}$$

Therefore,

$$\begin{aligned} &\|\Gamma(x,\cdot)\|^2 = \langle \Gamma(x,\cdot), \Gamma(x,\cdot)\rangle \\ &\quad= \langle \Gamma(x,\cdot) - \tilde{\Gamma}(x,\cdot), \Gamma(x,\cdot)\rangle + \langle \tilde{\Gamma}(x,\cdot), \Gamma(x,\cdot)\rangle \\ &\quad\leq |\langle \Gamma(x,\cdot) - \tilde{\Gamma}(x,\cdot), \Gamma(x,\cdot)\rangle| + |\langle \tilde{\Gamma}(x,\cdot), \Gamma(x,\cdot)\rangle| \\ &\quad\overset{(7.6)}{=} \left|\left\langle \int_{x_0}^{x_1} \Gamma(x,t)\, \Delta_{(t)}\tilde{\Gamma}(t,\cdot) dt, \Gamma(x,\cdot)\right\rangle\right| + |\langle \tilde{\Gamma}(x,\cdot), \Gamma(x,\cdot)\rangle| \end{aligned}$$

Applying the inequality of Schwarz twice we further obtain

$$\begin{aligned} \|\Gamma(x,\cdot)\|^2 &\leq \left\|\int_{x_0}^{x_1} \Gamma(x,t)\, \Delta_{(t)}\tilde{\Gamma}(t,\cdot)\, dt\right\| \|\Gamma(x,\cdot)\| + \|\tilde{\Gamma}(x,\cdot)\|\; \|\Gamma(x,\cdot)\| \\ &\leq h\; \|\Gamma(x,\cdot)\|^2 + \|\tilde{\Gamma}(x,\cdot)\|\; \|\Gamma(x,\cdot)\|, \end{aligned}$$

where h is defined as in (7.1). If $1-h > 0$, then we may deduce

$$\|\Gamma(x,\cdot)\| \leq \frac{\|\tilde{\Gamma}(x,\cdot)\|}{1-h}.$$

Together with (7.3), this completes the proof. □

Note that $\tilde{L}$ and $\tilde{\Gamma}$ do not depend on $\tilde{y}$. Nevertheless, from the computational point of view the condition $\tilde{L}\,\tilde{y} = f$ is advantageous for at least three reasons.

First, in this case we are able to build up $\tilde{\Gamma}$ using the information about $\tilde{y}$. We have already computed the determinant of the Wronskian for the Green function in the approximation step. This result can now be reused.

Second, in section 6.1 we proved by construction $\tilde{L}$ to be regular in $[x_0, x_1]$. (By our algorithm, the determinant of the Wronskian of the fundamental system of $\tilde{L}$ should be nonzero for all $\xi \in [x_0, x_1]$.) The regularity of $\tilde{L}$, a necessary condition for the error estimation, is not a matter of course. Consider the example in section 2.4.2. Although the solution space of $(x-1)\,D^2 - x\,D + 1$ is spanned by "harmless" functions $y_1(x) = x$, $y_2(x) = \exp(x)$, the differential operator is singular in point 1. Supposing y_1, y_2 are approximate solutions, the corresponding adaptive DO is not applicable to boundary value problems in segments including point 1.

Third, if we demand that the approximate solution should be reasonably good and construct $\tilde{\Gamma}$ from $\tilde{y}$, practice shows that we may expect $h < 1$.

7.2 Error estimation for initial value problems

This section contains results in analogy of [66, section 6.1], where Thinh modified the proof in [44].

Let $\tilde{y} \in \hat{C}^{(ord)}(x_0, x_1]$ be an approximate solution of (2.10) and an exact solution of some initial value conditions

$$y(x_1) = \gamma_0, y'(x_1) = \gamma_1, \ldots, y^{(ord-1)}(x_1) = \gamma_{ord-1}, \tag{7.7}$$

at the point x_1. We are interested in the behaviour of the error function $|y(x) - \tilde{y}(x)|$ in $(x_0, x_1]$, where $y(x)$ is the exact solution of (2.10,7.2).

Theorem 7.2 *Let $\tilde{L}$ be a homogeneous linear differential operator such that $\Delta := L - \tilde{L}$ is of an order less than L. Let L and $\tilde{L}$ be regular in $(x_0, x_1]$ and let $\tilde{\Gamma}_l(x, \xi)$ be the left Green function of $\tilde{L}$. Note that L and $\tilde{L}$ are allowed to be singular in x_0. If*

$$\forall x \in (x_0, x_1] : \; h(x) = \sqrt{\int_x^{x_1} \int_x^{\xi} |\Delta_{(t)}\; \tilde{\Gamma}_l(t, \xi)|^2 \; dt \; d\xi} < 1, \tag{7.8}$$

then

$$\forall x \in (x_0, x_1] : \; |y(x) - \tilde{y}(x)| \le \frac{\|\tilde{\Gamma}_l(x, \cdot)\|_x^{x_1}}{1 - h(x)} \; \|L\,\tilde{y} - f\|_x^{x_1}. \tag{7.9}$$

Proof. Let $x_0 < x \le \xi \le x_1$. On the one hand, $L\,[y(x) - \tilde{y}(x)] = f(x) - L\,\tilde{y}(x)$. Because we demanded L to be regular in $(x_0, x_1]$, $y - \tilde{y} \in \hat{C}^{(ord)}(x_0, x_1]$. Since both $y(x)$ and $\tilde{y}(x)$ fulfil (7.2), $y(x) - \tilde{y}(x)$ fulfils (2.22). With theorem 2.12 we get

$$y(x) - \tilde{y}(x) = \int_x^{x_1} \Gamma_l(x, \xi) \left(f(\xi) - L_{(\xi)}\,\tilde{y}(\xi)\right) d\xi.$$

Applying the inequality of Schwarz we get

$$|y(x) - \tilde{y}(x)| \le \|\Gamma_l(x, \cdot)\|_x^{x_1} \; \|L\,\tilde{y}(x)\|_x^{x_1}. \tag{7.10}$$

On the other hand, there exists a c with $x_0 < c \le x \le x_1$ such that with similar considerations as in theorem 7.1,

$$\begin{aligned} \Gamma_l(x, \xi) - \tilde{\Gamma}_l(x, \xi) &= -\int_c^{x_1} \Gamma_l(x, t) \Delta_{(t)} \tilde{\Gamma}_l(t, \xi)\, dt \\ &= -\int_x^{\xi} \Gamma_l(x, t) \Delta_{(t)} \tilde{\Gamma}_l(t, \xi)\, dt \end{aligned} \tag{7.11}$$

holds, where the second equal sign holds because of $\Gamma_l(x,t)=0$ for $x>t$ and $\tilde{\Gamma}_l(t,\xi)=0$ for $t>\xi$, see (2.19).

Therefore,

$$\begin{array}{rcl}
(\|\Gamma_l(x,\cdot)\|_x^{x_1})^2 & = & \langle\Gamma_l(x,\cdot),\Gamma_l(x,\cdot)\rangle_x^{x_1} \\
 & = & \langle\Gamma_l(x,\cdot)-\tilde{\Gamma}_l(x,\cdot),\Gamma_l(x,\cdot)\rangle_x^{x_1}+\langle\tilde{\Gamma}_l(x,\cdot),\Gamma_l(x,\cdot)\rangle_x^{x_1} \\
 & \leq & |\langle\Gamma_l(x,\cdot)-\tilde{\Gamma}_l(x,\cdot),\Gamma_l(x,\cdot)\rangle_x^{x_1}|+|\langle\tilde{\Gamma}(x,\cdot),\Gamma_l(x,\cdot)\rangle_x^{x_1}| \\
 & \overset{(7.11)}{=} & \left|\left\langle\int_x^{\cdot}\Gamma_l(x,t)\,\Delta_{(t)}\tilde{\Gamma}_l(t,\cdot)\;dt,\Gamma(x,\cdot)\right\rangle_x^{x_1}\right|+|\langle\tilde{\Gamma}_l(x,\cdot),\Gamma_l(x,\cdot)\rangle_x^{x_1}|
\end{array}$$

Applying the inequality of Schwarz twice we further obtain

$$\begin{array}{rcl}
\|\Gamma_l(x,\cdot)\|_x^{x_1\,2} & \leq & \left\|\int_x^{\cdot}\Gamma_l(x,t)\,\Delta_{(t)}\tilde{\Gamma}_l(t,\cdot)\;dt\right\|_x^{x_1}\;\|\Gamma_l(x,\cdot)\|_x^{x_1}+\|\tilde{\Gamma}_l(x,\cdot)\|_x^{x_1}\;\|\Gamma_l(x,\cdot)\|_x^{x_1} \\
 & \leq & \sqrt{\int_x^{x_1}\int_x^{\xi}|\Gamma_l(x,t)|^2\;dt\;\int_x^{\xi}|\Delta_{(t)}\tilde{\Gamma}_l(t,\xi)|^2\;dt\;d\xi}\;\|\Gamma_l(x,\cdot)\|_x^{x_1} \\
 & & \qquad\qquad +\|\tilde{\Gamma}_l(x,\cdot)\|_x^{x_1}\;\|\Gamma_l(x,\cdot)\|_x^{x_1} \\
 & & \left\{\text{if } \xi\leq x_1 \text{ then } \int_x^{\xi}|\Gamma_l(x,t)|^2\;dt\leq\int_x^{x_1}|\Gamma_l(x,t)|^2\;dt\right\} \\
 & \leq & \sqrt{\int_x^{x_1}\int_x^{x_1}|\Gamma_l(x,t)|^2\;dt\;\int_x^{\xi}|\Delta_{(t)}\tilde{\Gamma}_l(t,\xi)|^2\;dt\;d\xi}\;\|\Gamma_l(x,\cdot)\|_x^{x_1} \\
 & & \qquad\qquad +\|\tilde{\Gamma}_l(x,\cdot)\|_x^{x_1}\;\|\Gamma_l(x,\cdot)\|_x^{x_1} \\
 & \leq & h(x)\;\;(\|\Gamma_l(x,\cdot)\|_x^{x_1})^2+\|\tilde{\Gamma}_l(x,\cdot)\|_x^{x_1}\;\|\Gamma_l(x,\cdot)\|_x^{x_1},
\end{array}$$

where $h(x)$ is defined as in (7.8). If $1-h(x)>0$, then we may deduce

$$\|\Gamma_l(x,\cdot)\|\leq\frac{\|\tilde{\Gamma}_l(x,\cdot)\|}{1-h(x)}.$$

Together with (7.10), this completes the proof. □

Theorem 7.2 is only a modification of a known estimation from Thinh.

Corollary 7.3 *(Thinh's estimation (6.17) of [66])*
Let be given the same conditions as in theorem 7.2. If

$$h_T(x)=\sqrt{\int_x^{x_1}\int_{\xi}^{x_1}|(L\;\tilde{\Gamma}_l(x,t))(\xi,t)|^2\;dt\;d\xi}<1$$

then

$$\forall x\in(x_0,x_1]:\;|y(x)-\tilde{y}(x)|\leq\frac{\|\tilde{\Gamma}_l(x,\cdot)\|_x^{x_1}}{1-h_T(x)}\;\|L\,\tilde{y}-f\|_x^{x_1}.$$

Proof. We prove that $h_T(x)=h(x)$. Compare the area where the integrands are integrated.
A second proof is:
We denote $F(x,t)=|(L\;\tilde{\Gamma}_l(x,t))|^2$. Then,

$$(h(x)^2)_x=\left[\int_x^{x_1}\int_x^{\xi}F(t,\xi)\;dt\;d\xi\right]_x=-\int_x^{x_1}F(x,\xi)\;d\xi-\int_x^{x}F(t,x)\;dt=-\int_x^{x_1}F(x,\xi)\;d\xi$$

and

$$(h_T(x)^2)_x = \left[\int_x^{x_1}\int_\xi^{x_1} F(\xi,t)\ dt\ d\xi\right]_x = -\int_x^{x_1} F(x,t)\ dt = (h(x)^2)_x.$$

Additionally the functions have the point $h_T(x_1) = h(x_1) = 0$ in common. □

Then, what is the difference between h_T and h? Only the practical use shows it: Often h has an inner integral which can be solved symbolically while the inner integral of h_T cannot, because often the integrand contains singularities which make symbolic indefinite integration worthless. That is why we prefer $h(x)$.

7.3 Example and annotations

Consider the homogeneous DE

$$3\,x^2\,D^2\,y(x) + 2\,x\,(x^2+5)\,D\,y(x) + (2\,x+2)\,y(x) = 0$$

with the initial value conditions $y(1) = 1$, $y'(1) = 0$. The DE has a regular singularity in point 0. For comparison we compute the exact solution

$$y(x) = \frac{x^2+x+1}{3\,x^2} - 3\,\frac{x^2+x+1}{x^2 e^{(-1/3)}}\int_x^1 \frac{e^{(-1/3\,\alpha^2)}\,\alpha^{2/3}}{(\alpha^2+\alpha+1)^2}\,d\alpha.$$

Using the Frobenius method we construct an approximate solution basis

$$\begin{aligned}\tilde{y}_1(x) &= \frac{x^2+x+1}{x^2}\\ \tilde{y}_2(x) &= \frac{-\frac{65}{23562}\,x^4 + \frac{5}{924}\,x^3 + \frac{7}{132}\,x^2 - \frac{1}{4}\,x + 1}{x^{1/3}}\end{aligned}$$

The approximate solution of the initial value problem is

$$\tilde{y}(x) = -\frac{57581}{168960}\,\tilde{y}_1(x) + \frac{3213}{1280}\,\tilde{y}_2(x). \tag{7.12}$$

We want to estimate its error in $(0,1]$.

First of all we assume $\tilde{L}\,\tilde{y} = 0$ and construct $\tilde{\Gamma}_l$ for $x < \xi$ using (2.19). We use another degree of freedom by choosing the coefficient of the highest derivative of $\tilde{L}$ (and L) as $3\,x^2$. Then,

$$\begin{aligned}\tilde{\Gamma}_l(x,\xi) = &\Big((x^2+x+1)(130\xi^4 - 255\xi^3 - 2499\xi^2 + 11781\xi - 47124)\,x^{1/3}\,\xi^2\\ &-(\xi^2+\xi+1)(130x^4 - 255x^3 - 2499x^2 + 11781x - 47124)\,\xi^{1/3}\,x^2\Big)\Big/\\ &\Big(110\,\xi\,x^{7/3}(13\xi^6 - 2\xi^5 - 119\xi^4 + 714\xi^2 - 2142)\Big)\end{aligned}$$

and

$$\begin{aligned}h(x) &= \frac{2}{3}\sqrt{\int_x^1\int_x^\xi \frac{t^{28/3}(13t-15)^2(\xi^2+\xi+1)^2}{\xi^{4/3}(13\xi^6-2\xi^5-119\xi^4+714\xi^2-2142)^2}dt\ d\xi}\\ &= \frac{2}{3}\sqrt{\int_x^1 \frac{(\xi^2+\xi+1)^2}{\xi^{4/3}(13\xi^6-2\xi^5-119\xi^4+714\xi^2-2142)^2}\int_x^\xi t^{28/3}(13t-15)^2 dt\ d\xi}\end{aligned}$$

After computing the inner integral symbolically,

$$h(x) = \frac{2}{3}\sqrt{\int_x^1 \frac{(\xi^2+\xi+1)^2\left(\xi^{31/3}\left(\frac{507}{37}\xi^2 - \frac{585}{17}\xi + \frac{675}{31}\right) - x^{31/3}\left(\frac{507}{37}x^2 - \frac{585}{17}x + \frac{675}{31}\right)\right)}{\xi^{4/3}(13\xi^6 - 2\xi^5 - 119\xi^4 + 714\xi^2 - 2142)^2}\, d\xi},$$

we calculate $h(0) = 0.0004713121...$ numerically by quadratures. Because of $1 > h(0) > h(x)$, theorem 7.2 holds for $0 < x \le 1$. In analogy we could compute

$$h_T(x) = \frac{2}{3}\sqrt{\int_x^1 \xi^{28/3}(13\xi - 15)^2 \int_\xi^1 \frac{1}{t^{4/3}} \frac{(t^2+t+1)^2}{(13t^6 - 2t^5 - 119t^4 + 714t^2 - 2142)^2}\, dt\, d\xi}.$$

Parts of the inner integral of h_T can be evaluated symbolically, but the result is a huge expression. So we resign. (Computing the double integral numerically is slow.)

Then, we construct the defect function

$$L\,\tilde{y}(x) = \frac{1}{256}\, x^{14/3}\,(15 - 13\,x).$$

The norm of the defect function is

$$||L\,\tilde{y}||_x^1 = \sqrt{\frac{20769}{1277886464} - \frac{675}{2031616}\, x^{31/3} + \frac{585}{1114112}\, x^{34/3} - \frac{507}{2424832}\, x^{37/3}}.$$

We also compute $\|L\,\tilde{y}\|_0^1 = 0.004031\ldots$.

There is no hope of representing the norm $\|\tilde{\Gamma}_l(x,\cdot)\|_x^1 = \sqrt{\int_x^1 \tilde{\Gamma}_l(x,\xi)^2\, d\xi}$ symbolically in a satisfactory manner. We are forced to compute it numerically. Usually, numerical integration of continuous functions by quadratures can be done efficiently. In this example, we require only 4 verified decimal digits.

We get our error estimate (7.9). The errors made while computing the bound numerically are not dramatically large, because the numerical part consists only of quadratures.

If the error estimation is time-relevant, then one possibility to reduce computation time (but also reduce accuracy) is in our example

$$\begin{aligned} |y(x) - \tilde{y}(x)| &\le \frac{\|\tilde{\Gamma}_l(x,\cdot)\|_x^1}{1 - h(x)}\, \|L\,\tilde{y}\|_x^1 \le \frac{\|\tilde{\Gamma}_l(x,\cdot)\|_x^1}{1 - h(0)}\, \|L\,\tilde{y}\|_0^1 \\ &< 0.004033\, \|\tilde{\Gamma}_l(x,\cdot)\|_x^1. \end{aligned} \tag{7.13}$$

We want to represent some numerical results in a table:

			DE has leading coefficient		
x	$y(x)$	$\lvert y(x) - \tilde{y}(x)\rvert$	$3\,x^2$		x
			(7.9)	(7.13)	(7.9)
0.9	0.9919	0.00001652	0.00001909	0.00003025	0.00001962
0.5	0.4242	0.001022	0.001030	0.001033	0.001094
0.3	-1.781	0.004328	0.004376	0.004376	0.004450
0.1	-32.60	0.04643	0.04856	0.04856	0.04722
0.01	-3435	4.588	4.855	4.855	4.662
0.0001	$-3.413 \cdot 10^7$	45520	48180	48180	46250
10^{-10}	$-3.413 \cdot 10^{19}$	$4.551 \cdot 10^{16}$	$4.818 \cdot 10^{16}$	$4.818 \cdot 10^{16}$	$4.624 \cdot 10^{16}$
10^{-100}	$-3.413 \cdot 10^{199}$	$4.551 \cdot 10^{196}$	$4.818 \cdot 10^{196}$	$4.818 \cdot 10^{196}$	$4.624 \cdot 10^{196}$

Even close to the singular point the error is only about one promille of the exact solution, and even the less sharp error estimate (7.13) reflects the qualitative behaviour of the actual error.

To become realistic, we want to show, how our approximate solution (7.12) compares to the ouput values of a commercial numerical ODE solver. We tested the Maple V Release 4 solver `dsolve/numeric`. The available standard methods in Maple are

1. `rkf45`, a Fehlberg fourth-fifth order Runge-Kutta method
2. `classical`, a forward Euler method,
3. `dverk78`, seventh-eighth order continuous Runge-Kutta method,
4. `gear`, a Gear single-step extrapolation method,
5. `mgear`, a Gear multi-step method,
6. `lsode`, an implicit Adams method,
7. `taylorseries`.

We used the standard settings, among them `Digits:=10` (in floating point numbers 10 decimal digits are carried). We got the following answers:

x	1.	2.	3.	4.	5.	6.	7.
0.9	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
0.5	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
0.3	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
0.1	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
0.01	$\surd$	W	E	$\surd$	E	$\surd$	E
0.0001	$\surd$	W	E	$\surd$	E	$\surd$	E
10^{-10}	$\surd$	W	E	E	E	$\surd$	E
10^{-100}	W	E	E	E	E	E	E

Here 'W' means a completely wrong answer, the solver gives a result which is less than the fifth part of the correct result. We wrote 'E' when an error occured (division by zero, tolerance is too small, requested error not achieved, excessive amount of work, try other methods). An answer labelled $\surd$ is correct.

Thus, we must recognize that none of the numerical standard methods for solving linear ODEs in Maple is designed to describe the behaviour of the solution near singularities. Neither a higher accuracy in floating point numbers nor a lower error tolerance changes principally this situation.

Computing error bounds with the above techniques means to solve some typical tasks of computer algebra: computing determinants, simplification of expressions and symbolic differential calculus.

On the other hand, strong numerical capabilities are also needed, especially numerical integration, even though symbolic integration is tried wherever it is possible.

Because of our good experience with the Green functions technique in the linear case, we want to remark that the theoretical basis for error estimation in the non-linear case was also established by Lehmann [44].

Chapter 8

Implementation

All algorithms in this thesis were implemented in Maple V Release 4 and are now updated to Maple V Release 5. The implementation is restricted to the case $\mathbb{K} = \mathbb{Q}[\sqrt{-1}]$. We have tested more than 200 examples, among them all linear DEs with polynomial coefficients from the collection of Kamke [38]. The most recent version of the author's Maple package `DETools` can be downloaded from

http://www.informatik.uni-rostock.de/ls_atp/forschung.html.

In the package are included procedures for finding exact and approximate solutions of DEs.

We compared our search for exact solutions with the Maple package *diffop*, written by Mark van Hoeij, who has implemented his and Singer's factorization algorithm and the Maple differential equation solvers *dsolve/diffeq/expsols* (an implementation of Beke's algorithm following [16, 15]), *dsolve/diffeq/kovacic* (an implementation of Kovacic's algorithm). Although our implementation seems to be much more efficient than the standard implementations of *dsolve*, it is not practicable (compared with diffop) if in the given DE are singularities in an algebraic number of degree greater than 2, because computing with algebraic extensions of higher degree *is* slow.

Van Hoeij gives another argument against Beke's method [70, section 3.4]. Suppose the DE (1.1) of order *ord* has n singularities. In each singularity there are at most *ord* different singular or irregular singular parts. The number of singular or irregular singular parts may reach ord^n and $|\mathcal{M}|$ in step 4 of algorithm 6 is only bounded by $(1 + ord)^n$. Although we tried to reduce the number of singular parts and their combinations as much as it was feasible, an experienced mathematician may construct examples for which Beke's method fails horribly. (In Kamke's collection [38] there is no such example.)

But let us remember that our interest in Beke's algorithm came from finding approximate solutions for boundary and initial value problems. Here we are interested only in the singularities next to the boundaries. Approximate solutions which mirror the behaviour of the exact solution in far away singularities are undesired, because they are too complicated. Additionally, they lose their numerical sense at the next branch cut.

We cannot compare our search for approximate solutions with packages from other authors because it seems, there is no other package (except, maybe, the `FORMAC` package from [66]), which computes approximate solutions of "simple" differential equations in closed form. We are grateful to get hints from other authors! Thus, we decided to add a section with examples from our implementation.

Our Maple procedure for the adaptive approximation method is controlled by at least two global parameters. The first one is the predefined global variable `Order` which normally represents the order of series calculations performed by Maple. We use `Order` in a similar way. By setting `Order` to a natural number, we define the number m of coefficients $c_1, \ldots, c_m$

in the algorithm **Adaption**. Typically, `Order` will be less than 5 to avoid too complicated computations and solutions.

We also defined an upper bound for the number q of candidates $\Phi = \phi_1, \phi_2, \ldots, \phi_q$ in the algorithm **Adaption**. In our Maple program, we called this number `MAXSOL`. Thus, our Maple procedure `Adaption` computes at least `MAXSOL` candidates for approximate solutions. Typically, `MAXSOL` will be less than 9, because it defines the maximal number of functions in the tests for linear independence, which are performed by symbolic Wronskian determinant calculations. At this moment, Wronskian determinants with symbolic entries and more than 8 functions are out of our reach. Because the same Wronskian determinants will be used in the error estimation, we also know an upper bound for the practical use of our program: The order of the given DO should surely be less than 9.

Finally, we also added routines to `DETools` for computing automatically the error estimates. We combined symbolic with numerical algorithms and preferred exact computation whenever it was feasible. Even with (heuristical) simplification of expressions, this method turned out to be practicable only for DOs of order less than 7. Indeed, the Green functions and their norms are complicated expressions. Partially, we can blame it on Maple: We sometimes wished that Maple had faster numerical integration routines, symbolic (Risch) integration is quite slow and buggy.

The quality of the error estimates depends on the coefficient of the highest derivative in the given differential operator. By a multiplication of the DO with a function $f : \mathbb{C} \to \mathbb{C}$ the error estimates become more or less precise. Unfortunately, we could not discover any rule for this process. Thus, we decided to give the user one degree of freedom and let him choose f.

8.1 An example with user interaction

In this section, we give an example of the Maple implementation of our adaptive approximation algorithm applied to D-finite functions. Here we focus on a step-by-step computation in a Maple session with the help of our predefined procedures, but accompanied by the heuristical control of an experienced mathematician. We consider the homogeneous DO $L = 60\,(x-2)\,(x^2+2\,x+4)\,(x^3-2) - 60\,x\,(x-2)\,(x^3-2)\,(x^2+2\,x+4)\,D + 30\,x^2\,(x-2)\,(x^2+2\,x+4)\,(x^3-2)\,D^2 - 10\,(x^2+2\,x+4)\,(x-2)\,(x^3-2)^2\,D^3 + 2\,x^2\,(x-2)\,(x^2+2\,x+4)\,(2\,x^6-20\,x^3+5)\,(x^3-2)^2\,D^4 + 9\,x^2\,(x^6-4\,x^3+10)\,D^5 + (-x^9+6\,x^6-30\,x^3-10)\,D^6$, taken from [17]. Let be given an initial value problem $L\,y(x) = 0$ in point 0 with direction to the left where for the linear boundary constraints (1.3) holds: All entries of γ are equal to 1, U_{x_0} is the zero matrix, U_{x_1} is the identity matrix.

Additionally, we demand in $[-\frac{1}{2}, 0]$ the absolute error to be less than 1/250 of the exact solution. This means, we need an upper bound for the error. We recognize that the order of the DO is relatively high. Thus, we must find really simple solutions to realize the error estimation.

First, we verify, whether the DO fits our demands. There is only one real zero of the leading coefficient,

```
> fsolve(-x^9+6*x^6-30*x^3-10);
```

which is near $-0.678\ldots$. Thus, the necessary condition for an error estimation in $[-\frac{1}{2}, 0]$ is fulfilled: L in regular in the segment. Thus, we start our adaptive approximation.

Our class of adaptive differential operators is $\mathcal{ELF}$. Preparing the adaption step, we try to reduce this function space. We have a look at the singularities which hide in $q_6(x) = -x^9 + 6\,x^6 - 30\,x^3 - 10$. q_6 cannot be factored over the rationals

```
>  factor(-x^9+6*x^6-30*x^3-10);
```

$$-x^9 + 6\,x^6 - 30\,x^3 - 10$$

Therefore, we look for the indicial polynomial in q_6. For this purpose we wrote the Maple procedure `IndicialPol`, which uses the implementation of the p-adic recurrence equation (2.6):

```
>  factor(primpart(IndicialPol(L,y(x),n,-x^9+6*x^6-30*x^3-10),n));
```

$$-n\,(n-1)\,(n-2)\,(n-3)\,(n-4)\,(n-5)\,(n-6)$$

The indices are the same as in regular points, i.e., there are no interesting singular parts for the DO (c.f. algorithm **Singular_Parts**, section 4.4.3). Then, we look for polynomial parts:

```
>  PolyParts(L,y(x));
```

$$\{e^{(-8\,x^2+2/5\,x^5)},\, e^{(8\,x^2-2/5\,x^5)}\}$$

Together with 1, we have three polynomial parts.

We now start our adaption step. If we search for extended series solutions (multiplied with polynomial parts) using smooth adaption in point 0, we get three multiplied by the order six equals 18 candidates. In this example, we prefer using smooth adaption instead of least square methods, because usually the coefficients in smooth adaption are not so complicated as for least square methods. We define `Order`, the number m of coefficients $c_1, \ldots, c_m$ in the algorithm **Adaption**. The Maple procedure which implements the search for candidates in **Adaption**, is `ExtendedSeriesSols`:

```
>  Order:=4:
>  cands:=ExtendedSeriesSols(L,y(x),0,"Smooth"):
```

We don't show the candidates, because they would occupy too much space. We now could try to factor the given DO (and our `AdaptiveApproximation` Maple procedure would do that, if `MAXSOLS` would be greater than 18), but this is not advisable, because

- 18 candidates seems to be enough for the moment,
- we expect more complicated candidates from factorizations of L,
- from the analysis of the indicial polynomials we know, that the exact solution does not contain logarithmic parts.

Thus, we go to the approximation step.

In the approximation step we find out, that the candidates which have polynomial parts $\exp(-8\,x^2 + 2/5\,x^5)$ or $\exp(8\,x^2 - 2/5\,x^5)$ are of lower precision than the polynomial solutions. This can be tested by applying least square method for sequences at some point in $[-\frac{1}{2}, 0]$. Our `Approximation` procedure picks the best candidates:

```
>  cands:=Approximation(L,y(x),[cands],-1/3,"Least Square"):
```

$$x,\, x^2,\, 1 - \frac{1}{2}\,x^3,\, -\frac{4}{105}\,x^8 + \frac{16}{105}\,x^7 + x^4,\, -\frac{4}{63}\,x^9 + \frac{25}{168}\,x^8 + x^5,$$
$$\frac{1285}{4032}\,x^7 + \frac{4}{15}\,x^6 + \frac{257}{480}\,x^5 + \frac{6425}{3072}\,x^4 + x^3$$

Now, using the initial conditions, we combine the candidates:

```
>  gamma:=vector(6,seq(1,i=1..6)]):
>  appsolution:=ApproxSolution(map(unapply,[cands],x),0,gamma);
```

$$x \to 1 + x + \frac{1}{2}\,x^2 + \frac{1}{6}\,x^3 + \frac{1}{24}\,x^4 + \frac{1}{120}\,x^5 + \frac{8}{45}\,x^6 + \frac{2}{315}\,x^7 - \frac{1}{2880}\,x^8 + \frac{2511}{11340}\,x^9$$

The error bound (7.9) is a really huge expression. It would fill many pages and its computation takes about ten minutes. Nevertheless, error estimation for DEs of order 6 is possible in simple cases.

```
>  error:=ErrorEstimation(L,y(x),[cands],x..0,gamma):
```

Interestingly, from the above expression the numerical error bound for the absolute error at $-\frac{1}{2}$ can be evaluated in 10 seconds:

```
>  evalf(error(-1/2))
```

$$0.003141001085$$

Note, the error bound is a strictly descending function in $[-1/2, 0]$. Thus, our approximate solution meets our demands concerning the absolute error. From the approximate solution (even from the initial conditions) we know that the exact solution $y(x)$ behaves like the exponential function. Compared with the value of $y(x)$ the absolute error is small in $[-1/2, 0]$, too.

8.2 List of examples

In this section, we show results of our `AdaptiveApproximation` Maple procedure, which does not need any human interaction. This procedure lacks of the heuristical control by an expert, especially of the size of intermediate expressions.

The timings are given for a Pentium Notebook with 233 MHz, 16 KByte internal cache, 512 KByte external cache and 32 MByte RAM. In all examples, we used the same matrices and vectors for the linear boundary constraints (1.3): All entries of γ are equal to 1, in boundary value problems both matrices U_{x_0} and U_{x_1} are identity matrices. For initial value problems, one of the matrices is the zero matrix, the other is the identity matrix.

We don't show the error estimates - they are simply too complicated for getting information from them without numerical evaluation by a computer. The bound for the absolute error is guaranteed by the automatic error estimation, the real absolute error may be less.

$x\,(x+1)\,(x^2-x+1)\,D^2+(x-1)\,(x^2+x+1)\,D-x^2$, [38, 2.355],[15]

`Order`: 4, `MAXSOL`: 6, IVP in 1 with direction to the left
Adaption criterion: collocation in $[1/10, 1]$
Approximation criterion: integral norm in $[1/10, 1]$
Approximate fundamental system: $(x+1)^{1/3}\,(x^2-x+1)^{1/3}$,

$$\left(\frac{-113094050056360000000}{287875020123041676423}\,x^5+\frac{124000126740262504000}{370125025872482155401}\,x^4-\frac{289381504013296672000}{259087518110737508780\!7}\,x^3\right.$$
$$\left.+\frac{145407958486608153536\!0}{863625060369125029269}\,x^2-\frac{112535234568675490240}{2590875181107375087807}\,x-\frac{850529210986863108800}{2590875181107375087807}\right)(x+1)^{1/3}$$

Coefficients: .5893756286, .4163184184
Error: in 1/5 about 1/100 of the exact solution
Time: 116 seconds
Comments: Most of the time was needed for the integrals in the error estimation. The coefficients in form of rational numbers are only needed for the error estimation to avoid numerical instabilities.

$x\,(x+1)\,(x^2-x+1)\,D^2+(x-1)\,(x^2+x+1)\,D-x^2$, [38, 2.355],[15]

`Order`: 3, `MAXSOL`: 6, IVP in 1 with direction to the left
Adaption criterion: least square method in $[1/10, 1]$
Approximation criterion: integral norm in $[1/10, 1]$
Approximate fundamental system: $(x+1)^{1/3}\,(x^2-x+1)^{1/3}$,

$$\left(-\frac{8670575460172883237446400 0}{18230085493894660362824513 7}\,x^3+\frac{97706900535466256642714000}{6076695164631553454274837 9}\,x^2\right.$$

$$\left. -\frac{20093209855187833693825400}{60766951646315534542748379}\,x+\frac{141211308885933535806095540}{182300854938946603628245137}\right)(x+1)^{1/3}$$

Coefficients: .1231604738, .4254096398

Error: in 1/5 less than 1/100 of the exact solution

Time: 62 seconds

Comments: Most of the time was needed for the integrals in the error estimation. The coefficients in form of rational numbers are only needed for the error estimation to avoid numerical instabilities.

$4\,(x^2+x+1)^2\,D^2-3$, [15]

`Order`: 2, `MAXSOL`: 6, BVP in $[0,1]$

Adaption criterion: least square method in $[0,1]$

Approximation criterion: integral norm in $[0,1]$

Approximate fundamental system: $\sqrt{x^2+x+1}$,

$269998/22199731+42750036/22199731\,x+2218860/22199731\,x^2$

Coefficients: .2420556141, .1652175154

Error: in all points in $[0,1]$ less than 6/1000 of the exact solution

Time: 54 seconds

Comments: Most of the time was needed for the integrals in the error estimation.

$4\,(x^2+x+1)^2\,D^2-3$, [15]

`Order`: 3, `MAXSOL`: 6, BVP in $[0,1]$

Adaption criterion: least square method in $[0,1]$

Approximation criterion: integral norm in $[0,1]$

Approximate fundamental system: $\sqrt{x^2+x+1}$,

$\frac{58987412241806}{139653080499491}+\frac{150310517109396}{139653080499491}\,x+\frac{24601861974060}{139653080499491}\,x^2-\frac{5120446324224}{139653080499491}\,x^3$

Coefficients: .0896849665, .3663895759

Error: in all points in $[0,1]$ less than 2/1000 of the exact solution

Time: 584 seconds

Comments: Most of the time was needed for the integrals in the error estimation.

$4\,x\,(4\,x+1)\,D^3-2\,(8\,x^2-2\,x-3)\,D^2-(12\,x+11)\,D+4\,x+5$, [62]

`Order`: 4, `MAXSOL`: 6, BVP in $[1/10,1]$

Adaption criterion: smooth adaption in 0

Approximation criterion: l_2-norm in 1

Approximate fundamental system: $\exp(x)$, $1/362880\,x^{9/2}+1/5040\,x^{7/2}+1/120\,x^{5/2}$ $+1/6\,x^{3/2}+\sqrt{x}$, $1+15119/30239\,x+229/5498\,x^2+251/181434\,x^3+17/725736\,x^4$

Coefficients: .2615437058, 0, 0

Error: The absolute error is 0.

Time: 22 seconds

$x\,(x^2+1)^2\,D^2-(x^2+1)^2\,D+x^3$, [62]

`Order`: 2, `MAXSOL`: 6, BVP in $[0,1]$

Adaption criterion: least square method in $[0,1]$

Approximation criterion: integral norm in $[0,1]$

Approximate fundamental system: $1/2\,\sqrt{x^2+1}\,\ln(x^2+1)$, $\sqrt{x^2+1}$

Coefficients: .8744656611, .2366811976

Error: The absolute error is 0.

Time: 2 seconds

$x\,(x-1)^2\,D^2-2$, [74]

Order: 2, MAXSOL: 6, IVP in 1/2
Adaption criterion: smooth adaption in 0
Approximation criterion: l_2-norm in 1/2
Approximate fundamental system: $(x^2-1-2\,\ln(x)\,x)/(x-1)$, $x/(x-1)$
Coefficients: 3., $-.658883083$
Error: The absolute error is 0.
Time: 1 second

$x\,(x-1)^2\,D^2+(4\,x-1)\,D+2$, [74]

Order: 2, MAXSOL: 6, IVP in 2
Adaption criterion: smooth adaption in 2
Approximation criterion: l_2-norm in 2
Approximate fundamental system: $((6\,x^2-18\,x+12)\,\ln(x-1)+18-15\,x+x^3-3\,x^2)\,(x-1)^{-2}$, $(2-x)/(x-1)$
Coefficients: $-.06250000000$, -2.062500000
Error: The absolute error is 0.
Time: 1 second

$x\,D^2+4\,D-4\,x$, [74]

Order: 2, MAXSOL: 6, BVP in $[1,2]$
Adaption criterion: least square method in $[1,2]$
Approximation criterion: integral norm in $[1,2]$
Approximate fundamental system: $1/4\,(1+2\,x)\exp(-2\,x)\,x^{-3}$, $(-1/2+x)\exp(2\,x)\,x^{-3}$
Coefficients: .2164305919, .07015754817
Error: The absolute error is 0.
Time: 7 seconds

$3\,x^2\,D^2+2\,x\,(x^2+5)\,D+2\,x+2$, [66]

Order: 4, MAXSOL: 6, IVP in 1 with direction to the left
Adaption criterion: smooth adaption in 0
Approximation criterion: l_2-norm in 1
Approximate fundamental system: $(x^2+x+1)\,x^{-2}$, $(1-1/4\,x+7/132\,x^2+5/924\,x^3-65/23562\,x^4)\,x^{-1/3}$
Coefficients: -1.014926610, 5.020312500
Error: near 0 less than 1/1000 of the exact solution
Time: 8 seconds

$x^3\,D^3+296\,x^2\,D^2+x\,(x^2+28908)\,D+98\,x^2+931392$, [66]

Order: 1, MAXSOL: 8, BVP in $[1,2]$
Adaption criterion: smooth adaption in 0
Approximation criterion: l_2-norm in 1
Approximate fundamental system: $\exp(i\,x)\,x^{-99}$, $\exp(-i\,x)x^{-99}$, x^{-98}
Coefficients: $1449.282731+6813.723711\,i$,
$1449.282731-6813.723713\,i$,
$9901.999999-.1725606932\,10^{-6}\,i$
Error: The absolute error is 0.
Time: 42 seconds
Comments: The tests for linear independence in the adaption step with up to 8 functions cause the high time expense.

$x^2\,D^2 + x\,(5-2\,x)\,D + x^2 - 5\,x + 4$, [66]

Order: 2, MAXSOL: 6 , IVP in 1 with direction to the left

Adaption criterion: smooth adaption in 0

Approximation criterion: l_2-norm in 1

Approximate fundamental system: $\exp(x)\,\ln(x)\,x^{-2}$, $\exp(x)\,x^{-2}$

Coefficients: .7357588824, .3678794412

Error: The absolute error is 0.

Time: 1 second

Thesen

Der Gegenstand dieser Arbeit sind homogene lineare gewöhnliche Differentialgleichungen (Dgl.-n) beliebiger Ordnung mit polynomialen Koeffizienten, d.h. Dgl.-n der Form

$$L\ y(x) = \sum_{i=0}^{ord} q_i\, D^i\, y(x) \ = \ 0, \tag{1}$$

$D\,y(x) = \frac{\partial}{\partial x}\,y(x)$, $\mathbb{K}$ sei ein Konstantenkörper der Charakteristik 0, $\forall i = 0(1)ord:\ q_i \in \mathbb{K}[x]$ und $ord \geq 1$. Eine Funktion $y:\ \mathbb{C} \to \mathbb{C}$ heißt D-finit, wenn sie Lösung einer Dgl. der Form (1) ist. Die Klasse der D-finiten Funktionen ist von großer Bedeutung für die Ingenieur- und Naturwissenschaften: Die Mehrzahl der namentlich bekannten analytischen Funktionen ist D-finit.

Jede D-finite Funktion kann eindeutig durch eine Dgl. (1) mit Nebenbedingungen definiert werden. Die Computeralgebra liefert praktikable Algorithmen, die analytische Ausdrücke D-finiter Funktionen (wie z.B. $\sin(x) + 3\,x$) in diese Standarddarstellung transformieren.

Diese Dissertation beschäftigt sich mit der weitaus komplizierteren inversen Aufgabe: Gegeben ist eine D-finite Funktion in Form einer Dgl. (1) mit Rand- oder Anfangswertbedingungen, gesucht ist der analytische Ausdruck einer Funktion, die exakte oder angenäherte Lösung des Problems ist.

Bei der Lösung dieser Aufgabe folgt der Autor den Grundsätzen der von N.J. Lehmann † begründeten Computeranalytik. Dies macht die Besonderheit und den Vorteil der vorgestellten Algorithmen gegenüber rein numerischen oder rein algebraischen Algorithmen aus:

Die ermittelten analytischen Näherungslösungen spiegeln die wesentlichen Eigenschaften der D-finiten Funktionen gut wider. Die Lösungen bleiben einfach und überschaubar, wobei sie angemessenen Genauigkeitsansprüchen genügen. Die Genauigkeit wird durch Fehlerabschätzungen kontrolliert, die, wie die Lösungen selbst, ohne Hilfestellung des Nutzers erstellt werden.

Das Konzept dieser Dissertation folgt der Methode der adaptiven Approximation, die ein bewährter Bestandteil der Computeranalytik ist.

Im ersten Schritt wird der Raum der Funktionen festgelegt, in dem sich die Näherungslösung befinden soll. Hierfür wird die Klasse derjenigen Differentialoperatoren L aus (1) vorgeschlagen, welche ausschließlich elementare Lösungen im Sinne der Differentialalgebra besitzen und sich dabei in Faktoren der Ordnung 1 mit Koeffizienten aus $\bar{\mathbb{K}}(x)$ zerlegen lassen. Diese Klasse besitzt vorteilhafte Eigenschaften: Ihre elementaren Lösungsfunktionen sind geeignet, um das Verhalten der D-finiten Funktionen in allen Arten von Singularitäten zu approximieren; das Hauptsystem jedes Differentialoperators dieser Klasse läßt sich effizient berechnen.

Im zweiten, dem Adaptionsschritt, werden Näherungslösungen für das Hauptsystem des gegebenen Differentialoperators bestimmt. Mit Hilfe von Struktursätzen aus der Computeralgebra konnten wir Algorithmen angeben, die exakte Lösungen aus der Menge der exponentiellen Funktionen (sowie aus einigen ihrer Untermengen, wie den rationalen Funktionen oder endlichen Laurentreihen) ermitteln, falls solche existieren. Eine Funktion $y(x)$ heißt exponentiell über $\mathbb{K}$, wenn $y(x)/D\,y(x) \in \mathbb{K}(x)$. Die Algorithmen zur Suche von exakten Lösungen lassen sich so modifizieren, daß auch Kandidaten für Näherungslösungen angegeben werden können. Jeder dieser Kandidaten ist Lösung einer Dgl. erster Ordnung der Form (1). Die zugehörigen Differentialoperatoren werden in einer rekursiven Prozedur vom gegebenen Differentialoperator als rechte Linearfaktoren abgespalten. Adaptionskriterien werden durch

- die Anwendung der kleinsten Quadratmethode mit Normen komplexwertiger euklidischer Vektorräume, welche für Restterme minimal zu halten sind,
- vollständige Schmiegung oder
- durch Kollokation

realisiert.

Im dritten, dem Approximationsschritt, werden nur diejenigen Funktionen aus den bei der Adaption ermittelten Näherungslösungen übernommen, die den Approximationskriterien am besten genügen. Als Approximationskriterien werden ebenfalls Normen komplexwertiger euklidischer Vektorräume verwendet. Der Algorithmus in der Dissertation entkoppelt vollständig den Adaptionsschritt vom Approximationsschritt. Damit wird die Wiederverwertung der Lösungen aus dem Adaptionsschritt mit neuen Approximationskriterien erleichtert. Es ist mit heuristischen Methoden möglich, die beste Kombination von vorhandenen Adaptions- und Approximationskriterien für ein gegebenes Problem zu bestimmen.

Um Stabilität gegenüber Veränderungen der Nebenbedingungen zu garantieren, wird bei der Approximation auf schlecht konditionierte Hauptsysteme von Näherungslösungen geachtet. Die Funktionen des Hauptsystems von Näherungslösungen der Dgl. (1) werden erst jetzt anhand der Nebenbedingungen zur Näherungslösung für die D-finite Funktion kombiniert. So kann ein einmal gewonnenes Hauptsystem bei mehreren sich einander ähnelnden Nebenbedingungen Verwendung finden.

Im abschließenden Schritt wird der Fehler der Näherungslösung abgeschätzt. Die Eigenschaften Greenscher Funktionen ausnutzend, konnte ein Verfahren für die automatische Fehlerabschätzung von analytischen Näherungslösungen angegeben werden. Dieses Verfahren läßt sich auf den allgemeineren Fall von Rand- und Anfangswertproblemen linearer gewöhnlicher Differentialgleichungen anwenden; damit wird es möglich, die Genauigkeit der Lösungen für den speziellen Fall von Dgl.-n (1) zu kontrollieren.

Alle Algorithmen aus der Dissertation wurden in der Sprache des Computeralgebrasystems MAPLE implementiert. Das Programm nutzt Algorithmen der numerischen Mathematik und der Computeralgebra. Die Modularität der adaptiven Approximation vereinfacht Software Reengineering.

Die Art des Vorgehens in dieser Dissertation wurde durch Ideen aus der Computeralgebra und der Computeranalytik inspiriert. Es wurde gezeigt, daß beide Forschungsgebiete voneinander profitieren können und ihre Synthese zu neuen Resultaten führt.

Curriculum vitae

6. Mai 1966	Geburt in Greifswald
1984	Abitur am Institut zur Vorbereitung auf das Auslandsstudium der Martin-Luther-Universität Halle/Wittenberg
1986 bis 1991	Studium der Angewandten Mathematik an der Fakultät für Numerik und Kybernetik der Lomonossow-Universität Moskau
1991	Diplom mit dem akademischen Grad eines Magisters der technischen Wissenschaften
1991 bis 1998	Wissenschaftlicher Mitarbeiter am Institut für theoretische Informatik des Fachbereiches Informatik der Universität Rostock
seit September 1998	Berater bei der SerCon GmbH

Bibliography

[1] S. A. Abramov, M. Bronstein, and M. Petkovček. On polynomial solutions of linear operator equations. In A. Levelt, editor, *ISSAC '95, International Symposium on Symbolic and Algebraic Computation, Montreal, Canada*, pages 290–296, New York, 1995. ACM.

[2] S. A. Abramov and K. Y. Kvashenko. Fast algorithms to search for the rational solutions of linear differential equations with polynomial coefficients. In Watt [75], pages 267–270.

[3] M. A. Barkatou. Rational newton algorithm for computing formal solutions of linear differential equations. In P. Gianni, editor, *Symbolic and Algebraic Computation, International Symposium ISSAC '88, Rome, Italy, July 4–8, 1988, Proceedings*, volume 358 of *Lecture Notes in Computer Science*, pages 183–195, Berlin-Heidelberg-New York, 1989. Springer-Verlag.

[4] M. A. Barkatou. An algorithm to compute the exponential part of a formal fundamental matrix solution of a linear differential system. *Journal of Appl. Alg. in Eng. Comm. and Comp.*, 8:1–23, 1997.

[5] M. A. Barkatou. On rational solutions of systems of linear differential equations. *Journal of Symbolic Computation*, to appear.

[6] O. Becken. Simple exact solutions of simple differential equations. In *Proceedings of the 4th Rhine Workshop on Computer Algebra*, pages 81–83, 1994.

[7] O. Becken. Algorithmen zum Lösen einfacher Differentialgleichungen. Rostocker Informatik-Berichte 17, Universität Rostock, 1995.
http://www.informatik.uni-rostock.de/~obecken/.

[8] O. Becken. Ein Algorithmus zum Bestimmen aller exponentiellen Lösungen einer linearen gewöhnlichen Differentialgleichung und dessen Implementation in Reduce. In *Jahrestagung DMV 1995*, page 200, 1995.

[9] O. Becken. Exact and approximate solutions of simple DEs. In *Proceedings of the Workshop on Symbolic and Numerical Algorithms for Differential Equations (SNADE), Prague*, 1997.

[10] O. Becken and A. Jung. Error estimation in the case of linear ordinary differential equations. Rostocker Informatik-Berichte 22, Universität Rostock, 1998.

[11] E. Beke. Die Irreducibilität der homogenen linearen Differentialgleichungen. *Mathematische Annalen*, 45:278–294, 1894.

[12] C. M. Bender and S. A. Orszag. *Advanced Mathematical Methods for Scientists and Engineers.* McGraw-Hill, New York, Toronto, London, 1978.

[13] M. Bronstein. The Risch differential equation on an algebraic curve. In Watt [75], pages 241–246.

[14] M. Bronstein. Integration and differential equations in computer algebra. Программирование, 5:26–44, 1992.

[15] M. Bronstein. Linear ordinary differential equations: breaking trough the order 2 barrier. In Wang [73], pages 42–48.

[16] M. Bronstein. On solutions of linear ordinary differential equations in their coefficient field. *Journal of Symbolic Computation*, 13:413–439, 1992.

[17] M. Bronstein. An improved algorithm for factoring linear ordinary differential operators. In M. Giesbrecht, editor, *ISSAC '94, International Symposium on Symbolic and Algebraic Computation, Oxford, England*, pages 336–340, New York, 1994. ACM.

[18] M. Bronstein and M. Petkovček. On Ore rings, linear operators and factorization. Программирование, 7:27–44, 1994.

[19] M. Bronstein and M. Petkovček. An introduction to pseudo-linear algebra. *Theoretical Computer Science*, 157:3–33, 1996.

[20] D. V. Chudnovsky and G. V. Chudnovsky. On expansion of algebraic functions in power and Puiseux series, part 1. *Journal of Complexity*, 2:271–294, 1986.

[21] D. V. Chudnovsky and G. V. Chudnovsky. On expansion of algebraic functions in power and Puiseux series, part 2. *Journal of Complexity*, 3:1–25, 1987.

[22] E. Coddington and N. Levinson. *Theorie of Linear Ordinary Differential Equations.* McGraw-Hill, New York, Toronto, London, 1955.

[23] J. Della Dora and E. Tournier. Formal solutions of differential equations in the neighborhood of singular points. In P. Wang, editor, *1981 ACM Symposium on Symbolic and Algebraic Computation*, pages 25–29, New York, 1981. Academic Press.

[24] J. Della Dora and E. Tournier. Homogeneous linear difference equation (Frobenius-Boole method). In J. Fitch, editor, *EUROSAM 84, International Symposium on Symbolic and Algebraic Computation, Cambridge, England, July 9–1, 1984*, volume 174 of *Lecture Notes in Computer Science*, pages 2–12, Berlin-Heidelberg-New York, 1984. Springer-Verlag.

[25] J. Della Dora and E. Tournier. Formal solutions of linear difference equations method of Pincherle-Ramis. In B. Char, editor, *1986 ACM Symposium on Symbolic and Algebraic Computation, University of Waterloo, Ontario*, pages 192–196, New York, 1986. Academic Press.

[26] G. Frobenius. Ueber die Integration der linearen Differentialgleichungen durch Reihen. *Journal für die reine und angewandte Mathematik*, 76:214–235, 1873.

[27] L. Fuchs. Zur Theorie der linearen Differentialgleichungen mit veränderlichen Coefficienten. *Journal für die reine und angewandte Mathematik*, 66:121–160, 1866.

[28] L. Fuchs. Zur Theorie der linearen Differentialgleichungen mit veränderlichen Coefficienten. *Journal für die reine und angewandte Mathematik*, 68:354–385, 1868.

[29] K. Geddes, S. Czapor, and G. Labahn. *Algorithms for Computer Algebra.* Kluwer Academic Publishers, Boston, 1992.

[30] H. Goering. *Asymptotische Methoden zur Lösung von Differentialgleichungen.* Akademie-Verlag, Berlin, 1977.

[31] K. Hantzschmann. Zur Lösung von Randwertaufgaben bei Systemen gewöhlicher Differentialgleichungen mit dem Ritz-Galerkin-Verfahren. Habilitationsschrift Technische Universität Dresden, 1983.

[32] K. Hantzschmann. Implementierbare Fehlerabschätzungen für Näherungslösungen von Randwertaufgaben bei Systemen gewöhnlicher Differentialgleichungen. Studientexte Weiterbildungszentrum für Mathematische Kybernetik und Rechentechnik / Informationsverarbeitung 73, Technische Universität Dresden, 1984.

[33] K. Hantzschmann. Probleme und Algorithmen der Computeranalytik. In *Mitteilungen der Mathematischen Gesellschaft der DDR/Mathematikerkongress der DDR 1990, Dresden 10.-14. September 1990*, volume 2, pages 61–67. Mathematische Gesellschaft der DDR, Technische Universität Dresden, 1990.

[34] K. Hantzschmann and A. Jung. Adaptive Approximation zur näherungsweisen Lösung von Differentialgleichungen. Rostocker Informatik-Berichte 18, Universität Rostock, 1995. http://www.informatik.uni-rostock.de/˜ajung/.

[35] K. Hantzschmann and N. X. Thinh. Analytical approximate solution of singular ordinary differential equations. In D. Shirkov, V. Rostovtsev, and V. Gerdt, editors, *IV. International Conference on Computer Algebra in Physical Research, Dubna, USSR, 22–26 May 1990*, pages 169–174, Singapore, 1991. World Scientific Publishing.

[36] H. Kadner. Genähert-analytische Berechnung Greenscher Funktionen. Studientexte Weiterbildungszentrum für Mathematische Kybernetik und Rechentechnik / Informationsverarbeitung 89, Technische Universität Dresden, Sektionen Mathematik und Informationsverarbeitung, 1986.

[37] E. Kamke. *Differentialgleichungen. 1. Gewöhnliche Differentialgleichungen.* Akademische Verlagsgesellschaft Geest&Portig, Leipzig, sixth edition, 1969.

[38] E. Kamke. *Differentialgleichungen. Lösungsmethoden und Lösungen: 1. Gewöhnliche Differentialgleichungen.* B.G. Teubner, Stuttgart, tenth edition, 1983.

[39] W. Koepf. Power series in computer algebra. *Journal of Symbolic Computation*, 13:581–603, 1992.

[40] W. Koepf. Examples for the algorithmic calculation of formal Puisieux, Laurent and power series. *ACM SIGSAM Bulletin*, 27(1):20–32, 1993.

[41] W. Koepf and D. Schmersau. Spaces of functions satisfying simple differential equations. Technical report 2, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 1994. http://www.zib.de/bib/pw/.

[42] А. Н. Колмогоров and С. В. Фомин. Элементы теории функций и функционального анализа. Наука, Москва, sixth edition, 1989.

[43] J. J. Kovacic. An algorithm for solving second order linear homogeneous differential equations. *Journal of Symbolic Computation*, 2:3–43, 1986.

[44] N. J. Lehmann. Fehlerschranken für Näherungslösungen bei Differentialgleichungen. *Numerische Mathematik*, 10:261–288, 1967.

[45] N. J. Lehmann. Computer algebra and practical analysis. In B. Buchberger, editor, *EUROCAL '85, European Conference on Computer Algebra, Linz, Austria, April 1–3, 1985, Proceedings Vol. 1: Invited Lectures*, volume 203 of *Lecture Notes in Computer Science*, pages 102–113, Berlin-Heidelberg-New York, 1985. Springer-Verlag.

[46] N. J. Lehmann. Die Analytische Maschine - Grundlagen einer Computer-Analytik. In *Sitzungsberichte der Sächsischen Akademie der Wissenschaften zu Leipzig, Mathematisch- naturwissenschaftliche Klasse*, volume 118,4. Akademie Verlag, Berlin, 1985.

[47] N. J. Lehmann. Adaptive Approximation und Anwendungen. Studientexte Weiterbildungszentrum Computermathematik 105, Technische Universität Dresden, Sektion Mathematik, 1989.

[48] W. Luther et al. *Gewöhnliche Differentialgleichungen, Analytische und numerische Behandlung.* Friedrich Vieweg & Son, Braunschweig, 1987.

[49] Ø. Ore. Formale Theorie der linearen Differentialgleichungen (Erster Teil). *Journal für die reine und angewandte Mathematik*, 167:221–234, 1932.

[50] Ø. Ore. Formale Theorie der linearen Differentialgleichungen (Zweiter Teil). *Journal für die reine und angewandte Mathematik*, 168:233–252, 1932.

[51] Ø. Ore. Theory of non-commutative polynomial ring. *Annals of Mathematics*, 34:480–508, 1933.

[52] E. Pflügel. An algorithm for computing exponential solutions of first order linear differential systems. In W. W. Küchlin, editor, *ISSAC '97, International Symposium on Symbolic and Algebraic Computation, Maui, Hawaii*, pages 164–171, New York, 1997. ACM.

[53] P. Richter. *Einsatz eines analytischen Iterationsverfahrens zur Lösung von Randwertaufgaben.* PhD thesis, Technische Universität Dresden, 1982.

[54] R. Risch. The problem of integration in finite terms. *Transactions of the American Mathematical Society*, 139:167–189, 1969.

[55] B. Salvy and P. Zimmermann. Gfun: A Maple package for the manipulation of generating and holonomic functions in one variable. *ACM Transactions on Mathematical Software*, 20:163–177, 1994. http://pauillac.inria.fr/algo/papers/bibgen/salvy.html.

[56] F. Schwarz. A factorization algorithm for linear ordinary differential equations. In G. Gonnet, editor, *Proceedings of the ACM-SIGSAM 1989 International Symposium on Symbolic and Algebraic Computation, ISSAC '89, Portland, Oregon*, pages 17–25, New York, 1989. ACM.

[57] W. M. Seiler. Computer algebra and differential equations - an overview. Internal Report 97-25, Fakultät für Informatik, Universität Karlsruhe, 1997.

[58] M. F. Singer. Liouvillian solutions of nth order homogeneous linear differential equations. *American Journal of Mathematics*, 103(4):661–682, 1981.

[59] M. F. Singer. Formal solutions of differential equations. *Journal of Symbolic Computation*, 10:59–94, 1990.

[60] M. F. Singer. Liouvillian solutions of linear differential equations with Liouvillian coefficients. *Journal of Symbolic Computation*, 11:251–273, 1991.

[61] M. F. Singer. Testing reducibility of linear differential operators: A group theoretic perspective. *Journal of Appl. Alg. in Eng. Comm. and Comp.*, 7:77–104, 1996.

[62] M. F. Singer and F. Ulmer. Liouvillian solutions of third order linear differential equations: New bounds and necessary conditions. In Wang [73], pages 17–25.

[63] M. F. Singer and F. Ulmer. Galois groups of second and third order linear differential equations. *Journal of Symbolic Computation*, 16:9–36, 1993.

[64] M. F. Singer and F. Ulmer. Linear differential equations and products of linear forms. *Journal for Pure and Applied Algebra*, 117-118:549–563, 1997.

[65] R. P. Stanley. Differentiably finite power series. *European Journal of Combinatorics*, 1:175–188, 1980.

[66] N. X. Thinh. *Analytische Approximation für singuläre Anfangswertaufgaben gewöhnlicher Differentialgleichungen*. PhD thesis, Technische Universität Dresden, 1989.

[67] L. W. Thomé. Zur Theorie der linearen Differentialgleichungen. *Journal für die reine und angewandte Mathematik*, 74:193–217, 1872.

[68] E. Tournier. *Solutions Formelles d'Equations Différentielles*. PhD thesis, LMC / IMAG, Grenoble, 1987.

[69] F. Ulmer and J.-A. Weil. Note on Kovacic's algorithm. *Journal of Symbolic Computation*, 22:179–200, 1996.

[70] M. van Hoeij. *Factorization of Linear Differential Operators*. PhD thesis, Katholieke Universiteit, Nijmegen, 1996.

[71] M. van Hoeij. Rational solutions of the mixed differential equation and its application to factorization of differential operators. In Y. Lakshman, editor, *ISSAC '96, International Symposium on Symbolic and Algebraic Computation, Zurich, Switzerland*, pages 219–225, New York, 1996. ACM.

[72] M. van Hoeij. Formal solutions and factorization of differential operators with power series coefficients. *Journal of Symbolic Computation*, 24:1–30, 1997.

[73] P. Wang, editor. *ISSAC '92, International Symposium on Symbolic and Algebraic Computation, Berkeley, California*, New York, 1992. ACM.

[74] S. Watanabe. Formula manipulations solving linear ordinary differential equations. Publications of the Research Institute for Mathematical Sciences 11, Kyoto University, 1976.

[75] S. Watt, editor. *ISSAC '91, International Symposium on Symbolic and Algebraic Computation, Bonn, Germany*, New York, 1991. ACM.

[76] K. Wolf. *Effiziente Algorithmen zur Lösung linearer Differentialgleichungssysteme und zur Faktorisierung linearer Differentialoperatoren über liouvilleschen Körpern.* PhD thesis, Mathematisch-naturwissenschaftliche Fakultät der Rheinischen Friedrich-Wilhelms Universität, Bonn, 1992.

Index