

Symbolic Algorithms for the Local Analysis of Systems of Pseudo-Linear Equations

Gary John Broughton

A thesis submitted in partial fulfilment of the requirements of
Kingston University for the degree of
Doctor of Philosophy
in
Computer Science

May 2013

KINGSTON UNIVERSITY LIBRARY	
Acc. No.	
Class No.	

Abstract

This thesis is concerned with the design and implementation of algorithms in Computer Algebra - a discipline which pursues a symbolic approach to solving mathematical equations and problems in contrast to computing solutions numerically. More precisely, we study systems of pseudo-linear equations, which unify the classes of linear differential, difference and q-difference systems. Whilst the classical mathematical theory of asymptotic expansions and the notion of formal solutions of this type of solutions are well established for all these individual cases, no unifying theoretical framework for pseudo-linear systems was known prior to our work.

From an algorithmic point of view, the computation of a complete fundamental system of formal solutions is implemented by the formal reduction process. The formal reduction of linear differential systems had been treated in the past, and linear difference systems were also investigated and partly solved. In the case of linear q-difference systems, the structure of the formal solution is much easier which results in an alleviated formal reduction. However, no satisfying algorithm had been published that would be suitable to compute the formal solutions.

We place ourselves in the generic setting and show that various algorithms that are known to be building blocks for the formal reduction in the differential case can be extended to the general pseudo-linear setting. In particular, the family of Moser- and super-reduction algorithms as well as the Classical Splitting Lemma and the Generalised Splitting Lemma are amongst the fundamental ingredients that we consider and which are essential for an effective formal reduction procedure. Whereas some of these techniques had been considered and adapted for systems of difference or q-difference equations, our novel contribution is to show that they can be extended and formulated in such a way that they are valid generically.

Based on these results, we then design our generic formal reduction method, again inspired by the differential case. Apart from the resulting unified approach, this also yields a novel approach to the formal reduction of difference and q-difference systems.

Together with a generalisation of an efficient algorithm for computing regular formal solutions that was devised for linear differential systems, we finally obtain a complete and generic algorithm for computing formal solutions of systems of pseudo-linear equations. We show that we are able to compute a complete basis of formal solutions of large classes of

linear functional systems, using our formal reduction method.

The algorithms presented in this thesis have been implemented in the Computer Algebra System Maple as part of the Open Source project ISOLDE.

Acknowledgements

I would first like to thank my supervisors Dr. Eckhard Pfluegel, Dr. Mark Joy and Dr. Nigel Atkins, especially Dr. Eckhard Pfluegel whose support has been essential. I would also like to thank Prof. Moulay Barkatou for his valuable insights.

My experiences at Kingston University have had a tremendous impact on my education and on my life in general. I have met some wonderful people who will remain friends for life through the bonds of shared experience.

Many others at Kingston University have shown me kindness and in particular, I would like to offer my gratitude to Dr. Vincent Lau, Dr. Gordon Hunter and Prof. Graeme Jones.

The work presented in this thesis was supported by a Doctoral Training Award from the Engineering and Physical Sciences Research Council (EPSRC) of the United Kingdom.

From a personal perspective I would like to thank my family and extended family for their tireless support and encouragement throughout these years.

Above all, my thanks and love go to my wife Emma. Without her love, kindness and understanding, my life would not be what it is today.

Contents

1	Introduction	4
1.1	Motivation	4
1.2	Aims and Objectives	5
1.3	Contributions of this Thesis	6
1.4	Overview of Dissertation	7
2	Mathematical Framework	9
2.1	Differential, Difference and q -difference Calculus	10
2.2	Systems of Linear Differential, Difference and q -Difference Equations	14
2.3	Pseudo-derivations	17
2.4	Systems of Pseudo-linear Equations	19
2.5	Local (ϕ, δ) -fields	20
2.6	Singular Systems of Pseudo-Linear Equations	22
2.7	Computing Formal Solutions	23
2.8	Review of Computer Algebra Techniques	25
3	Super-Irreducible Forms of Systems of Pseudo-Linear Equations	27
3.1	Introduction	27
3.2	Moser-Irreducible Forms	28
3.3	The Moser-Reduction Algorithm	30
3.3.1	A Generalisation of the Moser-Reduction Algorithm	32
3.4	Super-Irreducible Forms	35
3.4.1	The Second Reduction Algorithm	35
3.4.2	A First Application: Polynomial Solutions of Linear q -Difference Systems	37
3.4.3	The Indicial Equation	37
4	Regular Formal Solutions	40
4.1	Solving Systems of the First Kind	40
4.2	The Dimension of the Regular Solution Space	50
4.2.1	Simple Pseudo-Linear Systems	50

4.3	The Algorithm: Monomial-by-Monomial Method	53
4.3.1	Idea of the Algorithm	54
4.3.2	Computing the Series Solutions	56
4.3.3	An Example	58
4.3.4	Discussion	60
4.4	Regular Solutions of Various Linear Functional Systems	62
5	Irregular Formal Solutions	64
5.1	Introduction	64
5.2	Formal Reduction of Pseudo-Linear Systems	65
5.2.1	Formal Solutions	66
5.2.2	Formal Reduction	68
5.2.3	Main Result	73
5.3	Formal Reduction Algorithm	75
5.3.1	A Generalised Splitting Lemma	76
5.3.2	Nilpotent Case	78
5.3.3	Discussion	79
5.4	Computing Formal Solutions	83
5.4.1	The Method	84
5.5	Resolving Irregular Parts of Some Linear Functional Systems	85
5.5.1	Linear Differential Systems	85
5.5.2	Linear Difference Systems	86
5.5.3	Linear q -Difference Systems	88
6	Conclusion	90
7	List of Terms	93

Chapter 1

Introduction

1.1 Motivation

Computer Algebra is a discipline at the interface of computer science and applied mathematics. It studies the design and implementation of symbolic algorithms for solving mathematical problems. Here, symbolic means performing exact computations with symbols or parameters in the same way as a mathematician would do with pencil and paper.

Solving systems of differential, or more generally, functional equations has always been an interesting topic for Computer Algebra research. The main advantage of the symbolic approach compared to solving numerically is that solutions are exact, may contain parameters and are written in closed form.

One of the main problems in Computer Algebra is *intermediate expression swell*. This is caused by the overhead that arises when having to deal with complex symbolic data during execution of an algorithm. Particular care needs to be taken when designing, and certainly when implementing algorithms in Computer Algebra.

This is particularly important for the efficient symbolic treatment of a particular class of functional systems, which is treated in this dissertation. We are concerned with the symbolic formal reduction of systems of pseudo-linear equations with formal power series coefficients. In particular, we develop a unifying approach to the computation of local solutions of linear systems of differential, difference and q -difference equations. Based on existing algorithmic results on the algorithmic formal reduction of linear differential systems, and using the unified approach, we are able to design a novel, generic method that greatly improves existing algorithms in the difference case and provides the first method available for q -difference systems.

The idea to use a unifying approach has been employed in the past. [67] uses “near-similarity” transformations in order to adopt Turrittin’s [69] and Wasow’s [74] classical approaches to compute formal solutions of a variety of linear functional equations. Our work is using *pseudo-linear algebra* [52], which has created interest in Computer Algebra before [36], with additional advantage that it covers the q -difference case. Instead of the classical approach, we start from more recent algorithmic work [14], [63], [20].

Pseudo-linear algebra has origins from the early part of the 20th century in works from Ore [59] and Jacobson [52]. The key aspect of this approach is the fact that the theory of differential, difference and q -difference calculus has many similarities. Many Computer Algebra algorithms for solving linear differential, difference and q -difference equations also show similarities [1]. [35, 75] have succeeded in generalising algorithmic treatments of differential and difference equations to the general pseudo-linear case.

The case of first-order linear differential, difference and q -difference systems however, has been lagging behind. The starting point of this dissertation was that although [63, 27] had also been adapted to the difference case [11, 17, 18], no results seemed published on the q -difference systems and the unifying view for the pseudo-linear case had not been adopted.

The approach we undertake is heavily relying on the concepts of Moser- and super-irreducible forms [58, 49], which we generalise to systems of pseudo-linear equations. This enables using and adopting many insights that have already proven to be helpful for the formal reduction of differential systems. Note that for us, solving systems excludes the so-called cyclic vector approach [44] which converts the system to a scalar n th-order equation. In order to avoid this time consuming and costly conversion process, we prefer the direct manipulation of the system by transforming it to Moser- and super-irreducible forms, which then in turn allows further algorithmic treatment.

The algorithms developed in this thesis have been implemented in the Computer Algebra system Maple and are contained in the latest version of the Maple package ISOLDE (Integration of Systems of Ordinary Linear Differential Equations) [22]. ISOLDE was originally developed for linear differential systems, and our contribution has extended its functionality to cover difference and q -difference systems.

1.2 Aims and Objectives

As we have mentioned already, the research carried out for this PhD is concerned with the design of symbolic algorithms for systems of linear functional equations with a particular

emphasis on efficient and robust implementation. The aim of our work is hence, to improve the existing Computer Algebra functionality for solving equations such as linear differential, difference and q -difference systems locally. This is achieved by exploiting the unifying concept of pseudo-linear systems and a developed local framework for computing formal invariants at singularities.

The work we have undertaken can be summarised by the following research objectives:

1. Design a reduction algorithm for the classification of singular points of given systems of pseudo-linear equations with coefficients in an arbitrary local field, based on generalising the Moser-reduction method known for the differential case.
2. Devise an algorithm for the computation of regular series solutions of given systems of pseudo-linear equations with formal power series coefficients, based on extending the “monomial-by-monomial” method for the computation of regular formal solutions of systems of linear differential equations.
3. Provide an efficient method for computing irregular formal solutions of systems of pseudo-linear equations, based on a formal reduction process for the input system.
4. Provide a robust implementation of these algorithms in the Computer Algebra system Maple by integrating novel functionality within the ISOLDE package.

1.3 Contributions of this Thesis

Due to the nature of this PhD topic, the novel contributions we have made comprise several aspects.

From a mathematical point of view, our main contributions are the adaptation of Moser’s and Hilali’s reduction principle to pseudo-linear systems; the unifying framework for defining regular, irregular and hyperexponential parts of these types of systems, leading to a generic definition of formal solutions and a complete formal reduction method.

The computer science aspect of this work is the algorithm and data structure design of the local reduction methods. Rather than basing them on the constructive proofs that we have given in various places, we design refined methods that are far more effective. We employ the mechanism of *lazy evaluation*, a popular Computer Algebra technique when performing arithmetic with power series containing an infinite number of coefficients, in order to minimise computational overhead. We adapt existing code from ISOLDE to this purpose.

This leads to the practical contribution in form of the new functionality of ISOLDE: computing formal solutions is now possible for large classes of linear functional systems, to be used by anyone who wishes to download the package freely from the internet.

1.4 Overview of Dissertation

This thesis is organised as follows: Chapter 2 sets the mathematical framework in which we base the algorithmic approach seen later in this work. We start with a motivational section on differential, difference and q -difference calculus including some examples. We briefly review local fields, pseudo-linear algebra and related notions. We define local systems of pseudo-linear equations and introduce pseudo-linear transformations.

Chapter 3 contains our first results. We define a suitable reduction principle for pseudo-linear systems, based on Moser's definition in the differential case. We then generalise this principle to the concept of super-irreducible forms analogously to Hilali and Wazner's work in the differential case. A constructive reduction criterion is given, computed from a finite number of coefficients of the system matrix, to decide whether a given system is reducible or not. This leads to two algorithms to reduce an arbitrary system of pseudo-linear equations into Moser- and super-irreducible forms.

In Chapter 4 we tackle the computation of actual solutions by, in the first instance, restricting ourselves to regular formal solutions. We devise an algorithm for the computation of this type of solutions, based on extending the "monomial-by-monomial" method for the computation of regular formal solutions of systems of linear differential equations [21]. In this chapter we consider a more restricted local coefficient field (the field of formal Laurent series with finite pole order with coefficients in a field of characteristic zero) and define and compute regular solutions. We derive a set of quantities that characterise the local behaviour of systems of pseudo-linear equations. We define pseudo-linear systems of the first kind and show that we can derive, from the local quantities, a fundamental system of regular solutions. This requires a definition of the indicial polynomial $\varphi(\lambda)$ and some additional functions e_λ and u . For arbitrary pseudo-linear systems we then show that computing a super-irreducible form is sufficient to compute $\varphi(\lambda)$ and to determine the number of linearly independent regular solutions. The proof is based on a generalised Splitting Lemma. We then design an algorithm to compute all regular solutions monomial-by-monomial which is inspired by a previous method ([21]) for the differential case.

The next step that immediately follows from the results of Chapter 4 is the study of the case of an irregular singularity. This is the subject of Chapter 5. We define a unifying notion

of hyperexponential parts and construct, under certain hypotheses, a full basis of formal solutions of arbitrary pseudo-linear systems in the neighbourhood of an irregular singularity. This is done using a constructive proof, which could also be implemented algorithmically. However, the resulting approach would lack efficiency. We therefore give a more practical algorithm for computing hyperexponential parts. We then show how to construct irregular parts from hyperexponential parts. Together with the result from Chapter 4, this yields a complete algorithm for computing formal solutions.

The results contained in Chapter 3 and Chapter 4 of this thesis form part of published work ([25, 9]). At the time of writing, we are planning to submit the work presented in Chapter 5 to a Computer Algebra journal.

Chapter 2

Mathematical Framework

The mathematical contribution of this thesis is to develop a unifying framework for the local analysis of systems of pseudo-linear equations. In particular, we consider linear differential, difference and q -difference systems as the main source of insights and inspiration throughout this work. It is well-known that the theoretical background from which algorithms treating the individual cases have been developed, have similarities. These common traits suggest the possibility of a unifying approach.

Before a unifying framework can be considered, let us first understand some of the similarities encountered when studying differential, difference and q -difference equations.

This chapter will outline the basic mathematical concepts used in this thesis beginning with a discussion and comparison of ideas behind common derivations. The main theoretical tool encountered later in this thesis will be *pseudo-derivations* which shall be reviewed in the context of Computer Algebra acting on elements of local fields later in this chapter. In particular, we shall give a suitable definition of a *system of pseudo-linear equations* which will act as an input system for the later algorithms.

The results obtained here shall form the basis of the common approach used to develop the algorithms in the rest of this thesis. We shall see that our definition of a system of pseudo-linear equations will allow us to give a set of parameters whose properties characterise each individual type of system encompassed by our unifying approach.

The concept of the standard calculus is well-known, the aim of this section is not to formally give the definitions of these concepts, but to discuss the ideas behind the standard calculus, difference calculus and q -difference calculus. This will enable us to better understand the motivation behind the definition of *pseudo-derivations*. The mathematical framework presented at the end of this chapter is based heavily on these concepts and the purpose of this section is to provide motivation and to give a sense of the unifying view this

framework creates.

2.1 Differential, Difference and q -difference Calculus

In this section, we introduce the standard, difference and q -difference calculus and point out the similarities between these different concepts.

In the standard calculus, a derivative can be seen to be a measure of how a function changes due to a change of input and is defined by the limit (assuming the limit $h \rightarrow 0$ exists):

$$Df(x) = \frac{dy}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}. \quad (2.1)$$

From this definition, it is easy to arrive at general rules for differentiation

- $D[f(x) + g(x)] = Df(x) + Dg(x)$
- $D[af(x)] = aDf(x)$ $a = \text{constant}$
- $D[f(x)g(x)] = f(x)Dg(x) + Df(x)g(x)$ (Leibniz's rule)
- $D\left[\frac{f(x)}{g(x)}\right] = \frac{Df(x)g(x) - f(x)Dg(x)}{[g(x)]^2}$ (quotient rule)
- $D[f(x)]^n = n[f(x)]^{n-1}Df(x)$ (chain rule)

and to give the derivatives of some common functions:

- $f(x) = c, D[c] = 0$
- $f(x) = x^n, D[x^n] = nx^{n-1}$
- $f(x) = e^{rx}, D[e^{rx}] = re^{rx}$
- $f(x) = \ln x, D[\ln x] = \frac{1}{x}$

Apart from the standard derivative, there are many other different types of derivation, for example, we shall now examine a certain type of difference operator. Given a function $f(x)$, we define the difference operator Δ

$$\Delta f(x) = f(x+1) - f(x). \quad (2.2)$$

This can be seen as a derivative as it can be seen to be a measure of how the function $f(x)$ changes based on the automorphism $x \mapsto x+1$ of the input of the function. With this particular derivative, we can develop a sense of *difference calculus*. In an analogous manner

as the standard derivative, higher order differences can be taken, for example, taking two successive differences, we obtain

$$\Delta^2 f(x) = \Delta[\Delta f(x)] = \Delta[f(x+1) - f(x)] = f(x+2) - 2f(x+1) + f(x) \quad (2.3)$$

and in general we have,

$$\Delta^n f(x) = \Delta[\Delta^{n-1} f(x)]. \quad (2.4)$$

We can see that the following general rules of difference calculus bear a close resemblance to the rules of standard calculus:

- $\Delta[f(x) + g(x)] = \Delta f(x) + \Delta g(x)$
- $\Delta[af(x)] = a\Delta f(x)$ where a is a constant
- $\Delta[f(x)g(x)] = f(x)\Delta g(x) + f(x+1)\Delta f(x)$ (analog of Leibniz's rule)
- $\Delta\left[\frac{f(x)}{g(x)}\right] = \frac{\Delta f(x)g(x) - f(x)\Delta g(x)}{g(x)g(x+1)}$ (analog of quotient rule)

When applying the difference operator Δ to common functions, we find that they do not resemble the effect of applying the standard differential operator D . To find an analogue, define the *falling factorial function*, which takes the form

$$x^{[k]} = x(x-h)(x-2h)\cdots(x-[k-1]h), \quad x^{[0]} = 1 \quad (2.5)$$

and for negative integers

$$x^{[-k]} = \frac{1}{x(x-h)(x-2h)\cdots(x-[k-1]h)} \quad (2.6)$$

Note that, as $h \rightarrow 0$ in (2.5) and (2.6), we obtain $x^{[k]} \rightarrow x^k$ and $x^{[-k]} \rightarrow x^{-k}$.

However, for $h = 1$, we see that

$$\Delta x^{[k]} = kx^{[k-1]} \quad (2.7)$$

which is analogue of the effect of the standard differential operator. If we wish to obtain a similar analogue with respect to more general functions $f(x)$, we can extend the definition of the factorial function to arbitrary functions

$$f^{[k]}(x) = f(x)f(x-1)f(x-2)\cdots f(x-[k-1]) \quad (2.8)$$

and

$$f^{[-k]}(x) = \frac{1}{f(x)f(x-1)f(x-2)\cdots f(x-[k-1])}. \quad (2.9)$$

Note that as $h \rightarrow 0$, $f^{[k]}(x) \rightarrow f(x)^k$ and $f^{[-k]}(x) \rightarrow f^{-k}(x)$. As with the standard derivative, we can now list formulae of common functions when applying the difference operator Δ .

- $\Delta[c] = 0$
- $\Delta[x^{[n]}] = nx^{[n-1]}h$
- $\Delta[e^{rx}] = e^{rx}(e^{rh} - 1)$
- $\Delta[\ln x] = \ln(1 + \frac{h}{x})$.

The above shows that there are many similarities between standard differential operator and the difference operator. Indeed, there are other well known discrete analogues of the standard derivative such as in quantum calculus [53].

We have seen that the difference operator can be seen as a measure of how a function changes when its input is perturbed by a particular automorphism of its input, however it is also possible to study how a function changes when other types of automorphism are applied to its input. For example, consider the automorphism $x \mapsto qx$ (where $q \neq 1$ and $q \neq 0$). Define

$$\Delta_q[f(x)] = f(qx) - f(x), \quad (2.10)$$

and therefore the q -derivative is defined as

$$\frac{\Delta_q[f(x)]}{\Delta_q x} = \frac{f(qx) - f(x)}{qx - x}. \quad (2.11)$$

In order to investigate properties of this derivative, certain q -analogues of other common mathematical objects have been defined. The first of these q -analogues will be the q -bracket or q -number defined (for $n \in \mathbb{Z}$) as

$$[n]_q = \frac{1 - q^n}{1 - q}. \quad (2.12)$$

This can be seen as a basic q -analogue of an integer. An important property of the q -bracket is to note that

$$\lim_{q \rightarrow 1} [n]_q = n.$$

The following well-known properties hold:

1. $[\lambda]_1 := \lambda = \lim_{q \rightarrow 1} \frac{1 - q^\lambda}{1 - q}$.
2. $[\lambda]_q = 1 + q + \dots + q^{\lambda-1}$ for $\lambda \in \mathbb{N}^+$.
3. $[0]_q = 0$.

4. $[\lambda_1 + \lambda_2]_q = [\lambda_1]_q + q^{\lambda_1}[\lambda_2]_q$ for any λ_1, λ_2 .

Also, there are several q -analogues of the exponential functions some with slightly different properties. What they all have in common is the fact that as $q \rightarrow 1$, the q -exponential tends to the usual exponential function e^x . The most obviously analogous q -analog to the exponential function is the q -exponential function denoted $e_q[x]$, and this differs from the usual exponential function by replacing the factorial in the series expansion by the q -factorial, i.e. it is defined as

$$e_q[x] = \sum_{n=0}^{\infty} \frac{x^n}{[n]_q!}. \quad (2.13)$$

There is also the closely related function

$$E_q[x] = \sum_{n=0}^{\infty} \frac{q^{n(n-1)/2} x^n}{(q; q)_n} \quad (2.14)$$

where $(q; q)_n = \prod_{k=0}^{n-1} (1 - q^{k+1})$ is the q -Pochhammer symbol. These functions are related by

$$e_q[z] = E_q[z(q-1)] \quad (2.15)$$

In the above definitions, the q -factorial is defined as

$$[n]_q! = [n]_q [n-1]_q \cdots [1]_q = \frac{1-q^n}{1-q} \frac{1-q^{n-1}}{1-q} \cdots \frac{1-q}{1-q}. \quad (2.16)$$

With these q -analogues and others, it is possible to show that the q -derivative displays properties analogous to those in the study of the standard and difference calculus. In this sense it is possible to define the notion of q -calculus. In a similar manner as differential calculus and difference calculus, it is possible to find similarities to standard and difference calculus when examining a q -derivatives of common functions. For example, the effect of applying the q -derivative (2.11) to the function x^k where k is a positive integer is

$$\Delta_q[x^k] = \frac{q^k - 1}{q - 1} x^{k-1} = [k]_q x^{k-1}$$

where $[k]_q$ is the q -bracket of k . We also see a similar analogy when considering the q -exponential (2.13). Indeed, we observe in the standard calculus that applying the derivative to the exponential function results in the exponential function (so it is an eigenfunction of the derivative D). The same is true when applying Δ_q to the q -exponential $e_q(x)$, hence it is an analogous eigenfunction.

2.2 Systems of Linear Differential, Difference and q -Difference Equations

The work in this thesis will concern algorithmic techniques for solving linear functional systems including linear systems of differential, difference and q -difference systems. A linear system of ordinary differential equation is of the form

$$Y'(x) = A(x)Y(x)$$

where $A(x)$ is an $n \times n$ dimensional matrix. Similarly a system of linear difference or q -difference equations is of the form

$$\phi Y(x) = A(x)Y(x)$$

where ϕ is defined as an automorphism of the input variable of the form $\phi x = x + h$ or $\phi x = qx$ with $h \neq 0$ ($q \neq 0$ respectively).

In order to illustrate how to solve very simple classes of the above systems, we consider the case of constant coefficients. The key to finding solutions in this case is to find an appropriate eigenfunction. This is a non-zero function which is returned in the same form (except for some multiplicative scaling factor) under the action of a linear operator. For example, an eigenfunction of the linear differential operator $\frac{d}{dx}$ is $\exp(cx)$.

Similarly, an eigenfunction of the difference operator ϕ where $\phi x = x + 1$ is ρ^x where ρ is some constant. In the system case, there are several methods for finding solutions involving eigenfunctions. In order to understand the similarities in techniques for solving these different types of linear system, consider the following simple examples:

Example 2.2.1 (Linear Differential System with Constant Coefficients). *Consider the linear differential system*

$$Y' = CY \tag{2.17}$$

where C is the constant matrix

$$C = \begin{pmatrix} 2 & 1 \\ & 2 & 1 \\ & & 2 \end{pmatrix}. \tag{2.18}$$

We wish to find a series solution of the above system of the form $Y = e^{Cx}$. We notice that in this example, we can write this solution in the form $Y = e^{(2I+N)x}$ where N is the nilpotent

matrix

$$N = \begin{pmatrix} 0 & 1 & \\ & 0 & 1 \\ & & 0 \end{pmatrix}. \quad (2.19)$$

Therefore, we can see that

$$\begin{aligned} Y &= e^{2Ix} \left[I + Nx + \frac{N^2x^2}{2!} \right] \\ &= e^{2Ix} \left[I + \begin{bmatrix} 0 & x & \\ & 0 & x \\ & & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & \frac{1}{2}x^2 \\ & 0 & 0 \\ & & 0 \end{bmatrix} \right] \\ &= e^{2x} \begin{bmatrix} 1 & x & \frac{1}{2}x^2 \\ & 1 & x \\ & & 1 \end{bmatrix} \end{aligned}$$

and by taking column vectors of (2.20) we find the closed-form series solution

$$Y = e^{2x} \left(C_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + C_2 \begin{bmatrix} x \\ 1 \\ 0 \end{bmatrix} + C_3 \begin{bmatrix} \frac{1}{2}x^2 \\ x \\ 0 \end{bmatrix} \right) \quad (2.20)$$

Similar eigenfunction techniques can be used in the difference case when considering systems with constant coefficients. The following example helps to illustrate the similarities in the approaches and gives the reader an insight into the work contained in the rest of this thesis.

Example 2.2.2 (Linear Difference System with Constant Coefficients). *Consider the linear difference system*

$$\phi(Y) = CY \quad (2.21)$$

where the matrix C is the constant diagonal matrix

$$C = \begin{pmatrix} 2 & & \\ & 2 & \\ & & 2 \end{pmatrix}. \quad (2.22)$$

As this matrix is diagonal, we can use the eigenfunction ρ^x with $\rho = 2$, since then

$$\phi(C^x) = \begin{pmatrix} 2^{x+1} & & \\ & 2^{x+1} & \\ & & 2^{x+1} \end{pmatrix} = C \cdot C^x.$$

If C is not diagonal then one can proceed similarly as in the differential case, the resulting solution is somewhat more complicated.

We will give another example where eigenfunctions can be used to construct a fundamental matrix solution: differential systems *of the first kind*. They are characterised by the fact that their system matrix has a simple pole at $x = 0$.

Example 2.2.3 (Differential System of the First Kind). *Consider the system*

$$xY' = CY. \quad (2.23)$$

The matrix C is again the constant matrix

$$C = \begin{pmatrix} 2 & 1 & \\ & 2 & 1 \\ & & 2 \end{pmatrix}. \quad (2.24)$$

We look for a solution of the form

$$Y = x^C \quad (2.25)$$

where, in this example, the matrix C can be written in the form $C = 2I + N$ where I is the identity matrix and N is the nilpotent matrix

$$N = \begin{pmatrix} 0 & 1 & \\ & 0 & 1 \\ & & 0 \end{pmatrix}. \quad (2.26)$$

From the definition of x^C we find

$$\begin{aligned} Y &= x^{2I+N} \\ &= e^{(2I+N)\log x} \\ &= e^{2I\log x} \cdot e^{N\log x}. \end{aligned}$$

Examining the first exponential in this product we find

$$\begin{aligned} Y &= e^{2I\log x} \cdot e^{N\log x} \\ &= e \begin{bmatrix} \log x^2 & & \\ & \log x^2 & \\ & & \log x^2 \end{bmatrix} \cdot e^{N\log x} \\ &= x^2 I \cdot e^{N\log x}. \end{aligned}$$

Now, examining the second exponential, we notice that N is nilpotent (since $N^3 = 0$). There-

fore taking the series expansion of $e^{N \log x}$, we find the finite series

$$e^{N \log x} = I + N \log x + \frac{N^2(\log x)^2}{2},$$

and the product becomes

$$\begin{aligned} Y &= x^2 I \left[I + N \log x + \frac{N^2(\log x)^2}{2} \right] \\ &= x^2 \begin{bmatrix} 1 & \log x & \frac{1}{2} \log^2 x \\ & 1 & \log x \\ & & 1 \end{bmatrix}. \end{aligned}$$

Taking column vectors, we find the closed-form series solution

$$Y = x^2 \left(C_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + C_2 \begin{bmatrix} \log x \\ 1 \\ 0 \end{bmatrix} + C_3 \begin{bmatrix} \frac{1}{2} \log^2 x \\ \log x \\ 0 \end{bmatrix} \right).$$

In Chapter 4, we will introduce linear difference and q -difference systems of the first kind and show how solutions can be constructed in these cases.

2.3 Pseudo-derivations

Given the similar techniques involved when looking for solutions of the types of linear functional systems presented in the previous section and our aim of finding a unifying algorithmic treatment of these systems, it is required that we first find a suitable mathematical framework so that we may consider these various linear operators within the definition of a single object. This motivates the definition of a generalised form of a derivation, known as *pseudo-derivation*. The first work in this area originated from Ore [59] who examined the n th order linear equation case and Jacobson [52] who considered a general type of transformation known as *pseudo-linear transformations* when studying the system case. In these works, a comparison of algebraic properties of differential and difference equations was made which suggested a unifying mathematical concept was possible. In more recent times, Bronstein and Petkovšek [36] provide an introduction to the basic objects of pseudo-linear algebra in the context of Computer Algebra and also give a description of some initial algorithms based on these objects. For further reading on pseudo-linear algebra we refer to the introductory work by [52], the modern exposition in the context of Computer Algebra by [36], [16] and [25].

Remark 2.3.1. Note that another approach for systems, based on “near-similarity” transformations [67], views the transformations used for solving linear functional systems as a

perturbation of algebraic similarity transformations. A drawback of this method is that it does not include the q -difference case.

For the purposes of this work, we shall use this Computer Algebra approach to form the basis of the local algorithms presented later in this thesis. Firstly, we shall utilise the work of [36] to give a suitable definition of a *system of pseudo-linear equations* defined over a suitable local field. We will then be able to view different types of linear functional systems within the framework of this definition, which will enable us to develop algorithms which take systems of pseudo-linear equations as input. For now, we will outline the main properties of pseudo-derivations.

Let F be a field of characteristic 0 and ϕ an automorphism of F . The fields that are relevant for this thesis are $F = K(x)$, the field of rational functions over K in the variable x , the field of formal Laurent series $F = K((\tau))$ and that of formal Puiseux series $F = K((\tau^{1/s}))$ with $s \in \mathbb{N}^*$. Here, K is a field of characteristic zero, not necessarily algebraically closed. We shall denote \bar{K} its algebraic closure.

Definition 2.3.1 ([36]). *A pseudo-derivation w.r.t. ϕ is any map $\delta : F \rightarrow F$ satisfying*

$$\delta(a + b) = \delta a + \delta b \quad (2.27)$$

and

$$\delta(ab) = \phi(a)\delta(b) + \delta(a)b, \quad \forall a, b \in K. \quad (2.28)$$

If $\phi = \text{id}$, then equation (2.28) is the Leibniz rule of products and δ is simply a derivation in the standard way. If $\phi \neq \text{id}$ then it can be shown that δ takes a specific form. This result, and subsequent proof is contained in [36] and shall be stated below.

Lemma 2.3.1 (Lemma 1 [36]). *Let F be a field as in Definition 2.3.1, ϕ an automorphism of F and δ a pseudo-derivation of F . If $\phi \neq \text{id}$ then there is an element $\gamma \in F$ such that*

$$\delta = \gamma(\phi - \text{id}). \quad (2.29)$$

Proof Since the field F is commutative, we have the relation $\delta(ab) = \delta(ba)$ for any $a, b \in F$, so we can apply (2.28) to both sides. This gives, after rearranging:

$$(\phi(a) - a)\delta(b) = (\phi(b) - b)\delta(a). \quad (2.30)$$

From (2.30), the result follows: If $\phi \neq \text{id}$ then there exists $\tilde{a} \in F$ such that $\phi(\tilde{a}) \neq \tilde{a}$. Therefore, if $\gamma = \delta(\tilde{a})/(\phi(\tilde{a}) - \tilde{a})$ then it follows that $\delta(b) = \gamma(\phi(b) - b)$ for all $b \in F$ and hence $\delta = \gamma(\phi - \text{id})$. \square

If $\phi = \text{id}$, in the literature, one commonly refers to the pair (F, δ) as a *differential field*, whereas for $\phi \neq \text{id}$, (F, ϕ) is denoted as a ϕ -*difference field*. We introduce the following

Definition 2.3.2. *We unify the notion of differential field and ϕ -difference field by denoting the field (F, ϕ, δ) as a (ϕ, δ) -field.*

Definition 2.3.3. *Given a (ϕ, δ) -field (F, ϕ, δ) , the field $K \subset F$ satisfying $\phi|_K = \text{id}$ and $\delta|_K = 0$ is called the field of constants.*

2.4 Systems of Pseudo-linear Equations

We will now introduce systems of pseudo-linear equations as a unifying way of expressing differential, difference and q -difference systems.

Definition 2.4.1. *A system of pseudo-linear equations over (F, ϕ, δ) is a system of the form*

$$\delta Y(x) = A(x)\phi Y(x) \tag{2.31}$$

with $A \in F^{n \times n}$ and Y a vector of n unknowns.

Remark 2.4.1. *If $\phi \neq \text{id}$, we can divide (2.29) by γ , hence we can without loss of generality (and will in the remainder of the thesis) assume that $\gamma = 1$.*

In this thesis, we will consider systems of pseudo-linear equations of the form $\delta Y = A\phi Y$ with $A \in \mathcal{M}_n(F)$. For $\phi \neq \text{id}$, one could (and other authors have done so) also consider systems of the form

$$\phi Y(x) = A(x)Y(x) \tag{2.32}$$

and

$$\delta Y(x) = A(x)Y(x). \tag{2.33}$$

In fact, all these different input forms are equivalent and conversions between (2.31), (2.32) and (2.33) can be carried out. In this dissertation, our view is that the form (2.31) is the most natural one and lends itself best for the algorithms designed for our work.

The conversion from (2.32) and (2.33) into the form (2.31) used in this paper could be done in several ways. We shall explain an approach that avoids matrix inversion and is particularly suitable for the implementation of our algorithms. Note that since $\phi \neq \text{id}$ we have $\delta = \gamma(\phi - \text{id})$ for some $\gamma \in F$.

Case 1

To convert from a system of the form $\phi Y = AY$ to a system of our form, rewrite it as

$$Y = \phi^{-1}(A)\phi^{-1}(Y). \quad (2.34)$$

We set $\tilde{\phi} = \phi^{-1}$ and introduce the new pseudo-derivation $\tilde{\delta} = \gamma(\tilde{\phi} - \text{id})$, leading to the identity $Y = \tilde{\phi}Y - \gamma^{-1}\tilde{\delta}Y$. Substituting this for Y in the LHS of (2.34) and rearranging yields the new system

$$\tilde{\delta}Y = B\tilde{\phi}Y \quad (2.35)$$

where $B = \gamma(I - \tilde{\phi}(A))$.

Case 2

Given a system of the form $\delta Y = AY$, using the identity $\delta Y = \gamma(\phi Y - Y)$ and applying ϕ^{-1} , we obtain a system of the form $\tilde{\delta}Y = B\tilde{\phi}Y$ where $\tilde{\delta} = \gamma(\text{id} - \phi^{-1})$, $\tilde{\phi} = \phi^{-1}$ and $B = \phi^{-1}A$.

The following example shows how to convert a difference system given in form (2.32) to a pseudo-linear system in form (2.31).

Example 2.4.1. *Assume we are given the linear difference system*

$$Y(x+1) = A(x)Y(x).$$

By subtracting $Y(x)$ on both sides of the equation, we can write this as

$$Y(x+1) - Y(x) = (A(x) - I)Y(x).$$

The substitution $x \mapsto x - 1$ yields a new system

$$Y(x) - Y(x-1) = (A(x-1) - I)Y(x-1).$$

Finally, we obtain a pseudo-linear system of the form $\delta Y(x) = B(x)\phi Y(x)$ by setting $\phi : x \mapsto x - 1$, $\delta = \text{id} - \phi$ and $B = \phi(A) - I$.

2.5 Local (ϕ, δ) -fields

In this section, we will introduce a fundamental aspect of our work which is the consideration of local aspects of (ϕ, δ) -fields with the most general situation where $F = K((\tau^{1/s}))$ and $s \in \mathbb{N}^*$ a positive natural number. The main idea is built around the concept of valuations.

Initially, we will assume $s = 1$ and then remark that a generalisation for $s > 1$ does not pose any fundamental problems.

Definition 2.5.1. *The valuation map $v : F \longrightarrow \mathbb{Z} \cup \{\infty\}$ is defined by*

$$v(f) = \begin{cases} m & \text{if } f = \tau^m(f_0 + f_1\tau + \dots), f_0 \neq 0, \\ \infty & \text{if } f = 0. \end{cases}$$

Also, we define the leading coefficient $\text{lc}(f)$ for $f \neq 0$ by $\text{lc}(f) = f_0$.

We introduce some more terminology.

Definition 2.5.2. *The valuation ring is the ring $\mathcal{O} = K[[\tau]]$. The evaluation map $\pi : \mathcal{O} \rightarrow K$ defined by $\pi(f) = F(0)$ is a homomorphism. Also denote by \mathcal{U} the group of units of \mathcal{O} .*

We extend the definition of the valuation to a matrix $A = (a_{ij}) \in \mathcal{M}_n(F)$ by $v(A) = \min(v(a_{ij}))$, the minimum of the valuations of each individual entry of A . Therefore, every nonzero matrix with entries in F can also be uniquely expanded and written in the form

$$A = \tau^{v(A)} \sum_{i=0}^{\infty} A_i \tau^i, \quad A_0 \neq 0 \quad (2.36)$$

where the coefficient matrices A_i are matrices with entries in K .

Furthermore, we make the additional assumption $v(\phi(f)) = v(f)$ for all $f \in F$. From this, we obtain

$$\begin{aligned} \phi(\tau) &= q\tau + O(\tau^2), \\ \tau^{-\omega}\delta(\tau) &= c\tau + O(\tau^2) \end{aligned}$$

with $q, c \in K^*$ and inductively for $h \in \mathbb{N}^+$

$$\phi(\tau^h) = q^h \tau^h + O(\tau^{h+1}), \quad (2.37)$$

$$\tau^{-\omega}\delta(\tau^h) = c[h]_q \tau^h + O(\tau^{h+1}) \quad (2.38)$$

where $\omega \in \mathbb{Z}$.

Definition 2.5.3. *The integer ω defined by $\omega(\delta) = v(\delta(\tau)) - v(\tau)$ is called the degree of δ .*

We can extend the definition of v , lc and ω to the field $K((\tau^{1/s}))$ for $s \in \mathbb{N}^*$ in a straightforward manner by considering the series expansion of elements of F with fractional exponents. In this case, the image of the valuation map v is $\mathbb{Q} \cup \{\infty\}$. The fields $(K((\tau)), \phi, \delta)$ and $(K((\tau^{1/s})), \phi, \delta)$ are examples of what we could call a *local ϕ, δ field*.

The quantities ω , q and c will be extremely important throughout the remainder of this thesis. They will allow expressing, in a unified manner, the subtle differences in the structure of solutions and behaviour of our algorithms for the individual types of systems.

2.6 Singular Systems of Pseudo-Linear Equations

We will now introduce systems of pseudo-linear equations with coefficients in $K((\tau))$. We will discuss the algebraic aspect of a singularity of the system and introduce a classification of singularities.

The idea of a singularity of linear functional systems such as, for example, a linear differential system, is usually motivated by analytical properties of solutions – usually, over the complex numbers. For our work, we seek a purely algebraic characterisation expressed by the structure of both the system and its solutions. Ultimately, these structures are related, but it is easier to focus on the form of the system first. Gradually, after carrying out certain transformations as described in Chapter 3, 4 and 5, we will be able to tackle the actual computation of solutions.

If we assume that the origin is a singularity of our system, we are led to study systems as defined below.

Definition 2.6.1. *A singular system of pseudo-linear equations is of the form:*

$$\tau^{r-\omega}\delta Y(\tau) = A(\tau)\phi Y(\tau) \quad (2.39)$$

with $r \in \mathbb{N}$ the Poincaré-rank of the system, and $A \in \mathcal{O}^{n \times n}$ ($A_0 \neq 0$).

Consider the system (2.39). The change of variable $Y = TZ$ where $T \in \text{GL}(n, F)$ leads to an *equivalent* system

$$\tau^{r-\omega}\delta Z = B\phi Z \quad (2.40)$$

where $B \in \mathcal{M}_n(F)$. If we can solve this system, then we can also compute the solutions of the original systems as they are related by multiplication with the invertible matrix T . The formula for the matrix B is given in the definition below.

Definition 2.6.2. *The change of variable (2.40) inducing the new system matrix*

$$B = T^{-1}(A\phi T - \tau^{r-\omega}\delta T) =: T_{\delta,\phi}[A] \quad (2.41)$$

where $T \in \text{GL}(n, F)$ is referred to as a Gauge transformation.

We will need this to define a classification of singularities in the next definition, but also from an algorithmic point of view for the reduction algorithms in the subsequent chapters.

Definition 2.6.3 (Classification of Singularities). *We distinguish between regular and irregular singularities. A system of the first kind has $r = 0$ and in this case, the singularity is regular. In general, if a system with $r > 0$ is equivalent to a system of the first kind, the singularity is also regular. Otherwise, it is an irregular singularity.*

A singular system of the form (2.39) can be obtained e.g. by localising (2.31) at an appropriate point $x_0 \in \bar{K} \cup \{\infty\}$. This means “switching” from $K(x)$ to $\bar{K}((x))$ by means of the map $x \rightarrow \tau + x_0$ for $x_0 \neq \infty$ or $x \rightarrow \tau^{-1}$ otherwise.

Example 2.6.1. *Consider the previous example and assume*

$$B(x) = \begin{pmatrix} x - 1 & 2x \\ 1 & -1 \end{pmatrix}.$$

We want to study the point $x_0 = \infty$. We put $\tau = x^{-1}$, $\phi(\tau) = (x - 1)^{-1} = \frac{\tau}{1 - \tau}$, and $\delta = id - \phi$. Hence, $\omega = 1$ and the system can be written as a singular system of the form

$$\tau \delta Y(\tau) = \begin{pmatrix} 1 - \tau & 2 \\ \tau & -\tau \end{pmatrix} \phi Y(\tau).$$

We can see that we have $r = 2$. In this case, it is not clear whether the system is regular or irregular singular.

2.7 Computing Formal Solutions

In this thesis, we will develop a framework for expressing and computing formal solutions of the system (2.39). Formal solutions are effectively asymptotic expansions of real solutions which have been studied extensively, for example [57] in the differential case. To our knowledge, a unifying view has not been pursued in the literature. Typically, the differential case was the initial point of investigation and authors subsequently extended their understanding to the difference and q -difference case. We will not explicitly state the structure of the formal solutions yet, the reader will find the information in Chapter 5, Section 5.5 – we will have to introduce additional terminology for that purpose. Instead, we give an account of previous theoretical work on formal solutions.

The theory of formal solutions of linear functional systems first created interest amongst mathematicians in the first half of the twentieth century. Various authors obtained theoretical results that ensured the existence of such solutions. Amongst these works, Birkhoff [29] was the first to give a canonical form of linear differential systems of the form

$$x^{r+1}Y'(x) = A(x)Y(x)$$

where $r \in \mathbb{N}$ is the so-called *Poincaré-rank* of the system and A a $n \times n$ matrix with formal Laurent series coefficients. He then went on [30] to give the structure of formal solutions around an irregular singular point in the difference case

$$x^{r+1}Y(x+1) = A(x)Y(x)$$

and q -difference case

$$x^rY(qx) = A(x)Y(x).$$

In the difference case, A is considered to be a formal Laurent series matrix in x^{-1} . Additional work by himself [28] and other authors such as Adams [7] and Trjitzinsky [68] continued the theoretical investigations where frequently, first order systems or single linear n th-order equations were employed interchangeably. These results however were unsatisfactory from a computational point of view.

Later on, research focussed on constructive proofs for the existence of a formal fundamental matrix solution by carrying out various transformations directly on the input system. This is commonly referred to as *formal reduction*. Turrittin was the first to consider the formal reduction of linear differential and difference systems [69, 70]. He showed that during this process, the system can be split into one or more systems of smaller dimension provided the leading term A_0 of the coefficient matrix has several distinct eigenvalues. His approach was later made more algorithmic by Wasow [74] where concepts such as the Splitting Lemma are introduced. For q -difference systems, the formal reduction is alleviated by the fact that the structure of the formal fundamental matrix is much simpler. This was announced by Birkhoff in [31] although a detailed description of the formal reduction algorithm was not provided therein and, to our knowledge, has not been reported elsewhere.

From our point of view, we are interested in a unifying view which makes the consideration of analytical aspects very difficult. We will obtain the formal solutions through symbolic manipulation of the input system, using various transformations, based on purely algebraic considerations. The reduction algorithms that we will give in Chapter 3 and the formal reduction algorithm that we will devise in Chapter 5 are generic in the sense that they work with any valid choices of ϕ and δ such that (F, ϕ, δ) is a local ϕ, δ -field. However, in order to compute formal solutions, we need explicit formulae for regular parts and resolving irregular parts (this will be introduced in Chapter 4 and in Chapter 5). There are three classes of linear functional systems for which we have symbolic expressions for these. For each of these classes, we have chosen representative systems in such a way that they help stating the generic algorithm with as little technical effort as possible. These representative systems are:

- (i) Linear differential systems, where we have $\phi = \text{id}$ and $\delta = \frac{d}{d\tau}$. In this case, one finds $\omega = -1$ and $q = 1$.
- (ii) Linear difference systems which are represented by choosing as an automorphism ϕ the Puiseux-series expansion of $\frac{\tau}{\sqrt{\tau^s+1}}$ and $\delta = \text{id} - \phi$. One finds $\omega = s$ and $q = 1$ with $s \in \mathbb{N}^*$.
- (iii) Linear q -difference systems, which are modeled using $\phi(\tau) = q\tau$ and $\delta = \text{id} - \phi$. We have $\omega = 0$ and $q \neq 0$ where q is not a root of unity.

Our algorithm is able to compute formal solutions of any linear functional system that can be transformed into one of the three types above. This is indeed possible for large classes of linear functional systems [38].

Remark 2.7.1. *The choice of automorphism ϕ in (ii) allows restricting the field of our input system to that of formal Laurent (rather than Puiseux) series. This is motivated by the fact that our algorithm avoids manipulating series with fractional exponents by using substitutions of the form $\tau \rightarrow \tau^s$ are used.*

2.8 Review of Computer Algebra Techniques

In this thesis, we are concerned with symbolic algorithms for systems of pseudo-linear equations in contrast to scalar n th order equations. For the latter, Computer Algebra has already developed a number of methods, such as algorithms for linear differential [10, 71, 72, 73, 33, 32, 66, 61] and difference equations [2, 5, 34, 6, 43] of order n where $n \in \mathbb{N}^*$. More general linear functional equations that comprise these particular types of equations have been investigated by several authors [37, 34, 51] and most of the important research questions such as computing polynomial, rational or power series solutions are well understood. This is due to the fact that essential local information at the equation's singularities, needed for the various types of algorithms, is easily accessible from the coefficients of the equation based on Newton-polygon constructions [72, 66, 43].

In the case of first-order linear differential and difference systems and, more generally, linear functional systems, the situation is more difficult and the state of the art of efficient methods for computing the same classes of solutions as in the scalar case is much less well developed. The main reason for this is, as far as we are aware, a lack of effective methods for directly computing required local information in the neighbourhood of irregular singularities. In principle, systems can be converted to scalar n th order equations via a cyclic vector process [41] or related methods [13]. This is useful from a theoretical point of view, but does not lead to practical algorithms [62, 42].

The crucial work that eventually led to an efficient local treatment of systems was initiated by [14] who showed that the use of the so-called Moser-reduction algorithm [58, 50, 45] is an essential building block for Turrittin's formal reduction in the differential case. The algorithm for computing Moser-reductions was later improved in [23, 27] and is the starting point for our generalisation to systems of pseudo-linear equations. Moser-reduction of difference systems was also considered by [11] and [12] where it is shown how to extend Moser's reduction principle to this case. In a series of papers [17, 18] and [39], the authors develop variations of the Moser-reduction in order to determine the structure of solutions. Linear systems of q -difference equations had not been considered for the class of Moser-reduction algorithms prior to our paper [25] for the general pseudo-linear case. The Chapter 3 contains the results of this work.

In [14], for linear differential systems, the Moser-reduction is then used for computing the exponential part of formal solutions. Furthermore, a method for minimising algebraic extensions of the constant field, as well as computing the Katz invariant of the system was given in the same paper, however it did not result in a method for computing with minimal ramifications. Based on this approach, [17, 18, 39] are able to treat linear difference systems. The paper [63] improves [14] by keeping ramifications minimal. This is achieved through refined block-decompositions, using a Generalised Splitting Lemma. This allows for a more efficient implementation of the formal reduction procedure. A specialised algorithm for computing the smaller class of regular formal solutions was given in [21]. In this thesis, we extend these results to the general pseudo-linear case.

The direct symbolic treatment of linear systems of q -difference equations however does not seem to have been addressed in an algorithmically satisfactory manner. Modern work has so far been focussing on the regular singular case [65], but no general method for computing a full formal fundamental matrix seemed available prior to our work.

Chapter 3

Super-Irreducible Forms of Systems of Pseudo-Linear Equations

In this chapter, we deal with our first research objective: the design of a reduction algorithm for the classification of singularities of systems of pseudo-linear equations. We define a suitable reduction principle based on Moser's definition in the differential case and generalise this principle analogously to the concept of super-irreducible forms. We then find a constructive reduction criterion, computed from a finite number of coefficients, to decide whether a given system of pseudo-linear equations is reducible. Finally, an algorithm to reduce an arbitrary system of pseudo-linear equations into irreducible form is developed. In the case of a regular singularity, this will compute a transformation which brings the given system into an equivalent system which is of the first kind.

3.1 Introduction

In this chapter, we consider the local pseudo-linear system (2.39) as introduced in the previous chapter. We remind the reader that this is a system of the form

$$\tau^{\omega-\tau} \delta Y(\tau) = A(\tau) \phi Y(\tau)$$

where A is a square matrix of dimension n with coefficients in F . As we will see in the subsequent chapters, it will turn out to be useful to tackle the following problem: given (2.39), decide whether the system is irregular or regular singular at a given point and, in the latter case, compute a gauge transformation $T_{\delta,\phi}$ which takes the system into an equivalent system $T_{\delta,\phi}[A]$ which is of the first kind. Giving an algorithm that does that is the subject of this chapter.

3.2 Moser-Irreducible Forms

We shall develop an algorithm to reduce a given system (2.31) to an equivalent *irreducible* system with system matrix B having maximal valuation:

$$v(B) = \max \{v(P[A]): P \in \text{GL}(n, F)\}.$$

Furthermore, the algebraic rank of its leading matrix is minimal amongst all systems with maximal valuation:

$$\text{rank}(\text{lc}(B)) = \min \{\text{rank}(\text{lc}(\tilde{B})) : \tilde{B} \text{ has maximal valuation}\}.$$

The method followed is analogous to the one used in the differential case [19, 63, 50], based on Moser's work [58].

We associate with the system the following rational numbers:

$$m_{\delta, \phi}(A) = \begin{cases} \omega - v(A) + \frac{\text{rank}(A_0)}{n} & \text{if } v(A) < \omega, \\ 0 & \text{if } v(A) \geq \omega \end{cases}$$

and

$$\mu_{\delta, \phi}(A) = \min \{m_{\delta, \phi}(T_{\delta, \phi}[A]) \mid T \in \text{GL}(n, F)\}.$$

Definition 3.2.1. *The matrix A is called irreducible w.r.t. δ and ϕ if $m_{\delta, \phi}(A) = \mu_{\delta, \phi}(A)$, otherwise it is called reducible.*

We remark that the system is regular if and only if $\mu_{\delta, \phi}(A) = 0$. The following result is the analogue of Theorem 1 in [58] which gives a reducibility criterion in the differential case.

Theorem 3.2.1. *Consider the system (2.39) with Poincaré-rank $r > -\omega$ and let $n_0 = \text{rank}(A_0)$. Then the system is reducible if and only if the polynomial*

$$\theta(\lambda) := \pi(\tau^{n_0} \det(\tau^{r-1}A - \lambda I))$$

vanishes identically in λ .

Remark 3.2.1. *It is easy to see that the polynomial $\theta(\lambda)$ depends only on A_0 and A_1 :*

$$\theta(\lambda) = \pi(\tau^{n_0} \det(\tau^{-1}A_0 + A_1 - \lambda I)).$$

This theorem can be proven in a quite similar way to that used for Theorem 1 in [58]. We first point out some useful facts which may help the reader to understand why the approach of Moser can be adapted to our situation. The first ingredient used in the proof by Moser

(see Lemma 1 of [58]), namely the property that any gauge transformation T can be written in Smith normal form, remains valid in our situation.

Lemma 3.2.1. Any matrix $T \in \mathrm{GL}(n, F)$ can be written in Smith normal form: $T = P\tau^\alpha Q$ where $P, Q \in \mathcal{M}_n(\mathcal{O})$ with $\det P, \det Q \in \mathcal{U}$,

$$\tau^\alpha = \mathrm{diag}(\tau^{\alpha_1}, \dots, \tau^{\alpha_n})$$

where α_i in \mathbb{Z} with $\alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_n$.

Proof This result follows from the fact that \mathcal{O} is a principal ideal domain (the ideals of \mathcal{O} are of the form $\tau^m \mathcal{O}$). \square

In the sequel we define the *span* of a transformation $T \in \mathrm{GL}(n, F)$ as the nonnegative integer $\sigma(T) := -v(T) - v(T^{-1}) = \alpha_n - \alpha_1$.

The second ingredient used by Moser (see Lemma 2, p 387 in [58]) can be stated using our notation as:

Lemma 3.2.2. Let $A \in \mathcal{M}_n(\mathcal{O})$ of the form

$$A = A_0 + A_1\tau + \dots \quad (A_0 \neq 0). \quad (3.1)$$

A necessary and sufficient condition that there exists a matrix $T \in \mathrm{GL}(n, F)$ such that the matrix

$$B := T^{-1}AT = B_0 + B_1\tau + \dots$$

belongs to $\mathcal{M}_n(\mathcal{O})$ and satisfies

$$\mathrm{rank}(B_0) < n_0 := \mathrm{rank}(A_0)$$

is that the polynomial

$$\theta(\lambda) := \pi(\tau^{n_0} \det(\tau^{-1}A_0 + A_1 - \lambda I))$$

vanishes identically in λ . Moreover T can be chosen with $\sigma(T) = 1$.

The proof of this lemma can be found in Moser's paper [58]. An important insight is that it remains valid if $T^{-1}AT$ is replaced by $T^{-1}A\phi T$. This can be seen as follows: suppose that $\phi \neq \mathrm{id}$ and write $\phi\tau = q\tau + O(\tau^2)$ with $0 \neq q \in K$. Then one can see that

$$T^{-1}\phi T = q^\alpha + O(\tau)$$

and hence

$$T^{-1}A\phi T = (T^{-1}AT)T^{-1}\phi T = B_0q^\alpha + O(\tau).$$

Since $\text{rank}(B_0 q^\alpha) = \text{rank}(B_0)$, our claim follows.

Finally, the key for establishing Theorem 3.2.1 is (as in the proof by Moser) to remark that, using a transformation T as above, the reduction of the rank of A_0 is not affected by the term $T^{-1}\delta T$ in $T_{\delta,\phi}[A]$.

3.3 The Moser-Reduction Algorithm

In this section, we give a version of the Moser-reduction algorithm for systems of pseudo-linear equations. When specialising this algorithm to the differential case, it corresponds to an improved version of the Moser-reduction algorithm as introduced in [27]. Our approach is to replace the differential equivalence by the Gauge transformation (2.41) and to show that this results in a valid reduction algorithm. The main task is to prove that Lemma 2.3 in [23] remains valid.

Before we outline the method for the differential case, we remark that if A_0 is not nilpotent, we have rank of $A_0 = n$ and

$$\theta(\lambda) := \pi \left(\tau^n \det \left(\tau^{-1} A_0 + A_1 - \lambda I \right) \right) = \pi \left(\det \left(A_0 + A_1 \tau - \lambda \tau I \right) \right) = \det(A_0) \neq 0$$

hence the system is irreducible. The interesting case, where the system is reducible, requires A_0 to be nilpotent.

Now assume A_0 to be in Jordan canonical form $A_0 = \text{diag}(J, 0_s)$ where J has d Jordan blocks of dimension $n_i \geq 2$ with $n_1 \geq \dots \geq n_d > n_{d+1} = \dots = n_d = 1$. Define l_i (c_i respectively) for $i = 1, \dots, d + s$, as the position of the i th zero row (column respectively) of A_0 .

The L -matrix $L(A, \lambda) \in k[\lambda]^{(d+s) \times (d+s)}$ is then defined by

$$L(A, \lambda) = ((\alpha_{i,j})) - \text{diag}(0_d, \lambda I_s)$$

where $((\alpha_{i,j})) \in k^{(d+s) \times (d+s)}$ is defined by setting $\alpha_{i,j}$ as the entry of A_1 of row l_i and column c_j .

Example 3.3.1. *The following example, taken from [23] illustrates this definition for $d = 2$,*

$n_1 = 3, n_2 = 2, n_3 = n_4 = 1$ and $s = 2$: we have

$$A_0 = \left(\begin{array}{cc|cc} 0 & 1 & & \\ & 0 & 1 & \\ & & 0 & \\ \hline & & 0 & 1 \\ & & & 0 \\ \hline & & & 0 \\ \hline & & & 0 \end{array} \right).$$

The L -Matrix is then a 4×4 matrix

$$L(A, \lambda) = \begin{pmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times - \lambda & \times \\ \times & \times & \times & \times - \lambda \end{pmatrix}$$

whose entries are copied from the entries of A_1 which are marked by “ \times ”:

$$A_1 = \left(\begin{array}{cccc} & & & \\ \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \end{array} \right).$$

In [23], the following was shown:

Proposition 3.3.1 ([23], Proposition 2.1). *The system (2.39) is Moser-reducible if and only if $\det L(A, \lambda) \equiv 0$.*

At each reduction step, this algorithm uses the following normalisation: construct a constant transformation C such that $C[A] = \tilde{A}$ has a corresponding L -matrix with the following structure:

$$\begin{pmatrix} L^{11} & L^{12} & 0 \\ L^{21} & L^{22} - \lambda & 0 \\ L^{31} & L^{32} & L^{33} - \lambda \end{pmatrix} \quad (3.2)$$

where L^{11} , L^{22} and L^{33} are square matrices of dimension d , $s - q$ and q respectively, $0 \leq q \leq s$

with the additional condition that:

$$\text{rank} \begin{pmatrix} L^{11} \\ L^{21} \end{pmatrix} + s - q = \text{rank} \begin{pmatrix} L^{11} & L^{12} \\ L^{21} & L^{22} \end{pmatrix} \quad (3.3)$$

and L_{33} is upper-triangular with a zero diagonal.

The diagonal transformation which carries out the reduction step is then of the form [23]

$$S = \text{diag}(I_{n_1-1}, x, \dots, I_{n_d-1}, x, xI_{s-q}, I_q). \quad (3.4)$$

3.3.1 A Generalisation of the Moser-Reduction Algorithm

Now assume that $\phi \neq \text{id}$ and $\delta \neq 0$, both fixed. Recall that δ is of the form $\delta = \gamma(\text{id} - \phi)$ for some nonzero element $\gamma \in F$. In order to simplify the proofs in this section, we use the fact that we can always assume that $\omega(\delta) = 0$. Indeed, it suffices to replace δ by $\tilde{\delta} := x^{-\omega(\delta)}\delta$.

We write $\phi x = qx + O(x^2)$ for some nonzero element $q \in K$ and we obtain for $k \neq 0$

$$\phi(x^k) = q^k x^k + O(x^{k+1}). \quad (3.5)$$

We consider a pseudo-linear system of the form (2.31) and suppose that $r = -v(A) > 0 = -\omega(\delta)$. Our assumption $\omega(\delta) = 0$ implies that $m_{\delta,\phi}(A) = m_{\delta,\text{id}}(A)$. We shall refer to this quantity simply by $m(A)$. Notice that in this case, the definition of the Moser-polynomial $\theta(\lambda)$ associated with a singular pseudo-linear system is independent from the particular choice of δ and ϕ . Therefore we have the following property: the matrix A is reducible w.r.t. (δ, ϕ) if and only if it is w.r.t. (δ, id) .

Suppose now that A is reducible and let T be a transformation such that $m(T_{\delta,\text{id}}[A]) < m(A)$. The natural question whether $m(T_{\delta,\phi}[A]) < m(A)$ arises. In general, this is clearly not the case. However, it does hold for transformations T of the form $T = CS$ with C constant and $S = \text{diag}(x^{d_1}, \dots, x^{d_n})$ such that $\sigma(S) = \max d_i - \min d_i \leq 1$. This is precisely, as we have seen in the previous section, the type of transformation used in each reduction step in the algorithm for the Moser-reduction in the differential case.

We shall now prove this result. The constant transformation C which is used for the normalisation of the system can be used similarly as in the differential case since

$$C_{\phi,\delta}[A] = C^{-1}A\phi C - C^{-1}\delta C = C^{-1}AC = C_{\delta,\text{id}}[A]$$

for any constant transformation C , due to the fact that ϕ is a K -automorphism. We will show that for the diagonal transformation S we have $m(S_{\phi,\delta}[A]) < m(A)$.

Lemma 3.3.1. *The rank-reduction in the differential algorithm is achieved by merely using the similarity transformation $S^{-1}AS$.*

Proof Using the fact that S is a diagonal transformation

$$S = \text{diag}(x^{d_1}, \dots, x^{d_n}) =: x^D,$$

where $D = \text{diag}(d_1, \dots, d_n)$ with $d_i \in \mathbb{N}$, we compute $S^{-1} = x^{-D}$ and $\delta S = Dx^D$, hence

$$B = S[A] = S^{-1}AS - D.$$

We can see that the matrix D does not affect the leading matrix B_0 of the transformed system since $r > 0$. Hence the reduction in rank will have to be achieved by $S^{-1}AS$ alone.

□

The following technical lemma studies the effect of the automorphism ϕ on the diagonal transformation S .

Lemma 3.3.2. *Let S and D be as in Lemma 3.3.1 and its proof, with $d_i \in \{0, 1\}$ ($i = 1 \dots n$). Then*

$$\phi S = Sq^D + O(x^2).$$

Proof Using (3.5), we obtain

$$\phi(x^{d_i}) = \begin{cases} 1 & \text{if } d_i = 0 \\ q^{d_i}x^{d_i} + O(x^{d_i+1}) & \text{otherwise} \end{cases} \quad (3.6)$$

and, using $d_i \in \{0, 1\}$ we compute

$$\begin{aligned} \phi S &= \text{diag}(q^{d_1}x^{d_1}, \dots, q^{d_n}x^{d_n}) + O(x^2) \\ &= Sq^D + O(x^2). \end{aligned}$$

□

We now establish that rank reduction can be done by merely involving the first term of a pseudo-linear transformation. In particular, this does not require δ .

Lemma 3.3.3. *The operation $S^{-1}A\phi S$ with S as in the previous lemma achieves a rank-reduction.*

Proof Using Lemma 3.3.2, we find

$$\begin{aligned} S^{-1}A\phi S &= S^{-1}A((Sq^D + O(x^2))) \\ &= S^{-1}ASq^D + S^{-1}A \cdot O(x^2). \end{aligned} \quad (3.7)$$

Using Lemma 3.3.1, we know that $S^{-1}AS$ has a leading matrix with reduced rank, hence this is also true for

$$S^{-1}ASq^D =: B_0x^{-r} + O(x^{-r+1}).$$

But it is clear that the second term in the last sum in (3.7) is

$$S^{-1}A \cdot O(x^2) = O(x^{-r+1})$$

since $v(S^{-1}) = -1$, implying that this term does not affect B_0 . □

The following result shows that any diagonal transformation as in the proof of Lemma (3.3.1) with $d_i \in \{0, 1\}$ that leads to a rank reduction in the differential case, also achieves this in the pseudo-linear setting.

Proposition 3.3.2. *Suppose that A is Moser-reducible and the diagonal transformation $S = \text{diag}(x^{d_1}, \dots, x^{d_n})$ with $d_i \in \{0, 1\}$ satisfies $m(S[A]) < m(A)$. Then we also have $m(S_{\phi, \delta}[A]) < m(A)$.*

Proof We have

$$\begin{aligned} S_{\phi, \delta}[A] &= S^{-1}A\phi(S) - S^{-1}\delta(S) \\ &= S^{-1}A\phi(S) - \tilde{C}x^\omega + O(x^{1+\omega}) \end{aligned}$$

where \tilde{C} is a constant diagonal matrix. Using Lemma 3.3.3, we have

$$S^{-1}A\phi S = B = x^{-r}(B_0 + O(x)),$$

with $\text{rank} B_0 < \text{rank} A_0$. But we have assumed $\omega = 0 > -r$ hence, the proposition follows. □

The reduction algorithm is then as follows:

Pseudo_Linear_Moser_Reduction(A, ϕ, δ)

Input: $A \in \mathcal{M}_n(F)$, ϕ a K -automorphism, δ a Pseudo-derivative w.r.t. ϕ

Output: Transformation T such that $T_{\phi, \delta}[A]$ is Moser (δ, ϕ) -irreducible

1. $T := I_n$;
2. **while** $(r(A) > -\omega)$ **and** $(\theta(\lambda) \equiv 0)$ **do**
 - (a) Compute a constant transformation C to normalise A ;
 - (b) $A := C_{\phi, \delta}[A]$; $T := TC$;
 - (c) Compute a diagonal transformation S as in the differential case;

(d) $A := S_{\phi, \delta}[A]$; $T := TS$;

3. return T ;

3.4 Super-Irreducible Forms

In [50], Hilali and Wazner introduced the concept of *super-irreducible* forms, which can be seen as a generalisation of the Moser-irreducible form. In this section we extend this concept to pseudo-linear systems.

Consider again a pseudo-linear system of the form (2.31) and suppose that $r = -v(A) > -\omega(\delta)$. Mimicking the differential case we define, for $1 \leq k \leq r + \omega$, the rational number $m_{\delta, \phi}^k(A)$ by

$$m_{\delta, \phi}^k(A) = r + \omega + \frac{n_0}{n} + \frac{n_1}{n^2} + \cdots + \frac{n_{k-1}}{n^k}$$

where $n_i = n_i(A)$ is the number of rows of A with valuation $v(A) + i$.

Now define

$$\mu_{\delta, \phi}^k(A) = \min\{m_{\delta, \phi}^k(T_{\delta, \phi}[A]) \mid T \in \text{GL}(n, F)\}.$$

Definition 3.4.1. *The matrix A or system (2.31) is said to be k -irreducible (w.r.t. (δ, ϕ)) if $m_{\delta, \phi}^k(A) = \mu_{\delta, \phi}^k(A)$. Otherwise A is called k -reducible. The matrix A or system (2.31) is said to be super-irreducible, if it is k -irreducible for every k , or equivalently if $m_{\delta, \phi}^{r+\omega}(A) = \mu_{\delta, \phi}^{r+\omega}(A)$.*

A criterion for k -reducibility is obtained in exactly the same way as in the differential case. One defines s_k and $\Theta_k(\lambda)$ as where

$$s_k = s_k(A) = kn_0 + (k-1)n_1 + \cdots + n_{k-1}.$$

$$\Theta_k(\lambda) := t^{s_k} \det(t^{r-k}A - \lambda I_n)$$

and verifies that $\Theta_k(\lambda)$ belongs to $\mathcal{O}[\lambda]$. Then one can define the polynomial $\theta_k(\lambda) \in K[\lambda]$ as

$$\theta_k(\lambda) = \pi(t^{s_k} \det(t^{r-k}A - \lambda I_n)) \quad (3.8)$$

In the same way as in the differential case, one can show that one has the following

Theorem 3.4.1. *The matrix A is k -irreducible, if and only if the polynomials $\theta_j(\lambda)$, ($j = 1, \dots, k$), do not vanish identically in λ .*

3.4.1 The Second Reduction Algorithm

In [23] it was shown that the computation of a super-irreducible system can be reduced to the computation of several Moser-irreducible systems of smaller size, using a block-reduction

algorithm.

In this section, we assume that the first reduction algorithm has been applied to the system (2.31). We will show that the *Direct Block Reduction Algorithm* as introduced in [27] can be used for systems of linear functional equations in order to obtain a second reduction algorithm which can be characterised as computing a new system of the form $\delta Y = B\phi Y$ where B is a block-triangular super-reduced matrix where each diagonal block is Moser-reduced. As explained in [27], Section 3, the block-reduction is achieved by using *Elementary Operations* of the form

$$E_{i,j}(\alpha) = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & & \ddots \\ & & & & & & 1 \end{pmatrix}$$

where the entry at position (i, j) is $\alpha \in K[[x]]$. We recall that transforming a given differential system with $E_{i,j}(\alpha)$ results in a new system whose coefficient matrix \tilde{A} is obtained from A by adding to the j th column the i th column multiplied by α , then subtracting the j th row multiplied by α from the i th row, and adding $\delta(\alpha)$ to the entry in the (i, j) position.

The effect of using $E_{i,j}(\alpha)$ as a pseudo-linear transformation is very similar to the differential case, with the difference that the j th column of the transformed systems results from adding to the j th column the i th column multiplied by $\phi(\alpha)$ and also adding $\delta(\alpha)$ to the entry in the (i, j) position.

The concept of *normalised Moser-irreducible forms* as introduced in [27] can be easily extended to the case of systems of linear functional equations as the normalisation is carried out using a constant transformation.

Proposition 3.4.1. *The direct Block-Reduction algorithm of [27] can be adapted for our use by replacing the elementary transformation $E_{ij}(\alpha)$ with $E_{ij}(q^{-h}\alpha)$ where $h = v(\alpha)$.*

Proof By reviewing the process of eliminating terms in A it becomes apparent that the elimination is achieved using linear combinations of leading coefficients of elements in A , multiplied by the leading coefficients of α . Let $\alpha = ct^h + \dots$, it follows

$$\phi(\alpha) = q^h ct^h + \dots,$$

hence the leading coefficient of $\phi(q^{-h}\alpha)$ equals c . Hence using $E_{ij}(q^{-h}\alpha)$ for a pseudo-linear transformation carries out an identical elimination process on the corresponding leading terms. □

Remark 3.4.1. *This does not imply that the final result of the block-reduction is identical in both of the differential and general pseudo-linear case. Crucially, each isolated step of the algorithm does indeed an identical elimination in both cases, but the transformations introduces also additional terms of higher order that are different.*

3.4.2 A First Application: Polynomial Solutions of Linear q -Difference Systems

Let $q \in K$ with $q \neq 1$ and consider a linear q -difference system with coefficients in $K(x)$:

$$Y(qx) = M(x)Y(x), \quad M(x) \in \mathcal{M}_n(K(x)). \quad (3.9)$$

We are interested in this section by the problem of computing all the polynomial solutions of a linear q -difference system. Algorithms for solving this problem in the differential and the difference cases have already been proposed in the past [15, 4]. We shall show that a similar approach as therein remains valid for the q -difference case, and that the necessary ingredient for the algorithm is the ability to compute super-irreducible systems. Using the algorithm from the previous section, this can be done efficiently. For sake of brevity we shall consider here only the problem of computing a bound on the degree of polynomial solutions. Such a bound can be obtained from the so-called *indicial equation* (at $x = \infty$). Unfortunately the indicial equation is not immediately apparent for a given general system. The idea consists in reducing the given system to a *simple* from which the *indicial equation* can be immediately obtained. We will show that such a simple form can be derived from a super-irreducible form in exactly the same way as in the difference and the differential cases.

3.4.3 The Indicial Equation

A polynomial solution $Y \in K[x]^n$ of degree ν can be viewed as local formal solution (at $x = \infty$) of the form

$$Y(x) = \sum_{i \geq 0} x^{-i+\nu} Y_i, \quad (3.10)$$

where $Y_i \in K^n$, $Y_0 \neq 0$ and $Y_i = 0$ for $i > \nu$.

The idea is to work with $F = K((x^{-1}))$, the completion of $K(x)$ w.r.t. to the t -adic valuation (here $t = x^{-1}$). Define ϕ and δ by $\phi(t) = qt$ and $\delta = \text{id} - \phi$. A linear q -difference system can then be written as a pseudo-linear system

$$\delta Y = A(t)\phi Y \quad (3.11)$$

where

$$A(t) = M(q^{-1}t^{-1}) - I_n \in \mathcal{M}_n(F).$$

Multiplying this system on the left by the diagonal matrix

$$D := \text{diag}(t^{\alpha_1}, \dots, t^{\alpha_n}),$$

where $\alpha_i = -\min(v(A_{i,\cdot}), 0)$, $A_{i,\cdot}$ being the i th row of the matrix A , yields the equation

$$D\delta Y = C\phi Y, \tag{3.12}$$

where $C = DA$. By definition, one has $D, C \in \mathcal{M}_n(K[[t]])$. Put

$$C = \sum_{i \geq 0} C_i t^i, \quad D = \sum_{i \geq 0} D_i t^i.$$

We look for formal solutions of the form:

$$Y = \sum_{i=0}^{+\infty} t^{i+\nu} Y_i \quad \nu \in K, Y_i \in K^n, Y_0 \neq 0.$$

One has

$$\phi Y = \sum_{i=0}^{+\infty} q^{i+\nu} t^{i+\nu} Y_i$$

and

$$\delta Y = Y - \phi Y = \sum_{i=0}^{+\infty} (1 - q^{i+\nu}) t^{i+\nu} Y_i.$$

Replacing $D, C, \phi Y$ and δY by their t -adic expansions in (3.12) and identifying coefficients of the powers t^ν yields, in particular, the equation

$$(1 - q^\nu) D_0 Y_0 = q^\nu C_0 Y_0.$$

Thus in order that the system (3.12) admits a formal solution of the form (3.10), ν and Y_0 must satisfy the equation

$$(C_0 - (q^{-\nu} - 1) D_0) Y_0 = 0$$

which implies that $(q^{-\nu} - 1)$ must be a root of the polynomial

$$E(\lambda) := \det(C_0 - \lambda D_0).$$

As a consequence:

- If $Y \in K[x]^n$ is a nonzero polynomial solution of the system of degree ν , then $E(q^{-\nu} -$

1) = 0.

- The degree of polynomial solution can be bounded by the biggest nonnegative integer ν such that $q^{-\nu} - 1$ is a root of $E(\lambda)$.

However, it may happen that the determinant $E(\lambda)$ vanishes identically in λ in which case it is quite useless to us. This motivates the following definition:

Definition 3.4.2. *The system (3.12) is said to be simple if $\det(C_0 - \lambda D_0) \neq 0$ (as a polynomial in λ). In this case the polynomial $E(\lambda) := \det(C_0 - \lambda D_0)$ is called the indicial polynomial of (3.12).*

As an example of simple systems, take a system of the form (3.11) with $v(A) \geq 0$. In this case $D = I_n$ and $C = A$. Hence $E(\lambda) = \det(A_0 - \lambda I_n) \neq 0$. Consequently, the system is simple and its indicial polynomial has degree n .

Proposition 3.4.2. *Every q -difference system (3.11) can be reduced to an equivalent system (3.12) which is simple.*

Proof Since every q -difference system (3.11) is equivalent to a super-irreducible one, it suffices to prove that every super-irreducible system is simple.

Consider a system of the form (3.11) and put $r = -v(A)$. If $r \leq 0$ then the system is simple. Suppose that $r > 0$ (notice that $\omega(\delta) = 0$) and let D and $C = DA$ be defined as above then

$$E(\lambda) = \det(C_0 - \lambda D_0) = \theta_r(\lambda).$$

Indeed, one easily verifies that $\det(D) = t^{s_r}$ (see Section 3.4 for the definition of s_r and θ_r) hence

$$\begin{aligned} t^{s_r} \det(A - \lambda I_n) &= \det D \det(A - \lambda I_n) \\ &= \det(DA - \lambda D). \end{aligned}$$

Hence

$$\begin{aligned} \theta_r(\lambda) &= t^{s_r} \det(A(t) - \lambda I_n)|_{t=0} \\ &= \det(C(t) - \lambda D(t))|_{t=0} \\ &= \det(C_0 - \lambda D_0). \end{aligned}$$

Now if (3.11) is super-irreducible then, by Theorem 3.4.1, the polynomial $\theta_r(\lambda)$ is not identically zero and (3.11) is simple. \square

Chapter 4

Regular Formal Solutions

To design an algorithm for the computation of regular formal solutions is our second research objective, and in this chapter, we extend the results presented in the previous chapter to achieve this aim. We consider pseudo-linear systems with coefficients in the field of formal Laurent series with finite pole order and define and compute regular solutions. This chapter is organized as follows: we introduce systems of the first kind, the indicial polynomial of the system, and show that these systems can be transformed into a new system with constant coefficient matrix. We use this to define a fundamental matrix solution and we prove its existence. After introducing simple pseudo-linear systems we then give our algorithm. We finish by reviewing some technical conversions between different forms of linear functional systems and reviews three particular examples of linear functional systems and their regular solutions.

4.1 Solving Systems of the First Kind

Given a local system of the form (2.31), we are interested in defining regular solutions. In the case of singular linear differential, difference and q -difference systems the simplest situation arises when a singularity is of *first kind* ([47], [46], [7]). In this situation, it is well known that there exists a full fundamental system of solutions that have certain properties, and which are referred to as regular formal solutions. This motivates the following definition for general local pseudo-linear systems:

Definition 4.1.1. *A local system of the form (2.31) is said to be of the first kind if for its Poincaré-rank r , we have $r = 0$.*

We will show that in this situation we are also able to establish the existence of n linearly independent regular solutions. Our approach being constructive, this will give in principle, a method for computing them. However, it will be less efficient than the monomial-by-monomial method from Section 4.3.

For sake of simplicity, we will assume that K is algebraically closed in the remainder of this section. This assumption however, is not necessary for the algorithm, as we will discuss later.

Given a system of the first kind (2.31), we define its *indicial polynomial*

$$\varphi(\lambda) = \det(q^\lambda A_0 - c[\lambda]_q I). \quad (4.1)$$

Remark 4.1.1. *Note that $\varphi(\lambda)$ is a polynomial in λ if $q = 1$, otherwise it is a polynomial in q^λ . This could be seen as a q -analogue of a polynomial, hence our convention is to still refer to it as the indicial polynomial.*

We also define the (c, q) -spectrum

$$\sigma_{c,q} := \{\mu \in K \mid \varphi(\mu) = 0\}.$$

By definition, $\sigma_{c,q}$ is invariant with respect to similarity transformations of A_0 .

Lemma 4.1.1. *If $q = 1$, the roots of φ are of the form μ/c where μ is an eigenvalue of A_0 . Otherwise, assume that c is not an eigenvalue of $(q - 1)A_0$. Then the roots of φ are of the form $\log_q c - \log_q((1 - q)\mu + c)$.*

Proof If $q = 1$, this follows immediately from the definition of $\varphi(\lambda) = \det(A_0 - c\lambda I)$. Otherwise, we can see that $\varphi(\lambda) = 0$ if and only if

$$\det\left(\left(\frac{q-1}{c}A_0 - I\right)q^\lambda + I\right) = 0.$$

If the assumptions on c and q of the lemma are met, the matrix $\frac{q-1}{c}A_0 - I$ is invertible and hence we can apply matrix inversion in the above equation. This then leads to the claim. \square

A simple consequence of the previous lemma is:

Lemma 4.1.2. *We have $\sigma_{c,q} = \{0\} \iff A_0$ is nilpotent.*

In the remainder of this chapter we will always assume that c is not an eigenvalue of $(q - 1)A_0$.

Definition 4.1.2. *We say that a matrix $A_0 \in \mathcal{M}_n(K)$ has good (c, q) -spectrum if for any $\lambda \in \sigma_{c,q}$ and any nonzero $k \in \mathbb{Z}$, $\lambda + k \notin \sigma_{c,q}$.*

Using the concept of (c, q) -spectrum we can state the following

Lemma 4.1.3. *A matrix $A_0 \in \mathcal{M}_n(K)$ has good (c, q) -spectrum if and only if for any eigenvalue $\mu \in \bar{K}$ of A_0 and $k \in \mathbb{Z}^*$, $q^k \mu - c[k]_q$ is not an eigenvalue of A_0 .*

Proof For $q = 1$, this follows directly from Lemma 4.1.1. Otherwise, denote $\mu_1, \dots, \mu_n \in \bar{K}$ the eigenvalues of A_0 and assume that $\mu_i \neq q^k \mu_j - c[k]_q$ for all $\mu_i \neq \mu_j$ and $k \in \mathbb{Z}^*$. Still using Lemma 4.1.1, this can be seen to be equivalent to $q^{\lambda_i} \neq q^{\lambda_j - k}$ where λ_i, λ_j are roots of φ . This finishes the proof. \square

We shall now see that, using an appropriate transformation, any local pseudo-linear system of the first kind can always be assumed to have a leading matrix with good (c, q) -spectrum.

Lemma 4.1.4. *Given a local system of pseudo-linear equations of the form (2.31) which is of the first kind, there exists an invertible matrix $S \in \mathcal{M}_n(K[\tau])$ transforming the system into a system of the first kind whose leading matrix has good (c, q) -spectrum.*

Proof Assume A_0 does not have good (c, q) -spectrum. Using a constant transformation, we can assume that A_0 is block-diagonal

$$A_0 = \begin{pmatrix} A_0^{11} & & \\ & \ddots & \\ & & A_0^{ll} \end{pmatrix}$$

with $l > 1$ and the A_0^{ii} having only one eigenvalue. Without loss of generality, we can restrict ourselves to the case $l = 2$, the general case follows by repeated application of the process indicated below. Hence, assume that A_0 has the form

$$A_0 = \begin{pmatrix} A_0^{11} & \\ & A_0^{22} \end{pmatrix}$$

where A_0^{11} has dimension n_1 with indicial polynomial $\varphi^{11}(\lambda)$ and $\sigma_{c,q}^{11} = \{\lambda_1\}$ and A_0^{22} has dimension n_2 with indicial polynomial $\varphi^{22}(\lambda)$ and $\sigma_{c,q}^{22} = \{\lambda_2\}$, and furthermore, $\lambda_2 = \lambda_1 + h$ with $h \in \mathbb{N}^+$. Hence we have (up to a constant factor, which we can assume to be 1) $\varphi^{22}(\lambda) = \varphi^{11}(\lambda - h)$.

Let $T = \text{diag}(I_{n_1}, I_{n_2}\tau)$ and compute

$$\tilde{A} = T_{\delta, \phi}[A] = \begin{pmatrix} A^{11} & A^{12}\phi(\tau) \\ A^{21}\tau^{-1} & A^{22}\tau^{-1}\phi(\tau) - \tau^{-\omega-1}\delta(\tau)I_{n_2} \end{pmatrix}. \quad (4.2)$$

Its leading term is

$$\tilde{A}_0 = \begin{pmatrix} A_0^{11} & 0 \\ \tilde{A}_1^{21} & A_0^{22}q - cI_{n_2} \end{pmatrix}. \quad (4.3)$$

The indicial polynomial of this new leading matrix is the product of $\varphi^{11}(\lambda)$ and

$$\begin{aligned}\tilde{\varphi}^{22}(\lambda) &= \det(q^\lambda(qA_0^{22} - cI) - c[\lambda]_q I) \\ &= \det(q^{\lambda+1}A_0^{22} - c[\lambda+1]_q I) \\ &= \varphi^{22}(\lambda+1)\end{aligned}$$

where we have used the additive property of the q -bracket. Hence, we have reduced the difference in the roots of $\varphi(\lambda)$ to $h-1$. Iterating this process $h-1$ more times leads to a leading matrix with good (c, q) -spectrum as required. The constructed transformation matrix S is a finite product of constant invertible matrices with coefficients in K and diagonal polynomial matrices. \square

Proposition 4.1.1. *A system of the form (2.31) that has a singularity of the first kind and whose leading matrix A_0 has good (c, q) -spectrum is equivalent to a local pseudo-linear system of the first kind whose coefficient matrix is a constant matrix A_0 .*

Proof We will show that there is a change of variable $Y = TZ$ where

$$T = I + \sum_{i=1}^{\infty} T_i \tau^i, \quad T_i \in \mathcal{M}_n(K)$$

such that $T_{\delta, \phi}[A] = A_0$. We proceed by induction, assuming that we have found a transformation

$$T^{(h)} = I + \sum_{i=1}^h T_i \tau^i$$

such that the coefficient matrix of the transformed system is

$$T_{\delta, \phi}^{(h)}[A] = A_0 + O(\tau^{h+1})$$

and we show that transforming with

$$T^{(h+1)} = T^{(h)} + T_{h+1} \tau^{h+1} \tag{4.4}$$

yields

$$T_{\delta, \phi}^{(h+1)}[A] = A_0 + O(\tau^{h+2}) \tag{4.5}$$

increasing the order of accuracy in the transformed coefficient matrix.

Inserting (4.4) into (4.5) and comparing leading terms, we find the matrix equation

$$(A_0 q^{h+1} - c[h+1]_q I) T_{h+1} - T_{h+1} A_0 = B^{(h+1)}$$

where the leading coefficient of $B^{(h+1)} \in \mathcal{M}_n(K)$ depends only on terms in A of higher order than $h + 1$. This equation is a matrix equation of the form $CX - XD = E$ which can be solved uniquely for X if and only if C and D have no eigenvalues in common. But this is in fact (Lemma 4.1.3) implied by the assumption that A_0 has good (c, q) -spectrum. This equation can hence be solved uniquely for T_{h+1} . This finishes the proof. \square

Applying Proposition 4.1.1, we can transform any system of the first kind to a system of the form

$$\tau^{-\omega} \delta Y = J \phi Y \quad (4.6)$$

where J is a constant matrix. Using a constant transformation, we can furthermore assume that J is in Jordan normal form. Without loss of generality we can consider the different blocks of J individually. Hence, for such a block, the indicial polynomial has only one root $\mu \in K$. If $\mu \neq 0$, we show that we can introduce a scalar change of variable that reduces our considerations to the case where the indicial polynomial has the only root zero.

Denote by e_λ a scalar function in some extension of F such that

$$\phi(e_\lambda)/e_\lambda = q^\lambda + O(\tau) \quad (4.7)$$

and

$$\tau^{-\omega} \delta(e_\lambda)/e_\lambda = c[\lambda]_q + O(\tau) \quad (4.8)$$

where q, c and $[\lambda]_q$ are defined as in Chapter 2.

Remark 4.1.2. *These definitions extend (??) and (??) to arbitrary exponents λ . A priori, it is not obvious that e_λ exists, we show in Section 4.4 that e_λ can be defined using the exponential and logarithm function for a wide range of linear functional systems.*

Furthermore, let u be an element of an extension of F satisfying

$$\tau^{-\omega} \delta(u) = 1 + O(\tau). \quad (4.9)$$

Then we have:

Lemma 4.1.5. *Consider the system (4.6) with associated indicial polynomial $\varphi_J(\lambda)$ and assume that J has only one Jordan block. Let μ be a root of φ_J . Then the change of variable $Y = e_\mu \tilde{Y}$ yields the new system*

$$\tau^{-\omega} \delta \tilde{Y} = N \phi \tilde{Y} \quad (4.10)$$

where $N \in \mathcal{M}_n(\mathcal{O})$, $N = N_0 + O(\tau)$ with N_0 a nilpotent matrix with single Jordan block of the same size.

Proof Carrying out the change of variable $Y = e_\mu \tilde{Y}$ we obtain the new system (4.10) where $N = J\phi(e_\mu)/e_\mu - \tau^{-\omega} I\delta(e_\mu)/e_\mu$. Taking into account the properties (4.7) and (4.8) we compute the indicial polynomial of the new system as

$$\begin{aligned}\varphi_{N_0}(\lambda) &= \det(q^\lambda(q^\mu J - c[\mu]_q I) - c[\lambda]_q I) \\ &= \varphi_J(\lambda + \mu).\end{aligned}$$

This shows that $\sigma_{c,q} = \{0\}$ for the new system. Applying Lemma 4.1.2 we obtain that N_0 is nilpotent. \square

We will now assume that we have applied the previous lemma in order to have a system of the form (4.10). Recall that N has a nilpotent constant term (but is not a constant matrix in general). Denote $\tilde{\delta} = \tau^{-\omega}\delta$ and, if $\phi \neq \text{id}$, write $\tilde{\delta} = \tilde{\gamma}(\phi - \text{id})$ where $\tilde{\gamma} = \tau^{-\omega}\gamma$. Note that $\omega = 0$ for this new pseudo-derivation. Also, slightly abusing notation, we will now denote by n the size of the single Jordan block of N_0 , which equals the dimension of the system (4.10).

We now show how to construct n linearly independent regular solutions of the above system: define the matrix

$$U = \begin{pmatrix} 1 & u & \cdots & \frac{1}{(n-1)!}u^{n-1} \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & u \\ 0 & \cdots & 0 & 1 \end{pmatrix}. \quad (4.11)$$

By definition, we have that U is a matrix exponential

$$U = e^{uN_0}.$$

This gives the following formula for the inverse of U :

$$U^{-1} = e^{-uN_0},$$

in other words, U^{-1} is obtained by replacing u by $-u$ in (4.11).

Proposition 4.1.2. *A local pseudo-linear system of the form (4.10) admits a fundamental matrix solution of the form*

$$\tilde{Y}(\tau) = Z(\tau)U \quad (4.12)$$

where $Z \in \text{GL}(n, \mathcal{O})$, and U is defined as in (4.11).

In order to prove this proposition, we need some intermediate results. The proof of the

following technical lemma will be left to the reader.

Lemma 4.1.6. *If $\phi \neq \text{id}$ one has*

$$\tilde{\delta}(u^i) = \sum_{l=1}^i \tilde{\gamma}^{1-l} \binom{i}{l} u^{i-l} (\tilde{\delta}(u))^l.$$

Lemma 4.1.7. *Performing the change of variable $\tilde{Y} = ZU$ in (4.10) leads to a new system of the form*

$$\tilde{\delta}Z = N\phi(Z)V - \phi(Z)W \tag{4.13}$$

where $V, W \in \mathcal{M}_n(\mathcal{O})$.

Proof First note that in the above lemma the system (4.10) is viewed as a matrix equation, i.e. the unknown \tilde{Y} is an $n \times n$ matrix rather than a vector.

One easily verifies that substituting $\tilde{Y} = ZU$ into (4.10) gives a system of the above form with

$$V = \phi(U)U^{-1}$$

and

$$W = \tilde{\delta}(U)U^{-1}.$$

We only have to show that $W \in \mathcal{M}_n(\mathcal{O})$ since either $\phi = \text{id}$ in which case the claim $V \in \mathcal{M}_n(\mathcal{O})$ follows trivially, or $\phi = \tilde{\gamma}^{-1}(\tilde{\delta} - \text{id})$ in which case the claim for V follows from the definition of W and $W \in \mathcal{M}_n(\mathcal{O})$.

Carrying out the matrix multiplication, we find

$$W = \begin{pmatrix} 0 & w_1 & \cdots & w_{n-1} \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & w_1 \\ 0 & \cdots & 0 & 0 \end{pmatrix}$$

where for $k = 1, \dots, n-1$

$$w_k = \sum_{i=1}^k \frac{1}{i!(k-i)!} (-u)^{k-i} \tilde{\delta}(u^i). \tag{4.14}$$

If $\phi \neq \text{id}$ then, using Lemma 4.1.6 and rearranging the sums, we obtain

$$\begin{aligned}
w_k &= \sum_{i=1}^k \sum_{l=1}^i \frac{1}{l!(i-l)!(k-i)!} (-1)^{k-i} u^{k-l} \tilde{\gamma}^{1-l} (\tilde{\delta}(u))^l \\
&= \sum_{l=1}^k \sum_{i=l}^k \frac{1}{l!(i-l)!(k-i)!} (-1)^{k-i} u^{k-l} \tilde{\gamma}^{1-l} (\tilde{\delta}(u))^l \\
&= \sum_{l=1}^k \frac{(-1)^k}{l!} \tilde{\gamma}^{1-l} u^{k-l} (\tilde{\delta}(u))^l \sum_{i=l}^k \frac{(-1)^i}{(i-l)!(k-i)!} \\
&= \sum_{l=1}^k \frac{(-1)^{k+l}}{l!(k-l)!} \tilde{\gamma}^{1-l} u^{k-l} (\tilde{\delta}(u))^l \sum_{i=0}^{k-l} \binom{k-l}{i} (-1)^i 1^{k-l-i} \quad (4.15) \\
&= \frac{1}{k! \tilde{\gamma}^{k-1}} (\tilde{\delta}(u))^k \in \mathcal{O}. \quad (4.16)
\end{aligned}$$

The last expression follows by remarking that the inner sum in (4.15) equals zero whenever $l < k$, and $v(\tilde{\gamma}^{-j}) \geq 0$ for all $j \in \mathbb{N}$. This proves the lemma in the case $\phi \neq \text{id}$. If $\phi = \text{id}$ then the system simplifies to $\tilde{\delta}Z = NZ - ZN_0\delta(u)$ which proves (4.13) in this case by (4.14). \square

Lemma 4.1.8. *Consider the pseudo-linear system*

$$P\tilde{\delta}(Z) - \tilde{\delta}(Z)Q = NZ - ZM \quad (4.17)$$

where $P, Q, M, N \in \mathcal{M}_n(\mathcal{O})$ and their respective constant terms P_0, M_0, N_0 are strictly upper-triangular (hence, nilpotent), furthermore M_0 and N_0 having the same Jordan normal form with one Jordan block of size n , and Q_0 is upper-triangular with all diagonal entries equal to 1. Then this system has a matrix solution $Z \in \text{GL}(n, \mathcal{O})$.

Proof Inserting a solution of the form

$$Z = \sum_{i=0}^{\infty} Z_i \tau^i \quad (Z_i \in \mathcal{M}_n(K))$$

gives the necessary condition

$$N_0 Z_0 - Z_0 M_0 = 0.$$

An explicit solution $Z_0 = C_S^{-1} C_T \in \text{GL}(n, K)$ of this equation can be derived where C_S and C_T are similarity transformations needed to bring N_0 and M_0 to their (same) Jordan normal form.

Assume we have found parts of a solution

$$Z^{(h)} = Z_0 + \sum_{i=1}^h Z_i \tau^i \quad (Z_i \in \mathcal{M}_n(K))$$

that satisfies

$$P\tilde{\delta}(Z^{(h)}) - \tilde{\delta}(Z^{(h)})Q - (NZ^{(h)} - Z^{(h)}M) = O(\tau^{h+1}),$$

we put

$$Z^{(h+1)} = Z^{(h)} + Z_{h+1}\tau^{h+1}.$$

Inserting into the system and comparing the coefficients of τ^{h+1} one finds the matrix equation

$$(c[h+1]_q P_0 - N_0)Z_{h+1} - Z_{h+1}(c[h+1]_q Q_0 - M_0) = R_{h+1}$$

where R_{h+1} only depends on terms in Z and the various coefficients of (4.17) of lower order. According to the assumptions of the lemma, the first coefficient in this matrix equation is nilpotent and the second has the unique non-zero eigenvalue $c[h+1]_q$ for any value of $h > 0$. As we have already remarked in the proof of Proposition 4.1.1 earlier in this chapter, this implies there is a unique solution Z_{h+1} for any value of R_{h+1} . \square

Lemma 4.1.9. *The system (4.13) has a matrix solution $Z \in \text{GL}(n, \mathcal{O})$.*

Proof Assume first that $\phi \neq \text{id}$. We recall that $W \in \mathcal{M}_n(\mathcal{O})$ is strictly upper-triangular. We will show that the system can be rewritten as a system of the form (4.17) in Lemma 4.1.8. By using $\phi = \tilde{\gamma}^{-1}\delta + \text{id}$ we can express the system as

$$\tilde{\delta}(Z) = N\phi(Z)(I + \tilde{\gamma}^{-1}W) - \phi(Z)W.$$

We remark that $I + \tilde{\gamma}^{-1}W \in \text{GL}(n, \mathcal{O})$ and furthermore is a unimodular matrix. Hence, we can consider the system

$$\tilde{\delta}(Z)(I + \tilde{\gamma}^{-1}W)^{-1} = N\phi(Z) - \phi(Z)M$$

where $M = W(I + \tilde{\gamma}^{-1}W)^{-1} \in \mathcal{M}_n(\mathcal{O})$ is strictly upper-triangular. Finally, we can rewrite the system using only $\tilde{\delta}$ as

$$P\tilde{\delta}(Z) - \tilde{\delta}(Z)Q = NZ - ZM$$

where $Q = -(I + \tilde{\gamma}^{-1}W)^{-1} - \tilde{\gamma}^{-1}M$ is upper-triangular with all diagonal entries equal to 1 and $P = -\tilde{\gamma}^{-1}N$ is strictly upper-triangular.

In order to show that this is indeed a system of the form (4.17), the last property that remains to be proved is that the Jordan normal form of M_0 has one block of size n . By inspecting the structure of M one can see that all its elements in the first off-diagonal are equal to $w_1 = \tilde{\delta}(u)$ which is a unit in \mathcal{O} , hence the Jordan normal form of M_0 consists of one block of maximal size.

If $\phi = \text{id}$, we have previously seen that the system (4.13) simplifies to

$$\tilde{\delta}Z = NZ - ZN_0\delta(u).$$

Using techniques very similar to that in the proof of the previous lemma, we can assert that a matrix solution $Z \in \text{GL}(n, \mathcal{O})$ can be constructed also in this case. \square

It is now easy to prove Proposition 4.1.2 since a solution \tilde{Y} of the system (4.10) is given by ZU where Z is a solution of the system (4.13). But Lemma 4.1.9 shows the existence of a solution $Z \in \text{GL}(n, \mathcal{O})$ of the latter system (4.13).

By applying the various transformations that were used in Lemma 4.1.4, the proof of Proposition 4.1.1 and, for each eigenvalue of the coefficient matrix of the resulting constant system, combining the scalar changes of variable as in Lemma 4.1.5 and the result of Proposition 4.1.2 into the block-diagonal matrix in (4.18), we can now derive the structure of a regular fundamental matrix solution of a system of the first kind.

Theorem 4.1.1. *A local pseudo-linear system of the form (2.31) which is of the first kind admits a fundamental matrix solution of the form*

$$Y(\tau) = H(\tau) \text{diag}(e_{\lambda_1}U_1, \dots, e_{\lambda_k}U_k) \tag{4.18}$$

where $H(\tau) \in \mathcal{M}_n(\mathcal{O})$, $\lambda_1, \dots, \lambda_k \in K$, e_{λ_i} defined as previously and the U_i are the matrices defined by (4.11) where n is replaced by the size of the various Jordan blocks in J .

Remark 4.1.3. *We refer to Section 2.4 where we show how this form of regular solutions can be expressed using the exponential and logarithm function in the case of differential, difference and q-difference systems.*

In [25], we have defined regularity of an arbitrary pseudo-linear system in the following way:

Definition 4.1.3. *A system of the form (2.31) is called regular if there exists a matrix $T \in \text{GL}(n, F)$ such that $T_{\delta, \phi}[A]$ is of the first kind.*

Together with this definition of regularity and Theorem 4.1.1, the class of regular pseudo-linear systems is characterised in terms of the structure of their solutions as follows:

Theorem 4.1.2. *A local pseudo-linear system of the form (2.31) is regular singular if and only if it admits n linearly independent solutions that are all of the form*

$$y = e_\lambda \sum_{i=0}^s h_i \frac{u^{s-i}}{(s-i)!} \quad (4.19)$$

with some $\lambda \in K$, $s \in \mathbb{N}$ and $h_i \in K[[\tau]]^n$.

Proof This follows by inspecting the columns of the fundamental matrix solution (4.18) and taking into account the additional transformation with the matrix T . \square

Having obtained the structure of a regular formal solution, we define:

Definition 4.1.4. *A formal solution of the form (4.19) is referred to as a regular formal solution. We denote $\text{reg}(y) = e_\lambda$ as the regular part of y .*

This will be useful as part of the structure of general formal solutions in the next chapter.

4.2 The Dimension of the Regular Solution Space

In the previous section we have clarified the structure of local solutions in the case of a regular singularity. But, even for an irregular singularity, there may still be solutions of the form (4.18) which we still refer to as regular solutions. In this section, we will generalise the concept of simple systems from the differential case ([21]) to pseudo-linear systems and show that the number of linearly independent regular solutions can be computed once the input system has been converted to such a simple system.

4.2.1 Simple Pseudo-Linear Systems

The goal of this section is to show that for a pseudo-linear system that is super-irreducible, the associated polynomial θ_r reveals the dimension of the regular solution space. The idea is to remark that a super-irreducible system can be rewritten as a system of the more general form

$$D\tilde{\delta}Y = N\phi Y \quad (4.20)$$

where $\tilde{\delta} = \tau^{-\omega}\delta$, D is a diagonal polynomial matrix and $N \in \mathcal{M}_n(\mathcal{O})$. This had been first done in the differential case ([15]) and later in the difference case ([4]) and we presented the q -difference case in ([25]). We now state this in the general pseudo-linear setting.

The pseudo-linear operator that corresponds to (4.20) is

$$L_{\tilde{\delta},\phi} = N\phi - D\tilde{\delta}. \quad (4.21)$$

Here, the matrices D and N are given by the matrix factorisation

$$\tau^{-r}A = D^{-1}N \quad (4.22)$$

where

$$D = \text{diag}(\tau^{\alpha_1}, \dots, \tau^{\alpha_n})$$

and $\alpha_i = \min(0, v(R_i^A) - r)$. Here R_i^A denotes the i th row of the matrix A . Following [15], [4] and [25], we make the following

Definition 4.2.1. *A system as in (4.20) is called a simple pseudo-linear system if $\det(D_0\lambda - N_0) \neq 0$.*

The following computation reveals that the matrix factorisation (4.22), leading to the system (4.20), indeed leads to a simple system. One has

$$\det(D_0\lambda - N_0) = \pi(\det(D\lambda - N)) = \pi(\tau^{sr} \det(A - \lambda I)) = \theta_r(\lambda) \neq 0 \quad (4.23)$$

since the system is super-irreducible. By using one of the reduction algorithms mentioned in the previous section, we can effectively compute θ_r .

Using a change of unknown of the form $Y = TZ$ and left-multiplication by S where $S, T \in \text{GL}(n, \mathcal{O})$ and S_0, T_0 non-singular, a simple system can be transformed into an equivalent new simple system

$$\tilde{D}\tilde{\delta}Z = \tilde{N}\phi Z$$

where $\tilde{D} = SDT$ and $\tilde{N} = S(N\phi(T) - D\tilde{\delta}(T))$. We will now give a decomposition lemma for simple pseudo-linear systems that extends the Generalised Splitting Lemma (Proposition 3.1 in [63]) known for simple linear differential systems. We call $D_0\lambda - N_0$ the *leading pencil* of the simple system (4.20). Note that this is a *regular matrix pencil*, for which a definition of eigenvalues exists (see e.g. [63]). Denote by $\text{spec}(A + \lambda B)$ the set of eigenvalues of the regular matrix pencil $A + \lambda B$.

Proposition 4.2.1 ([9], Proposition 4.1). *Consider a simple operator $D\tilde{\delta} - N\phi$ and assume that its leading pencil is block-diagonal*

$$D_0\lambda - N_0 = \text{diag}(D_0^{11}\lambda - N_0^{11}, D_0^{22}\lambda - N_0^{22})$$

with $\det(D_0^{11}\lambda - N_0^{11}) \equiv 1$ and $\det(D_0^{22}\lambda - N_0^{22}) = \theta_r$. Then there exist $S, T \in \text{GL}(n, \mathcal{O})$ with $S_0 = T_0 = I_n$, transforming the operator into a new operator that is block-diagonal with block sizes matching the block structure of its leading pencil.

Proof We reproduce here the proof in [9]. Let us rewrite the definition of equivalence of

simple systems as

$$\hat{S}\tilde{D} = DT, \quad \hat{S}\tilde{N} = N\phi(T) - D\tilde{\delta}(T) \quad (4.24)$$

where $\hat{S} = S^{-1}$. We shall show that a transformation with matrices of the special form

$$S^{-1} = \begin{pmatrix} I & S^{12} \\ 0 & I \end{pmatrix}, \quad T = \begin{pmatrix} I & T^{12} \\ 0 & I \end{pmatrix} \quad (4.25)$$

leads to an operator of the form

$$\tilde{D}\tilde{\delta} - \tilde{N}\phi = \begin{pmatrix} \tilde{D}^{11}\tilde{\delta} - \tilde{N}^{11}\phi & 0 \\ * & \tilde{D}^{22}\tilde{\delta} - \tilde{N}^{22}\phi \end{pmatrix}.$$

It is easily seen that the complete block-diagonalisation can then be achieved using an additional transformation with matrices as in (4.25) where upper-right and lower-left blocks are exchanged.

Assume that (4.24) is satisfied up to a certain order $h \in \mathbb{N}$, i.e. we have

$$\hat{S}^{(h)}\tilde{D} - DT^{(h)} = O(\tau^{h+1}) \quad (4.26)$$

and

$$\hat{S}^{(h)}\tilde{N} - N\phi(T^{(h)}) + D\tilde{\delta}(T^{(h)}) = O(\tau^{h+1}) \quad (4.27)$$

where $\hat{S}^{(h)}$ and $T^{(h)}$ are the truncated series

$$\hat{S}^{(h)} = I + \sum_{j=1}^h S_j \tau^j, \quad T^{(h)} = I + \sum_{j=1}^h T_j \tau^j.$$

Put

$$\hat{S}^{(h+1)} = \hat{S}^{(h)} + \hat{S}_{h+1} \tau^{h+1}, \quad T^{(h+1)} = T^{(h)} + T_{h+1} \tau^{h+1}.$$

Transforming the operator with $\hat{S}^{(h+1)}$ and $T^{(h+1)}$ yields, taking into account the assumptions (4.26) and (4.27), the matrix equations

$$\hat{S}_{h+1} \tau^{h+1} \tilde{D} - DT_{h+1} \tau^{h+1} = O(\tau^{h+1})$$

and

$$\hat{S}_{h+1} \tau^{h+1} \tilde{N} - NT_{h+1} \phi(\tau^{h+1}) + DT_{h+1} \delta(\tau^{h+1}) = O(\tau^{h+1}).$$

Identifying the coefficient of τ^{h+1} and combining the resulting equations using matrix pencil notation gives:

$$\hat{S}_{h+1}(\tilde{D}_0 c[\lambda]_q - q^\lambda \tilde{N}_0) - (D_0 c[\lambda + h + 1]_q - q^{\lambda+h+1} N_0) T_{h+1} = \lambda P_{h+1} - Q_{h+1}.$$

This matrix equation is an equation of the form $X(A + \lambda B) - (C + \lambda D)Y = E + \lambda F$. It is well known [40] that a unique solution of this equation exists provided that the two involved matrix pencils are regular matrix pencils and do not have any eigenvalues in common. It can be seen that, from the assumptions of the proposition, these conditions are met and we can hence find unique coefficients for the transformation matrices S and T . \square

Theorem 4.2.1. *Consider a pseudo-linear system of the form (2.31) that is super-irreducible with associated polynomial θ_r . Assume furthermore that c is not an eigenvalue of the regular matrix pencil $D_0\lambda - (q-1)N_0$. Then there exist precisely $\deg(\theta_r)$ linearly independent regular solutions.*

Proof First, we convert the system to a simple system as explained earlier in this section. Then for the leading pencil we have $\det(D_0\lambda - N_0) \neq 0$ and we can find a constant transformation so that the leading pencil of the transformed system is block-diagonal

$$D_0\lambda - N_0 = \text{diag}(D_0^{11}\lambda - N_0^{11}, D_0^{22}\lambda - N_0^{22})$$

where $\det(D_0^{11}\lambda - N_0^{11}) \equiv 1$ and $\det(D_0^{22}\lambda - N_0^{22}) = \theta_r(\lambda)$ ([63]). For the second block, the additional assumption on c translates into c not being an eigenvalue of $(q-1)D_0^{22}\lambda - N_0^{22}$. We can now apply Proposition 4.2.1 in order to decouple the system. This will give two new independent simple systems, the second of which will be of the form

$$D^{22}\tilde{\delta}Y = N^{22}\phi Y$$

with $D_0^{22} = I$. Multiplying by the inverse of D^{22} , this can be converted into a system of the form $\tilde{\delta}Y = B\phi Y$ with $B \in \mathcal{M}_n(\mathcal{O})$ which is of the first kind and $\det(B_0 - \lambda I) = \theta_r(\lambda)$. In addition, we have that c is not an eigenvalue of $(q-1)B_0$. According to Theorem 4.1.1, we can find a regular fundamental matrix solution of this system, from which the claim follows. \square

4.3 The Algorithm: Monomial-by-Monomial Method

In the previous sections we have seen that the parameters (ω, c, q) characterise the local behaviour of a pseudo-linear system $\delta Y = A\phi Y$ and, together with the definition of suitable scalar functions e_λ and u , allow for the definition and computation of regular solutions. The proofs we have given are constructive and could be implemented in principle in a Computer Algebra System, however this would not be very efficient. For further information, we refer

to the discussion at the end of this section.

The method for computing regular solutions we will present in this section is inspired by [21] for solving the differential case and could be called a “monomial-by-monomial” method as it will determine the different monomials of the series parts of the regular solutions successively, without computing an explicit recurrence relation. The advantages of this approach have already been established in the differential case. The method can be summarised as follows: we compute a super-irreducible form of the input system and then convert it to a simple system. From this, we will see that we can compute the indicial polynomial $\varphi(\lambda)$. We then design a method that computes a subclass of regular solutions for a more general (inhomogeneous) system. We are then able to reduce the task of determining the full set of regular solutions (of the original input system) to that of computing the subclass of regular solutions of several non-homogeneous systems.

For the remainder of this chapter, we now return to the situation where K is not necessarily algebraically closed and denote by \bar{K} the algebraic closure of K where necessary.

4.3.1 Idea of the Algorithm

We have seen previously that a system of pseudo-linear equations admits p linearly independent solutions of the form

$$y = e_\lambda \sum_{i=0}^s h_i \frac{u^{s-i}}{(s-i)!} \quad (4.28)$$

where $p = \deg(\theta_r)$. We will first explain the idea of the algorithm and, in order to simplify the exposition, assume for now that $e_\lambda = 1$ (we will explain in the following section how to proceed in the general case). Let us focus on the case $\phi \neq \text{id}$, as otherwise the method essentially coincides with that of the differential case. Consider the system

$$L_{\delta,\phi}(Y) = D\bar{\delta}Y - N\phi Y = 0 \quad (4.29)$$

with $L_{\delta,\phi}$ a simple operator. Assume there exists a chain of k ($0 \leq k+1 \leq p$) linearly independent solutions of the form

$$y_i = \sum_{j=0}^i h_{ij} \frac{u^{i-j}}{(i-j)!} \quad (i = 0, \dots, k). \quad (4.30)$$

If we define

$$\tilde{U} = \text{diag}(U, I_{n-k})$$

where U is a $k \times k$ matrix of the structure as defined in (4.11), the change of variable $Y = Z\tilde{U}$ will lead to the new pseudo-linear system

$$D\tilde{\delta}Z + (D\phi ZW - N\phi ZV) = 0 \quad (4.31)$$

where $W = \tilde{\delta}\tilde{U}\tilde{U}^{-1} \in \mathcal{M}_n(\mathcal{O})$ and $V = \phi\tilde{U}\tilde{U}^{-1} \in \text{GL}(n, \mathcal{O})$. This system is of very similar structure as the system (4.13). Inspecting this latter system by columns, we find for the i th column z_i the equation

$$\begin{aligned} D\tilde{\delta}z_i + D \sum_{j=1}^{i-1} w_{i-j}\phi z_j - N \left(\sum_{j=1}^{i-1} \tilde{w}_{i-j}\phi z_j + \phi z_i \right) &= 0 \\ \iff L(z_i) + \sum_{j=1}^{i-1} w_{i-j}\tilde{\gamma}^{-1}L(z_j) + D \sum_{j=1}^{i-1} w_{i-j}z_j &= 0. \end{aligned} \quad (4.32)$$

The last equation (4.32) has been obtained from the previous one by substituting $\phi = \tilde{\gamma}^{-1}\tilde{\delta} + \text{id}$ in the first occurrence of ϕ and using the formula (4.16) for w_{i-j} (here, $\tilde{w}_{i-j} = \tilde{\gamma}^{-1}w_{i-j}$). Finding the h_{ij} in the solutions (4.30) is now reduced to finding column vector solutions $z_j \in \bar{K}((\tau))^n$ of (4.32). Using induction, equation (4.32) leads to the following lemma which shows how to compute the different h_{ij} as series solutions of some inhomogeneous system, where we will be dropping the index i in the h_{ij} for sake of simplicity.

Lemma 4.3.1. *For a regular solution of the form (4.28), we have*

1. $L_{\delta,\phi}(h_0) = 0$, hence, h_0 is a power series solution of the homogeneous system $L_{\delta,\phi}(y) = 0$.
2. For $i > 0$, $L_{\delta,\phi}(h_i) = \sum_{j=0}^{i-1} B_j h_j$ with $B_j \in \mathcal{M}_n(\bar{K}((\tau)))$, hence h_i is a power series solution of a non-homogeneous system.

Proof From [21], we know that in the differential case we have $B_{i-1} = -D\tilde{\delta}(u)$ and $B_j = 0$ for $1 \leq j < i-1$. If $\phi \neq \text{id}$, from equation (4.32) we can show using induction that $B_j = \frac{(-1)^{i-j}}{(i-j)!} (\delta(u))^{i-j} \tilde{\gamma}^{j-i+1} D$. \square

Remark 4.3.1. *A closer inspection of the form of B_j in the differential, difference and q -difference case reveals that we have in fact $B_j \in \mathcal{M}_n(\bar{K}[[\tau]])$. This means that in our implementation, it is sufficient to compute the individual h_i up to a fixed order k and this yields a basis of regular solutions up to that order k .*

s This leads to the algorithm **Regular_Solutions** as presented below. The algorithm uses the function **Series_Solutions** in order to compute power series solutions of the system,

we will be explained this method in more detail in the next section.

Regular_Solutions($L_{\delta,\phi}, b, k$)

Input: $L_{\delta,\phi}$ a simple pseudo-linear system with indicial polynomial φ ,

$b \in \bar{K}^n((\tau))$, $k \in \mathbb{N}^+$ the desired order

Output: all regular solutions of $L_{\delta,\phi}(y) = b$ up to order k .

1. `reg_sols := \emptyset` ;
2. `$s := 0$` ;
3. `$\tilde{b} := b$` ;
4. **while** `|reg_sols| < deg(φ)` **do**
 - (a) `$h_s := \text{Series_Solutions}(L_{\delta,\phi}, \tilde{b}, k)$` ;
 - (b) Compute new RHS \tilde{b} given by $\sum_{j=0}^{i-1} B_j h_j$ where B_j as in the proof of Lemma 4.3.1;
 - (c) `$y := \sum_{i=1}^s h_i \frac{\tau^{i-1}}{(i-1)!}$` ;
 - (d) `$s := s + 1$` ;
 - (e) `reg_sols := reg_sols \cup { y }`;
5. **return** `reg_sols`;

Note that the output of **Series_Solutions** contains parameters in order to allow for a simultaneous processing of all the solutions, in a similar fashion as the algorithm in [21] for linear differential system.

4.3.2 Computing the Series Solutions

In this section we consider the non-homogeneous system

$$L_{\delta,\phi}(y) = D\tilde{\delta}y - N\phi y = b \quad (4.33)$$

with $b \in \bar{K}^n((\tau))$ and $L_{\delta,\phi}$ a simple operator with associated polynomial $\theta_r(\lambda) = \det(D_0\lambda - N_0) \neq 0$. We look for solutions of the form

$$y = e_\lambda \sum_{i=0}^{\infty} y_i \tau^i \quad (\lambda \in \bar{K}, y_i \in \bar{K}^n)$$

where e_λ is defined as in Section 4.1.

Definition 4.3.1. For a given simple pseudo-linear system of the form (4.33), define its indicial polynomial as $\varphi(\lambda) = \det(q^\lambda N_0 - c[\lambda]_q D_0)$.

Since the associated polynomial $\theta_r(\lambda) \neq 0$, we also have $\varphi(\lambda) \neq 0$. Consider a solution of the form

$$y = e_\lambda \sum_{i=0}^{\infty} y_i \tau^i \quad (\lambda \in \bar{K}, y_i \in \bar{K}^n).$$

By inserting y into (4.33) and comparing coefficients we obtain, amongst others, the equation

$$(q^\lambda N_0 - c[\lambda]_q D_0)y_0 = 0. \quad (4.34)$$

This shows that a necessary condition for the existence of $y_0 \in \bar{K}^n$ is that we find $\lambda \in \bar{K}$ such that $\varphi(\lambda) = 0$.

In the algorithm, we proceed as follows: for each root μ of θ_r (using Maple, we can calculate with algebraic numbers up to conjugation, there is no need to compute the splitting field of θ_r), we perform the change of variable $Y = e_\mu Z$. This yields a new simple system with indicial polynomial $\varphi(\lambda + \mu)$. We are now led to look for series solutions only, i.e. solutions of the form

$$y = \sum_{i=0}^{\infty} y_i \tau^i \quad (\lambda \in \bar{K}, y_i \in \bar{K}^n).$$

The algorithm for computing these series solutions follows closely the approach of [21] in the differential case. The idea is to compute successively monomials of the form $m\tau^\mu$ ($m \in \bar{K}^n$, $\mu \in \bar{K}$) of a solution y up to any order $k \in \mathbb{N}^+$ by repeating the following two steps:

1. Choose μ as an appropriate number. This will be either a suitable integer root of the indicial polynomial or the valuation of the RHS.
2. Set $y = m\tau^\mu + z$ and solve the new system in z . This will be of the form $L_{\delta,\phi}(z) = \tilde{b}$ where $\tilde{b} = b - L_{\delta,\phi}(m\tau^\mu)$.

The algorithm can be summarised in pseudo-code as follows:

Series_Solutions($L_{\delta,\phi}, b, k$)

Input: $L_{\delta,\phi}$ a simple pseudo-linear system,

$b \in \bar{K}^n((\tau))$, $k \in \mathbb{N}^+$ the desired order of solutions

Output: all series solutions of $L_{\delta,\phi}(y) = b$ up to order k

1. series_sols := \emptyset ;

2. $\mathcal{R} :=$ the set of the integer roots of the indicial polynomial φ of $L_{\delta,\phi}$;
3. $d := v(b)$;
4. $y := 0$;
5. **while** $d \leq k$ **do**
 - (a) **if** there exists a λ in \mathcal{R} such that $\lambda < d$ **then**
 - i. Set $\mu :=$ minimal element in \mathcal{R} ;
 - ii. $\mathcal{R} := \mathcal{R} \setminus \{\mu\}$;
 - (b) **elif** all elements of $\mathcal{R} \geq d$ **then**
 - i. Set $\mu := d$;
 - (c) Compute the monomial $m\tau^\mu$;
 - (d) $y := y + m\tau^\mu$;
 - (e) Perform the change of variable $y = m\tau^\mu + z$ in $L_{\delta,\phi}$ and compute the new RIIS
 $\tilde{b} = \tilde{b} - L_{\delta,\phi}(m\tau^\mu)$;
 - (f) $d := v(\tilde{b})$;
 - (g) $\text{series_sols} := \text{series_sols} \cup \{y\}$;
6. **return** series_sols ;

4.3.3 An Example

In order to illustrate the algorithm, in particular the computation of the logarithmic terms using Lemma 4.3.1, we shall give an example.

We load the ISOLDE library functions into a Maple session:

```
> with(ISOLDE):
```

We consider a linear system of difference equations of the form $\phi Y = AY$, with system matrix

```
> A:=matrix([[1,1+(1/x^2)], [0,1]]);
```

$$A := \begin{bmatrix} 1 & 1 + x^{-2} \\ 0 & 1 \end{bmatrix}$$

We now convert the system to a local system of pseudo-linear equations of the form $\delta Y = A\phi Y$ (see Section A.1 for more details).

```
> L:=LocalLinearDifferenceSystem(A,x,'Phi');
```

$$L := L1$$

The output is the symbol $L1$ which is used as a key for a lookup table of internal data structures for the lazy evaluation mechanism. The system matrix of the new system can be retrieved:

```
> mat_eval(GetSystemMatrix(L,x),x,0,2);
```

$$\begin{bmatrix} 0 & -2 - x^2 + 2x \\ 0 & 0 \end{bmatrix}$$

which is a polynomial matrix. We apply the super-reduction algorithm which yields the following output:

```
> tmp:=SuperReduction(L,lambda);
```

$$tmp := [L2, [[0, (1 + \lambda) \lambda], [-1]]]$$

The first element in this list is a symbol representing a super-irreducible system equivalent to the input system. The computed transformation matrix is

```
> T:=GetTransformation(tmp[1],x);
```

$$T := \begin{bmatrix} 0 & 1 \\ x & 0 \end{bmatrix}$$

We compute the coefficients of the super-irreducible system matrix up to order 4:

```
> mat_eval(GetSystemMatrix(tmp[1],x),x,0,4);
```

$$\begin{bmatrix} -x - x^2 - x^3 - x^4 & 0 \\ -2x - x^3 - x^4 & 0 \end{bmatrix}$$

The second item in the list returned by `SuperReduction` contains, amongst other information, the associated polynomial of the system.

```
> theta:=tmp[2][1][2];
```

$$\theta := (1 + \lambda) \lambda$$

In this example, we have $c = 1$ and $q = 1$:

```
> c:=GetLocalCharacteristics(L,'c');
```

```
q:=GetLocalCharacteristics(L,'q');
```

$$c := 1$$

$$q := 1$$

Hence, the indicial polynomial is $\varphi(\lambda) = \theta(\lambda)$. Its roots are $\lambda_1 = 0$ and $\lambda_2 = -1$. The

degree of the indicial polynomial is 2, therefore the dimension of the regular solutions space is 2 and the system is regular singular. Due to the integer differences of the roots of φ , we cannot detect the presence of logarithmic terms at this stage.

In order to find the series solutions up to order $k = 3$, we solve $L(h_0) = 0$ using the algorithm `Series_Solutions`. In our implementation, this can be done using the function `RegularSolutions` with an additional option “logfree”. This returns one linearly independent series solution associated with $\lambda = 0$.

```
> h_0:=RegularSolutions(L,x,3,{ 'logfree' });
```

$$h_0 := [[[-C_2, 0], \{\}]]$$

In order to find the remaining solution, using Lemma 5.1 we have to solve the non-homogeneous system $L(h_1) = b_1$ with $b_1 = -\delta(u)Dh_0$. Here, $D = I$ and $\delta(u) = \log(x/(1-x)) - \log(x)$. Since the order up to which we want to compute solutions is $k = 3$, it is sufficient to use a truncated series.

```
> du := convert(series(log(x/(1-x)) - log(x),x,3), 'polynom');
```

$$du := x + 1/2 x^2$$

We compute

```
> b_1 :=evalm(-du*[1,0]);
```

$$b_1 := [-x - 1/2 x^2, 0]$$

The algorithm `Series_Solutions` is then called again to compute h_1 .

```
> h_1:=RegularSolutions(L,x,b_1,3,{ 'logfree' });
```

$$h_1 := \left[\left[\left[\frac{1 + x - C_2 - x^2}{x}, 1/2 \right], \{\} \right] \right]$$

Using these results, we have obtained the two solutions $y_0 = h_0$ and $y_1 = h_1 + uh_0$. This can be verified from the default output when calling `RegularSolutions`:

```
> y:=RegularSolutions(L,x,3);
```

$$y := \left[\left[\left[\frac{2 - C_3 + x - C_4 - C_3 x^2 + 2 \ln(x) - C_3 x}{x}, -C_3 \right], \{\} \right] \right]$$

4.3.4 Discussion

As we have already indicated, the algorithm we have presented in this section is preferable to the “theoretical” method in Section 4.1, based on a constructive proof for the structure of the regular solutions. We will give some more ideas why this is a plausible assessment of

the situation.

The monomial-by-monomial algorithm only needs a super-irreducible form of the input system to be computed as a pre-requisite. This only requires a polynomial transformation matrix. In the theoretical method, several series of power series transformations are required, bringing the system into a new form with only constant coefficients, followed by additional transformations that may reintroduce formal power series coefficients (in the difference case) and also lead to a system of a more general form (the system (4.13)). Although from a complexity point of view, this is probably not substantially harder as a function of n , the computations in the coefficient field would require extensive use of lazy evaluation using formal power series. In contrast, the monomial-by-monomial method can be performed on a truncated system once the super-reduction algorithm has been applied.

The efficient handling of algebraic extensions is a crucial feature in any Computer Algebra algorithm. The main inconvenience with the theoretical method is that it will require the computation of the Jordan normal form of the leading matrix, which may require the computation of the splitting field of the characteristic polynomial of A_0 . Although it might be possible to reformulate our results using the rational Jordan normal form, it is still less convenient than the monomial-by-monomial method which only requires Gaussian elimination on various regular matrix pencils with coefficients in $K(\alpha)$ where α is a root of this characteristic polynomial.

Lastly, the case where roots of the indicial equation are present that differ by integers is resolved differently in the two discussed methods: the theoretical method applies a series of transformations in order to gradually reduce all integer differences to zero. The advantage of this approach is that once the system is of good (c, q) -spectrum, the structure of the regular solution space (in particular the existence of logarithmic terms) is known. The disadvantage is that carrying out the transformations might require a considerable amount of time and memory. For this reason, in the monomial-by-monomial method we do not seek to require good (c, q) -spectrum but rely on the fact that during execution of the algorithm, as the order of the computed monomials increases, the maintained set of conditions and parameters determines the structure of the solutions “up to current order”. This means that in some circumstances (for example, if the required order k is smaller than the biggest integer difference of roots) the algorithm does not find the final structure (which it would find eventually if a larger order will be requested).

4.4 Regular Solutions of Various Linear Functional Systems

In Theorem 4.1.1, we have given the structure of the regular solutions of a general pseudo linear system. In this section we will review various types of linear functional systems and show that the structure of the regular solutions in these individual cases is

$$y = \tau^\lambda \sum_{i=0}^s h_i \frac{\log^{i-1}(\tau)}{(i-1)!}$$

where $\tau^\lambda = e^{\lambda \ln(\tau)}$ and the log-function has an appropriate base. This has been established already in previous works studying these cases independently but now also follows from our general framework.

Differential Case

A singular system of linear differential equations is a system of the form

$$Y' = AY \tag{4.35}$$

where $A \in \mathcal{M}_n(K((x - x_0)))$ ($x_0 \in \bar{K}$ a finite singularity of the system) or $A \in \mathcal{M}_n(K((x^{-1})))$ for a singularity at infinity. We will detail the case of a finite singularity: here, we have $\tau = x - x_0$. The automorphism $\phi = \text{id}$ is trivial hence, $q = 1$. $\delta = \frac{d}{d\tau}$ is the standard derivation and in this case $\omega = v(\tau^{-1} \frac{d}{d\tau} \tau) = -1$. We have $\tau \frac{d}{d\tau} \tau = \tau$ so $c = 1$. It is clear that $e_\lambda = \tau^\lambda$. Choosing $u = \ln(\tau)$ satisfies the condition in (4.9) as $\tau^{-\omega} \delta(u) = 1$.

Difference Case

Systems of singular linear difference equations can be considered in either the form

$$Y(x+1) = A(x)Y$$

or

$$Y(x+1) - Y(x) = A(x)Y$$

where $A \in \mathcal{M}_n(K((x^{-1})))$. As explained in the previous section, we can convert either equation into a pseudo-linear system of the form (2.31)

$$\tau^{r-\omega} \delta Y = A(\tau) \phi Y,$$

with $\tau = x^{-1}$, ϕ the K -automorphism defined by $\phi\tau = \frac{\tau}{1-\tau}$ and $\delta = \phi - 1$. Note that ϕ is given as the inverse automorphism as in (2.34). We compute $\omega(\delta) = v(\tau^{-1}(\frac{\tau}{1-\tau} - \tau)) =$

$v(\frac{\tau}{1-\tau}) = 1$. We have $\phi(\tau) = \tau + O(\tau^2)$ hence $q = 1$ and we find $\tau^{-\omega}\delta(\tau) = 1 + O(\tau)$ so $c = 1$. We choose $e_\lambda = \tau^\lambda$, we then have

$$\phi(e_\lambda) = e_\lambda(1 + \lambda\tau + O(\tau^2))$$

using the series expansion of $e^{-\lambda \log(1-\tau)}$. This shows $\phi(e_\lambda)/e_\lambda = 1 + O(\tau) = q^\lambda + O(\tau)$ since $q = 1$. Furthermore,

$$\tau^{-1}\delta(e_\lambda)/e_\lambda = \lambda + O(\tau)$$

satisfies (4.8) since here, $c[\lambda]_q = \lambda$. We choose $u = \ln(\tau)$ as $\tau^{-\omega}\delta(\ln(\tau)) = \tau^{-\omega}(\ln(\frac{\tau}{1-\tau}) - \ln(\tau)) = -\tau^{-1}\ln(1-\tau) = 1 + O(\tau)$ satisfying (4.9).

q -Difference Case

As in the difference case, singular systems of linear q -difference equations have been considered in two forms:

$$Y(qx) = A(x)Y$$

or

$$Y(qx) - Y(x) = A(x)Y$$

with $q \in K \setminus \{0, 1\}$ and either $A \in \mathcal{M}_n(K((x)))$ (singularity at $x = 0$) or $A \in \mathcal{M}_n(K((x^{-1})))$ (singularity at $x = \infty$). We will detail the first case: we can convert either system into the local pseudo-linear system

$$\tau^{r-\omega}\delta Y = A(\tau)\phi Y,$$

where $\tau = x$. The automorphism is $\phi x = qx$ and we have $\delta = \phi - \text{id}$. We obtain $\omega(\delta) = v(q-1) = 0$ since $q \neq 1$. The quantity q is identical to that given by the definition of $\phi(x) = qx$. From $\tau^{-\omega}\delta(\tau) = (q-1)\tau$ we obtain $c = q-1$. Choosing $e_\lambda = \tau^\lambda$, we obtain

$$\phi(e_\lambda) = q^\lambda e_\lambda$$

and

$$\delta(e_\lambda) = (1 - q^\lambda)e_\lambda = c[\lambda]_q e_\lambda$$

since $c = q-1$. We choose $u = \log_q(\tau)$ since then $\tau^{-\omega}\delta(\log_q(\tau)) = \log_q(q\tau) - \log_q(\tau) = 1 + O(\tau)$ satisfies (4.9).

Chapter 5

Irregular Formal Solutions

In this chapter, we are concerned with the formal reduction of systems of pseudo-linear equations with formal Laurent series coefficients. We define a unifying notion of hyperexponential parts and of formal solutions of arbitrary pseudo-linear systems in the neighbourhood of a singularity. We give a generic formal reduction algorithm and apply this in order to compute a complete set of formal solutions of systems of linear differential, difference and q -difference equations. This results in a novel approach to the formal reduction in the difference and q -difference case.

5.1 Introduction

The aim of this chapter is to extend the progress in the algorithmic formal reduction that has been made in the differential system case [14, 63] to difference and q -difference systems. Rather than studying these systems separately, we adopt a unifying view. The benefits of this approach are twofold: firstly, the practical outcome is a generic algorithm that can be used for the individual systems by giving specific input parameters. This helps reducing code duplication and makes it easier to extend the algorithm eventually to, for example, higher-order systems. Secondly, from a theoretical point of view, our results show how concepts that look quite different at first glance can actually be encompassed in a unifying framework.

In this chapter, we choose pseudo-linear systems as the mathematical tool and address the following objectives:

- To give a unifying theoretical framework for defining local formal solutions of arbitrary pseudo-linear systems,
- To provide a generic and efficient formal reduction algorithm, that can also compute formal solutions for a wide range of classes of linear functional systems.

Let us summarise what the novel contributions of this chapter are, that resulted from the successful completion of the objectives above:

- Improvements in the formal reduction for linear difference systems analogous to those of [63] in the linear differential case: our algorithm uses minimal algebraic extensions of the coefficient field (this is tantamount to using minimal ramifications) and a finer splitting into smaller systems (which can be interpreted as a local factorisation of the original system), hence getting closer to a complete system equivalent of Malgrange’s local Newton-polygon factorisation method as in the scalar case [56, 72]. Furthermore, the unified algorithm is much simpler than previous methods, avoiding the distinction between two stages as in [17] and [18], that needed rewriting of the system.
- To our knowledge, our algorithm yields the first complete formal reduction method for linear q -difference systems.
- For both cases (difference and q -difference), we achieve a reduced amount of symbolic manipulations due to delayed introduction of irregular parts involving gamma functions and exponential terms. This makes it possible to benefit from the lazy evaluation approach as in the differential case.

This chapter is organised as follows: in the next section, we introduce the notations and mathematical concepts used throughout the chapter. In Section 5.2, we motivate the structure of the generic formal solutions. We then explain a first version of the formal reduction, based on using the Moser-algorithm, the classical Splitting-Lemma, so-called term-transformations, and a method for computing the system’s Katz-invariant. In the following section, inspired by the approach in the differential case [63], we refine the algorithm and make it more practical. We discuss how our method improves the existing algorithms for difference and q -difference systems. In Section 5.4 we show how to compute a full set of formal solutions from the output of the formal reduction. Throughout the chapter, we give examples to illustrate our approach. In the last section, we give the structure of formal solutions of differential, difference and q -difference equations. This will be helpful for the reader who wishes to know more details, as we have not seen this information in one single place before.

5.2 Formal Reduction of Pseudo-Linear Systems

In this section, we will give the first version of an algorithm for the formal reduction of pseudo-linear systems with formal power series coefficients. We start by defining formal solutions, their hyperexponential parts and their relationship with the formal reduction. We then explain the principle of our formal reduction. This will lead us to the formal reduction

as a method to compute all hyperexponential parts.

Let us remark that in general, if $\phi \neq \text{id}$, a system of the form (2.39) does not necessarily admit n non-trivial solutions. This is easy to see for systems that are written as in (2.32): if $\text{rank}(A) < n$, there are only $n - \text{rank}(A)$ non-zero solutions. The system (2.39) admits n non-trivial solutions if and only if that is also the case for the system that results from converting it to the format (2.32). Following the approach explained in the last section of the previous chapter, we see that the converted system is

$$\tilde{\phi}(Y) = (I - \gamma^{-1}\tilde{\phi}(\tau^{\omega-r}A))Y$$

where $\tilde{\phi} = \phi^{-1}$. Hence, a full set of solutions exists if and only if $\phi(\gamma)$ is not an eigenvalue of A .

In the remainder of this chapter, we will always assume that this assumption is satisfied. A particular consequence of this is that, if $r = \omega$, and $\gamma = -1$ (as in the systems for which we have defined our formal reduction algorithm), -1 is not an eigenvalue of A_0 . This condition will be necessary in the formal reduction algorithm in order to ensure that leading terms of hyperexponential parts can be computed, see Section 5.2.2.

5.2.1 Formal Solutions

In the case of linear differential, difference and q -difference systems, the structure of formal solutions is well known. We refer the reader to Section 5.5 for more technical details. Formal solutions are often expressed as formal fundamental matrix solutions (FFMS). For algorithmic purposes, it is more convenient to look at vector solutions, contained in the columns of a FFMS. Denote by y such a formal (vector) solution.

In the previous chapter, we have investigated the definition and computation of regular formal solutions. For this purpose, we introduced the regular part $\text{reg}(y)$ of a formal solution (Definition 4.1.4), and the scalar function u extending the role of the logarithm to the pseudo-linear case. We now define the *irregular part* in the following way:

Definition 5.2.1. *We say that a scalar function $\text{irr}(y)$ is an irregular part of a formal solution y with ramification index s and multiplicity m , if s is the smallest integer such that the change of variable $Y = \text{irr}(y)Z$, followed by the substitution $\tau \rightarrow \tau^s$, yields a new system of the form (2.39) that has a basis of m regular formal solutions.*

From this, we see that a formal solution of the system (2.39) can be written as

$$y = \text{irr}(y)\text{reg}(y)z \quad \text{with} \quad z \in \bar{K}[[\tau^{1/s}]]^n[u], s \in \mathbb{N}^+.$$

Example 5.2.1. Consider the following formal solution of a linear differential system $Y' = AY$

$$y(\tau) = e^{\frac{1}{\tau^2} + \frac{2}{\tau}} \tau^{\sqrt{2}} z(\tau)$$

where z is a formal power series. In this example, we have $\text{irr}(y) = e^{\frac{1}{\tau^2} + \frac{2}{\tau}}$, $\text{reg}(y) = \tau^{\sqrt{2}}$ and if we put $e_y = \text{irr}(y)\text{reg}(y)$ then we find with $\delta = \frac{d}{d\tau}$ and $\phi = \text{id}$:

$$\frac{\delta(e_y)}{\phi(e_y)} = -\frac{2}{\tau^3} - \frac{2}{\tau^2} + \frac{\sqrt{2}}{\tau} =: w \in F.$$

Note that this formal solution could then also be written as

$$y(\tau) = \exp\left(\int w\right) z(\tau).$$

This is the form considered in [63], where w was referred to as exponential part.

This example shows how combining regular and irregular parts simplifies the presentation of formal solutions, as already explained in [63, 72]. This is also true for formal solutions of linear difference and q -difference systems. In order to state this for pseudo-linear systems, one is led to introduce hyperexponential extensions of the pseudo-differential field (F, ϕ, δ) as an appropriate domain for $\text{irr}(y)\text{reg}(y)$.

Definition 5.2.2. We call an element h hyperexponential over the (ϕ, δ) -field (F, δ, ϕ) iff $\delta(h)/\phi(h) \in F$. For h hyperexponential, we denote $\text{cert}(h) = \delta(h)/\phi(h) \in F$ the certificate of h .

Remark 5.2.1. If $\phi = \text{id}$, the certificate of h is $\delta(h)/h$, the logarithmic derivative of h . Hence the certificate can be seen as a generalisation of the logarithmic derivative. Note that hyperexponential extensions and certificates have already been defined elsewhere (see e.g. [54, 51]). Our definition is slightly different, but related to existing ones, as partly expressed in (i) and (ii) of the following

Lemma 5.2.1. Let h be hyperexponential over (F, δ, ϕ) . Then it holds

- (i) $\phi(h)/h \in F$,
- (ii) $\delta(h)/h \in F$,
- (iii) $v(\delta(h)/h) - v(\text{cert}(h)) = v(\phi(h)/h)$,
- (iv) $\text{lc}(\delta(h)/h) = \text{lc}(\text{cert}(h)) \cdot \text{lc}(\phi(h)/h)$.

Proof If $\phi = \text{id}$, (i) and (ii) are trivial. Otherwise, using $\delta = \gamma(\text{id} - \phi)$, one obtains $\gamma(h/\phi(h) - 1) \in F$ from which (i) follows; (ii) can be shown very similarly. Now, given that

$\delta(h)/\phi(h) \in F$, we can write

$$\frac{\delta(h)}{\phi(h)} = \text{lc}(\text{cert}(h)) \cdot \tau^{v(\text{cert}(h))} + \dots$$

Comparing valuations and leading coefficients on both sides of the previous equation, one derives (iii) and (iv). \square

The following definition combines irregular and regular parts, generalising exponential parts as in [63, Section 5.2]. Note that we anticipate here Theorem 5.2.2 concerning the structure of $\text{irr}(y)$ and $\text{reg}(y)$.

Definition 5.2.3. *Let y be a formal solution of (2.39) with irregular part $\text{irr}(y)$ of ramification index s and multiplicity m . We define*

$$\text{hyp}(y) = \text{cert}(\text{irr}(y)\text{reg}(y)) \bmod \tau^{\omega+1/s}$$

as the hyperexponential part of y (with ramification index s and multiplicity m).

This shows that in general, in order to compute hyperexponential parts, one needs to consider hyperexponential extensions of the field $(K((\tau^{1/s})), \phi, \delta)$. However, during the formal reduction algorithm, we can restrict ourselves to $s = 1$, using appropriate substitutions (the operation (R3) as defined in the next section).

5.2.2 Formal Reduction

The formal reduction is an algorithm transforming the input system into a new system of simpler structure. During the formal reduction, we will compute hyperexponential elements that will later be used to construct formal solutions. The following operations are used by the formal reduction:

(R1) *Gauge transformations*: changes of variable $Y = TZ$ with $T \in \text{GL}(n, \bar{K}((\tau)))$ resulting in an *equivalent* system of the form

$$B = T^{-1}(A\phi T - \tau^{r-\omega}\delta T) =: T[A].$$

In order to distinguish (R1) from (R2), we assume that T is not a scalar multiple of the identity matrix.

(R2) *Term transformations*: these are transformations of the form

$$B = \text{term}_{\mu, \alpha, t}(A) = \alpha\tau^t(A - \mu I) \tag{5.1}$$

where $\mu, \alpha \in \bar{K}$ and $t \in \mathbb{N}$. A term transformation implements a change of variable $Y = hZ$ where h is hyperexponential over F , satisfying $\text{cert}(h) = \mu/\tau^{r-\omega}$.

(R3) Introduction of *Ramifications*: substituting

$$\tau \rightarrow \tau^s$$

with $s \in \mathbb{N}$, $s > 1$. During the formal reduction, algebraic extensions of the coefficient field $K((\tau))$ may become necessary. From a computational point of view, calculating with elements of the field $K((\tau^{1/s}))$ is not very convenient, and the operation (R3) reduces this to computing in $K((\tau))$ with different values for δ and ϕ .

Remark 5.2.2. *In the literature, so called “smart ramifications” of the form $\tau \rightarrow c\tau^s$ with $c \in \bar{K}$ have been introduced [10], which we use in our implementation as well, but in this chapter we omit describing this approach for sake of simplicity.*

We will now explain our approach for the formal reduction algorithm, however this will not be the most efficient approach. For now, we will only demonstrate that it is a constructive proof for the existence of formal solutions. In Section 5.3, we will then refine this in order to derive a more efficient algorithm.

Finding Gauge Transformations

We shall first see how suitable operations of the type (R1) can be found whenever the system is not Moser-irreducible or the leading matrix A_0 of the system has several eigenvalues, up to multiples of integer powers of q . The notion of Moser-irreducible forms is one of the key ingredients in the algorithmic formal reduction of linear systems of differential (and difference) systems, and as we showed in Chapter 3, the theoretical concepts discovered by Moser and their algorithmic applications can be extended to pseudo-linear systems as well.

Assuming that the system is Moser-irreducible, we can still carry out (R1) to further simplify the system, provided A_0 has several eigenvalues. This is achieved by the following lemma, where we use the notation $q^{\mathbb{Z}}$ for the set $\{q^d \mid d \in \mathbb{Z}\}$.

Lemma 5.2.2 (Splitting Lemma). *Given the system (2.31) with $r > 0$, assume that the leading matrix A_0 is block-diagonal*

$$A_0 = \begin{pmatrix} A_0^{11} & 0 \\ 0 & A_0^{22} \end{pmatrix}$$

such that

$$q^{\mathbb{Z}} \text{spec}(A_0^{11}) \cap \text{spec}(A_0^{22}) = \emptyset.$$

Then there exist an efficient algorithm computing $T \in \text{GL}(n, \mathcal{O})$ with $T_0 = I_n$, transforming

the system into a new system that is block-diagonal

$$T[A] = \begin{pmatrix} B^{11} & 0 \\ 0 & B^{22} \end{pmatrix}$$

with block sizes matching the block structure of A_0 .

We will postpone the proof of this proposition until Section 5.3.1, where we will state and prove a Generalised Splitting Lemma that comprises the above lemma as a particular case.

Introducing Ramifications

The next case that we consider is that of the system being Moser-irreducible and A_0 nilpotent. We will show that a combination of (R1) and (R3) leads to a new system with non-nilpotent leading matrix.

Lemma 5.2.3. *Any pseudo-linear system with coefficients in F is equivalent to a block-diagonal system whose individual blocks are companion matrices of the form*

$$\begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & 1 \\ c_0 & c_1 & \dots & c_{n-2} & c_{n-1} \end{pmatrix}, \quad (5.2)$$

by using an operation of the type (R1).

A proof of this lemma can be found in [75] or [36] for the field $K(x)$ and can be easily adapted the field $F = K((\tau))$.

Lemma 5.2.4 (Katz invariant). *Assume the system (2.39) is Moser-irreducible and its leading matrix A_0 is nilpotent. Then the system's Katz invariant $\kappa = p/s$ is a rational number and it exists a transformation that transforms the system into a new system with Poincaré-rank κ and non-nilpotent leading matrix.*

Proof Applying Lemma 5.2.3, we consider the different blocks individually. Let C be one of the individual system matrices in form of a companion matrix of the form (5.2). We compute the rational number

$$\kappa = \min \left\{ \frac{v(c_j)}{j - n - 1} \right\}$$

and carry out the diagonal transformation $Y = \tau^\epsilon Z$ where $\epsilon = \text{diag}(\epsilon_1, \dots, \epsilon_n)$ with $\epsilon_i = (1 - i)\kappa$. The new system matrix has valuation $-\kappa > \omega$. If $\kappa \leq -\omega$, the new system is not

irregular singular and we can stop. Otherwise we can see that the resulting system has a non-nilpotent leading matrix. Indeed, the new system matrix is $\tau^{-k}C_0$ where C_0 has non-zero elements in the upper off-diagonal and its bottom row has at least one non-zero entry – the non-zero terms come from the leading coefficients of the c_i for which the minimum in the formula above is obtained. \square

Carrying Out Term Transformations

We now consider the case where all the eigenvalues of A_0 are of the form $\mu = q^k \tilde{\mu}$ where $\tilde{\mu} \in \bar{K}$ fixed and $k \in \mathbb{Z}$ arbitrary. We recall that if $\phi \neq \text{id}$, we can assume $\mu \neq -1$ if $r = \omega$. We will construct an operation of the form (R2) in order to reduce all eigenvalues of A_0 to zero and possibly, lower the Poincaré-rank of the system.

Example 5.2.2. *In the differential case, this can easily be done using a change of variable*

$$Y = e_{r,\mu}(\tau)Z \tag{5.3}$$

where $e_{r,\mu}$ is a suitably defined exponential function: let $\delta = \frac{d}{d\tau}$, $\phi = \text{id}$ and $\omega = -1$. One has $q = 1$ and for

$$e_{r,\mu}(\tau) = \exp\left(-\frac{\mu}{r\tau^r}\right)$$

that $\delta(e_{r,\mu}) = \frac{\mu}{\tau^{r+1}}e_{r,\mu}$. The change of variable (5.3) gives a new system $\tau^{r+1}\delta Z = \tilde{A}\phi Z$ where the system matrix is

$$\tilde{A} = A - \tau^{r+1} \frac{\delta(e_{r,\mu})}{e_{r,\mu}} I = A - \mu I.$$

Using the Splitting Lemma, we can assume that $\mu \neq 0$ is the only eigenvalue of A_0 . The leading matrix $\tilde{A}_0 = A_0 - \mu I$ of the above matrix will be nilpotent, and we could apply the formal reduction recursively.

In the general case of arbitrary pseudo-linear systems, several problems arise if we wanted to use a similar approach as for differential systems, using the change of variable (5.3): we might not always be able to find a closed-form expression for $e_{r,\mu}$. Even if we did for some systems, it might not always be unique, and could also lead to intermediate expression swell due to symbolic simplifications (for example, in the case of difference systems, one would have to manipulate expressions containing the Γ -function).

A solution to this problem is to apply operation (R2), which effectively implements the change of variable (5.3) without explicitly knowing $e_{r,\mu}$. We will now explain the approach, where the following definition will be helpful:

Definition 5.2.4. *In the context of the change of variable (5.3), assume that $e_{r,\mu}$ is hyperexponential over $K((\tau))$ with $\text{cert}(h) = \mu/\tau^{r-\omega} + O(\tau^{r-\omega+1})$. Then we call $e_{r,\mu}$ a resolving irregular part.*

Carrying out the change of variable (5.3) with $e_{r,\mu}$ a resolving irregular part, we obtain the new system

$$\tau^{r-\omega}\delta Y = \left(\frac{\phi(e_{r,\mu})}{e_{r,\mu}} A - \tau^{r-\omega} \frac{\delta(e_{r,\mu})}{e_{r,\mu}} I \right) \phi Y. \quad (5.4)$$

From Lemma 5.2.5 (iii) and (iv) we find with $\alpha := \text{lc}(\phi(e_{r,\mu})/e_{r,\mu})$ and $t := v(\phi(e_{r,\mu})/e_{r,\mu})$ that

$$\frac{\phi(e_{r,\mu})}{e_{r,\mu}} = \alpha\tau^t + O(\tau^{t+1}) \quad (5.5)$$

and

$$\frac{\tau^{r-\omega}\delta(e_{r,\mu})}{e_{r,\mu}} = \alpha\mu\tau^t + O(\tau^{t+1}). \quad (5.6)$$

Hence, the new system has Poincaré-rank $r-t \leq r$ and its leading matrix has 0 as eigenvalue of multiplicity m . The hyperexponential parts of the new system are exactly those of the original system with the term $\mu/\tau^{r-\omega}$ removed, as desired.

Comparing (5.5) and (5.6) with the definition of a term transformation

$$\text{term}_{\mu,\alpha,t}(A) = \alpha\tau^t(A - \mu I) \quad (5.7)$$

where $\mu, \alpha \in \bar{K}$ and $t \in \mathbb{N}^+$, we can see that a term transformation implements a change of resolving irregular part $Y = e_{r,\mu}Z$ where the terms of valuation $> t$ in (5.5) and (5.6) vanish. We call $e_{r,\mu}$ a *canonical* resolving irregular part in this case. Let us define the *lift* and the *scale* of a term transformation (a resolving irregular part resp.) as the values of t and α in (5.7) (and as in (5.5) and (5.6) resp.). We do need to know what the possible values for the lift and the scale of a term transformation are, before we can apply it. The following lemma prepares an answer to this question.

Lemma 5.2.5. *Let h be hyperexponential over (F, ϕ, δ) and assume that $v(h/\phi(h)) \leq 0$. If $v(\text{cert}(h)) = 0$, also assume $\text{lc}(\text{cert}(h)) \neq -1$. Then the following holds: if $\phi = \text{id}$, then $v(\phi(h)/h) = 0$ and $\text{lc}(\phi(h)/h) = 1$. Otherwise, $\omega \geq 0$ and:*

- (i) *If $v(\text{cert}(h)) < 0$, then $\text{lc}(\phi(h)/h) = 1/\text{lc}(\text{cert}(h))$ and $v(\phi(h)/h) = -v(\text{cert}(h))$.*
- (ii) *If $v(\text{cert}(h)) = 0$, then $\text{lc}(\phi(h)/h) = 1/(\text{lc}(\text{cert}(h)) + 1)$ and $v(\phi(h)/h) = 0$.*
- (iii) *If $v(\text{cert}(h)) > 0$, then $\text{lc}(\phi(h)/h) = 1$ and $v(\phi(h)/h) = 0$.*

Proof If $\phi = \text{id}$, then $\phi(h)/h = 1$ from which we immediately obtain the claim of the lemma in this case. Now assume $\phi \neq \text{id}$. Then $\delta = \text{id} - \phi$ and hence by definition $\omega =$

$v(\tau - \phi(\tau)) - 1 \geq 0$ since $v(\phi(\tau)) = 1$. From the definition of the certificate, we obtain

$$\frac{h}{\phi(h)} - 1 = \text{cert}(h). \quad (5.8)$$

We now address the individual cases. In each case, we will use the fact that both valuations and leading coefficients of the LHS and the RHS of (5.8) need to coincide.

- (i) Case $v(\text{cert}(h)) < 0$. This implies that the valuation of the LHS of (5.8) needs to be < 0 , $v(h/\phi(h)) = v(\text{cert}(h))$ and $\text{lc}(h/\phi(h)) = \text{lc}(\text{cert}(h))$. From this, the claim follows.
- (ii) Case $v(\text{cert}(h)) = 0$. Here, we need $v(h/\phi(h)) = 0$ and $\text{lc}(h/\phi(h)) - 1 = \text{lc}(\text{cert}(h))$. Solving for the leading coefficient in this equation shows the desired properties.
- (iii) Case $v(\text{cert}(h)) > 0$. This is only possible if leading terms in the LHS of (5.8) cancel, which requires $v(\phi(h)/h) = 0$ and $\text{lc}(\phi(h)/h) = 1$.

□

A direct consequence of the previous lemma is that it clarifies what the possible values for α and t are.

Corollary 5.2.1. *Let $\mu \neq 0$ be an eigenvalue of A_0 , and if $r = \omega$, also assume $\mu \neq -1$. For any term transformation $\text{term}_{\mu,\alpha,t}$ the following holds: if $\phi = \text{id}$, then $\alpha = 1$ and $t = 0$. Otherwise, $\omega \geq 0$ and:*

- (i) *If $r > \omega$, then $\alpha = 1/\mu$ and $t = r - \omega$.*
- (ii) *If $r = \omega$, then $\alpha = 1/(\mu + 1)$ and $t = 0$.*
- (iii) *If $r < \omega$, then $\alpha = 1$ and $t = 0$.*

This proposition being used during the formal reduction describes the behaviour of the generic algorithm, in terms of "jumps" of the slopes of the Newton polygon. Both the difference and q -difference case make use of (i) and (ii), whereas situation (iii) only appears in the difference case.

5.2.3 Main Result

Motivated by a similar consideration for linear differential systems [63, Section 5.3], we define an equivalence relation on hyperexponential parts of a pseudo-linear system.

Definition 5.2.5. *Let h_1 and h_2 be two hyperexponential parts of the system (2.39). We say that h_1 and h_2 are equivalent ($h_1 \sim h_2$) if they have the same ramification index s and it holds $h_1 - h_2 \in \frac{1}{s}\mathbb{Z}\tau^\omega$ if $q = 1$, $h_1 - h_2 \in q^{\frac{1}{s}}\mathbb{Z}\tau^\omega$ otherwise.*

Theorem 5.2.1. *Given a singular pseudo-linear system of the form (2.39), there exists an algorithm using the operations (R1) – (R3) that computes the hyperexponential parts of all formal solutions of the system, up to the equivalence relation \sim .*

Proof The proof is obtained by following the formal reduction algorithm outlined in Sections 5.2.2, 5.2.2 and 5.2.2. We expand on this in more detail.

Applying the Moser-reduction algorithm, one computes a transformation of the type (R1) which leads to a system with minimal Poincaré-rank \tilde{r} . If the singularity is regular, we have $\tilde{r} = 0$. This is a system of the first kind, and for all formal solutions y we have $\text{irr}(y) = 0$. In [9] we have shown how the Splitting Lemma can then be used to block-diagonalise the system, by still using operations of the form (R1). The resulting individual systems all have regular parts associated with the roots of the system's *indicial equation* $\varphi(\lambda)$, or alternatively, the eigenvalues of A_0 . The corresponding hyperexponential parts are then of the form $\mu\tau^\omega$. Furthermore, all the differences of eigenvalues are elements of \mathbb{Z} if $q = 1$, and of $q^{\mathbb{Z}}$ otherwise. Note that, as in the differential case, the roots of φ are not invariant with respect to the operation (R1) – they can vary by integer differences, and this variation depends on the precise gauge transformation that has been computed by the reduction algorithms. Hence, the system's hyperexponential parts are only computed modulo the equivalence relation \sim .

Otherwise, if $\tilde{r} > 0$, we apply the Splitting Lemma in order to compute a transformation of the type (R1) that splits the system into one or several systems whose leading matrices have only one eigenvalue. If a Moser-irreducible system has a nilpotent leading matrix, one determines the Katz-invariant $\kappa = p/s$, followed by the operation (R3) which introduces the ramification s in the system. This might need choosing a different automorphism ϕ (and the also δ).

The case where A_0 has one unique eigenvalue $\mu \neq 0$ remains; let μ be such an eigenvalue of multiplicity m . We apply the term-transformation $\text{term}(\mu, \alpha, t)$ which may decrease the Poincaré-rank, and in any case produces a new system with nilpotent leading matrix. The number of different leading coefficients of hyperexponential parts that can be found is restricted by the number of roots of φ (counting multiplicities and the eigenvalue ∞), which is bounded by the dimension of the system n . Hence, we can re-iterate the process and will eventually terminate with one or several systems with Poincaré-rank zero. \square

Recall that the output of the formal reduction is a block-diagonal system, where the individual systems are of the form

$$\tau^{-\omega}\delta Y = A\phi Y. \tag{5.9}$$

Hence, each of these systems has a Poincaré-rank $r = 0$, so is a system of the first kind. Furthermore, the indicial polynomial as defined in [9][Section 3] has only integer roots.

An immediate consequence of this result is the following theorem which shows how to compute, under certain hypotheses, the valuations and leading coefficients of some hyperexponential parts. We state this for reasons of completeness, as it is the pseudo-differential version of Theorem 2.2 in [63].

Theorem 5.2.2. *Consider the system (2.39) with Poincaré-rank r and leading matrix A_0 . Let μ be an eigenvalue of A_0 with multiplicity m . In addition, if $r = \omega \geq 0$, assume that $\mu \neq -1$. Then there exist m formal solutions whose hyperexponential parts are of the form*

$$\text{hyp}(y) = \frac{\mu}{\tau^{r-\omega}} + \dots$$

where the dots stand for terms of higher valuation.

5.3 Formal Reduction Algorithm

The goal of this section is to give an algorithm that computes formal solutions in a more efficient way than it would be done using the method from Section 5.2. This method is inspired by [63, Section 5] for linear differential systems.

Let us summarise the formal reduction as we have developed it so far, by giving a description using pseudo-code. The main algorithm `Formal_Reduction` calls itself recursively, following the constructive proof of Theorem 5.2.1. The functions `Moser_Reduce`, `Splitting_Lemma` and `Katz_Invariant` are toolboxes implementing the concepts introduced in Section 5.2.2, which we shall not detail here. Term transformations are implemented using the function $\text{term}_{\mu,\alpha,t}(A)$.

Algorithm `Formal_Reduction`(A, ϕ, δ, ω)

Input: $A \in \mathcal{M}_n(F_s)$, δ a pseudo-derivation w.r.t. ϕ , $\omega \in \mathbb{Z}$.

Output: A list of pairs (h_i, \tilde{A}_i) , h_i a hyperexponential part, $v(\tilde{A}_i) \geq \omega$.

1. **if** $v(A) \geq \omega$ **and** $\varphi(\lambda)$ has only integer roots **then return** $[0, A]$;
2. $r := \omega - v(A)$;
3. **if** A is not Moser-irreducible **then**
 - // Compute a Moser-irreducible form of A
 - (a) **return** `Formal_Reduction`(`Moser_Reduction`(A, ϕ, δ, ω));
4. **if** A_0 has several eigenvalues **then**

// We use the Splitting Lemma to separate the eigenvalues of A_0 , up to the multiplicative factor q^k where $k \in \mathbb{Z}$.

(a) $(A^{(1)}, \dots, A^{(k)}) := \text{Splitting_Lemma}(A, \phi, \delta, \omega)$;

(b) **return** $\bigcup_i \text{Formal_Reduction}(A^{(i)}, \phi, \delta, \omega)$;

5. **else if** A_0 is nilpotent **then**

(a) $\kappa = p/s := \text{Katz_Invariant}(A, \phi, \delta, \omega)$;

(b) **return** $\text{Formal_Reduction}(A(\tau^s), \tilde{\phi}, \tilde{\delta}, \tilde{\omega})$;

6. **else**

// A_0 has a unique eigenvalue $\mu \neq 0$

(a) **if** $r = \omega$ **and** $\mu = -1$ **then return** \emptyset

(b) $B := \text{term}_{\mu, \alpha, t}(A)$;

(c) **return** $\{[\frac{\mu}{\tau^{r-\omega}} + h, C] \mid [h, C] \in \text{Formal_Reduction}(B, \phi, \delta, \omega)\}$;

The main drawback of this method is that a splitting of the system can only be achieved provided A_0 is non-nilpotent. However, in many situations, when comparing with the Newton-polygon method in the scalar case, it becomes clear that there should be such a splitting. This is essentially the same problem discussed in [63] in the differential case. Furthermore, the method for computing the Katz-invariant is based on converting the system to one (or several) scalar equations, which can be a costly process. Also, even if such a conversion would have been carried out efficiently, one would rather continue with the scalar algorithm. The method given by Lemma 5.2.4 effectively discards the scalar equation and continues with a system.

5.3.1 A Generalised Splitting Lemma

Before we show how to address the problems encountered in the previous section, let us first generalise our notion of simple systems in Section 4.2.1 to systems of a more general form which we call *k-simple*. For $k = 0$, they coincide with simple systems.

Definition 5.3.1. *A system of the form*

$$D\tau^{k-\omega}\delta Y = N\phi Y \tag{5.10}$$

where $D, N \in \mathcal{M}_n(\mathcal{O})$, D invertible and $k \in \mathbb{N}$ is called *k-simple* if the associated polynomial $\theta_k(\lambda) = \det(N_0 - \lambda D_0)$ satisfies $\theta_k(\lambda) \not\equiv 0$.

Using a change of unknown of the form $Y = TZ$ and left-multiplication by S where $S, T \in \text{GL}(n, \mathcal{O})$ and S_0, T_0 non-singular, a k -simple system can be transformed into an equivalent new k -simple system

$$\tilde{D}\tau^{k-\omega}\delta Y = \tilde{N}\phi Y$$

where $\tilde{D} = SDT$ and $\tilde{N} = S(N\phi(T) - D\tau^{k-\omega}\delta(T))$.

We will now give a decomposition lemma that extends the Generalised Splitting Lemma (Proposition 3.1 in [63]) known for k -simple linear differential systems to k -simple pseudo-linear systems. As for a simple system ($k = 0$), we call $D_0\lambda - N_0$ the *leading pencil* of the k -simple system (5.10).

Proposition 5.3.1 (Generalised Splitting Lemma). *Consider a k -simple operator $D\tau^{k-\omega}\delta - N\phi$ with $k > 0$ and assume that its leading pencil is block-diagonal*

$$D_0\lambda - N_0 = \begin{pmatrix} D_0^{11}\lambda - N_0^{11} & 0 \\ 0 & D_0^{22}\lambda - N_0^{22} \end{pmatrix}$$

with

$$q^{\mathbb{Z}}\text{spec}(D_0^{11}\lambda - N_0^{11}) \cap \text{spec}(D_0^{22}\lambda - N_0^{22}) = \emptyset.$$

Then there exist $S, T \in \text{GL}(n, \mathcal{O})$ with $S_0 = T_0 = I_n$, transforming the operator into a new k -simple operator that is block-diagonal with block sizes matching the block structure of its leading pencil.

Proof The proof is identical to the one of Proposition 4.1 in [9] for the case $k = 0$ apart from the final matrix equations that determine the individual coefficients of transformation matrices. These equations are of the form $X(A + \lambda B) - (C + \lambda D)Y = E + \lambda F$. It is well known that a unique solution of this equation exists provided that $A + \lambda B$ and $C + \lambda D$ are regular matrix pencils and do not have any eigenvalues in common. From the assumptions of the proposition, these conditions are met and we can hence find unique coefficients for the transformation matrices S and T . \square

The following theorem generalises [63, Theorem 4.2] from linear differential systems to pseudo-linear systems.

Theorem 5.3.1. *Consider the system (2.39) with Poincaré-rank r and assume that it can be written as a k -simple system with associated polynomial θ_k that has a root $0 \neq \mu \in \bar{K}$ of multiplicity $m > 0$. Then there exist m formal solutions whose hyperexponential parts are of the form*

$$\text{hyp}(y) = \frac{\mu}{\tau^{k-\omega}} + \dots$$

where the dots stand for terms of higher valuation.

Proof The proof of this theorem relies on the Generalised Splitting Lemma. Using a constant transformation with matrices $S, T \in GL(n, \bar{k})$, we can find an equivalent k -simple system with leading pencil $\tilde{D}_0\lambda - \tilde{N}_0$ which is block-diagonal, satisfying the condition on the spectrum as in Proposition 5.3.1. Furthermore, denote $\theta_k^{11}(\lambda) = \det(N_0^{11} - D_0^{11})$. We then have D_0^{11} invertible, $\deg(\theta_k^{11}) = m$ and μ is its only root. Applying the Generalised Splitting Lemma, we obtain a block-diagonal k -simple system. The first block can be rewritten as a pseudo-linear system of the form (2.31) with Poincaré-rank $r = k$ since D_0^{11} is invertible. Theorem 5.2.2 then shows the theorem, together with the fact that the second block has unchanged Poincaré-rank and a constant associated Moser-polynomial. \square

The crucial fact is that it is possible to rewrite our input system (2.39) as a k -simple system for appropriate values of k . This approach is based on using the concept of *super-irreducible* forms of pseudo-linear systems as introduced in Section 3.4.

This has two applications for the formal reduction. Using Theorem 5.3.1 in combination with the super-reduction algorithm, one can compute all resolving irregular parts that are associated with integer slopes of the system's Newton polygon. After a change of variable, the algorithm proceeds recursively and this leads to a Newton polygon iteration similar to the scalar case. We shall not give more details here but refer to [60] and [63] for a presentation in the case of linear differential systems.

The second application is to use the Generalised Splitting Lemma as part of the formal reduction algorithm in order to achieve a splitting of the system in some cases when the classical Splitting Lemma is not applicable (due to A_0 being nilpotent). This is further explained in [63] in the differential case and works very similarly in the pseudo-linear case. We will describe the resulting algorithm in the following section.

5.3.2 Nilpotent Case

This case presents itself if after having applied the Moser-reduction algorithm, the leading matrix A_0 is still nilpotent. A splitting of the system is not possible by using the classical Splitting Lemma alone. We now distinguish two cases, depending on the degree of the associated Moser-polynomial θ . The function `Generalised_Splitting_Lemma` implements Proposition 5.3.1.

Case of Non-Constant Moser-Polynomial

We are able to apply the Generalised Splitting Lemma if $\deg(\theta) > 0$. This gives two individual systems, the first of which will only possess ramified formal solutions, and the second

will have a lower Poincaré-rank. We can hence apply recursion on this system.

Nilpotent_Case(A, δ, ϕ, ω)

Input: $A \in \mathcal{M}_n(F_s)$, δ a pseudo-derivation w.r.t. ϕ with degree ω , A Moser-irreducible and A_0 nilpotent.

Output: Pairs $[h_i, \tilde{A}_i]$, h_i a hyperexponential part, $v(\tilde{A}_i) \geq \omega$.

// θ is the Moser-polynomial associated with the system.

1. $d := \deg(\theta)$;

2. **if** $d > 0$ **then**

// Apply the Generalised Splitting Lemma to compute a new system $B = \text{diag}(B^{11}, B^{22})$
 where B_0^{11} is nilpotent and B^{22} is of dimension d with lowered Poincaré-rank

(a) $(B^{11}, B^{22}) := \text{Generalised_Splitting_Lemma}(A, \delta, \phi, \omega)$;

(b) **return** $\text{Ramified_Case}(B_{11}, \delta, \phi, \omega) \cup \text{Formal_Reduction}(B_{22}, \delta, \phi, \omega)$;

3. **else return** $\text{Ramified_Case}(A, \delta, \phi, \omega)$;

In the case where A_0 is nilpotent and $d = \deg(\theta)$, the degree of the Moser-polynomial, is zero, we are not able to further split the system, and call `Ramified_Case` straight away.

Case where Splitting is not Possible

In order to compute the Katz-invariant and all other ramified slopes that are greater than $r-1$, the function `Ramified_Case` pursues several strategies, similar to those explained in [63] in the differential case: for systems of reasonably small dimension, simply trying successive integers s as ramifications will be efficient enough to be used in practice. A ramification will be detected if, after carrying out the operation (R3) followed by a Moser reduction, leading matrix of the new system is not nilpotent. Also, computing the characteristic polynomial of a truncation with a small number of coefficients of A and using the maximal slope of its algebraic Newton polygon is conjectured to always yield the Katz-invariant in the differential case and it seems usable in our setting, as well. Finally, let us remark that the idea underlying the methods in [14, 17, 18] for computing the Katz-invariant in the differential and difference case could be generalised to pseudo-linear systems.

5.3.3 Discussion

When applying our generic formal reduction algorithm to the differential case, it coincides with the method in [63]. This was in fact the starting point of our work. In this section, we will illustrate how our algorithm proceeds in the cases of linear difference and q -difference systems, and highlight the improvements made compared to previously known methods.

The Algorithm for Difference Systems

For linear difference systems, our algorithm `Formal_Reduction` unifies and improves the method that is obtained by combining [17] and [18]. In [17], the system is assumed to be of the form (2.32) whereas the algorithm in [18] needs the input in the form (2.33). One benefit of our approach is that there is no need for a conversion between these different formats. Neither we need different versions of the Moser-reduction, as we use a generic Moser-reduction for systems of the form (2.39). The key mechanism to implement this is to use Proposition 5.2.1 throughout the algorithm. This can be interpreted as computing the “jumps” that occur in the change of Newton-polygon slopes. In order to illustrate this further, we give a high level description of the resulting specialisation of our generic algorithm.

Recall that for a system with coefficients in $K((\tau))$, we initially have $\omega = 1$. The first reduction stage consists in calling `Moser_Reduction` in order to find a system with either $r = 0$ (in which case the system is regular singular and the algorithm stops), or with $r > 0$ and a non-nilpotent leading matrix by using `Nilpotent_Case` and possibly `Ramified_Case`. If the introduction of new ramifications was necessary, we have $\omega = s > 1$. After applying the Splitting Lemma we have that A_0 has one single eigenvalue $\mu \neq 0$. There are now four particular situations to consider:

1. Case 1: $r > s$. In this case, a term-transformation with lift $t = r - s > 0$ and scale $\alpha = \mu^{-1}$ will lead to a new system with $\tilde{r} \leq s$ and \tilde{A}_0 nilpotent if $\tilde{r} = s$. A system of this form is called *normal* in [17] and can be characterised by a specific structure of the irregular parts of the formal solutions: they only contain functions of the form as in the case $0 < r < s$ in Proposition 5.5.2. After applying the term-transformation, recursive application of the algorithm will either result in a system with Poincaré-rank $r = 0$, in which case we stop, or eventually (after introduction of a ramification) result in a non-nilpotent leading matrix and $0 < r < s$.
2. Case 2: $r = s$ and $\mu \neq -1$. We carry out a term transformation with lift $t = 0$ and scale $\alpha = 1/(1 + \mu)$. The resulting system is normal and the algorithm proceeds as after the term-transformation in Case 1.
3. Case 3: $r < s$. A ramification must have been introduced previously since $0 < r < s$. The algorithm invokes a sequence of recursive calls, each involving a term transformation with lift zero. During each call, using the first reduction stage, one eventually reaches a non-nilpotent leading matrix (after possibly introducing additional ramifications) and the algorithm remains in Case 4. This computes the so-called *exponential matrix* of the system’s FFMS, with the same improvements reported in [63] for the differential case, compared to the method in [18].

Example 5.3.1. Consider the matrix

$$A = \begin{pmatrix} 2 & 2 \\ 2\tau & 2 \end{pmatrix}$$

and the system $\tau\delta Y = A\phi Y$ with $\phi(\tau) = \frac{\tau}{1+\tau}$ and $\delta = id - \phi$. We have $\omega = 1$ in this case, and hence $r = 2$. The leading matrix has the only eigenvalue $\mu = 2$. Since $s = 1$, we have $r > s$ and the first reduction stage is already completed. The algorithm now continues with Case 1: a term-transformation with lift $t = 1$ and scale $\alpha = 1/2$ is applied. The algorithm computes the hyperexponential parts of the system in the following way: whenever a term-transformation $\text{term}_{\mu,\alpha,t}$ is applied to a system with Poincaré-rank r , the monomial $\mu/\tau^{r-\omega}$ of a hyperexponential part is computed. In our example, the term-transformation is $\text{term}_{2,1/2,1}$ and the resulting hyperexponential part is

$$w = \frac{2}{\tau}.$$

This term-transformation leads to the new system $\delta Y = B\phi Y$ with system matrix

$$B = \begin{pmatrix} 0 & \tau \\ \tau^2 & 0 \end{pmatrix}.$$

For this new system, we have $r = 1$ and B_0 is nilpotent. The algorithm now proceeds with Case 3. One can show that the system is Moser-irreducible, and that a call to `Ramified_Case` is needed. For this example, the only choice is the ramification $s = 2$, yielding the system

$$\tilde{\delta} Y = \begin{pmatrix} 0 & 1 \\ \tau^2 & 0 \end{pmatrix} \tilde{\theta} Y = C\tilde{\theta} Y$$

and $\tilde{\theta} = \frac{\tau}{\sqrt{1+\tau^2}}$, $\tilde{\delta} = id - \tilde{\phi}$. The algorithm now enters the first reduction stage again, and the Moser-reduction algorithm computes the gauge transformation $T = \text{diag}(1, \tau)$ such that

$$T[C] = \begin{pmatrix} 0 & \frac{\tau}{\sqrt{1+\tau^2}} \\ \tau & \frac{\tau}{\sqrt{1+\tau^2}} - \tau \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tau + O(\tau^2)$$

has an invertible leading matrix with eigenvalues $\mu \in \{-1, 1\}$. Furthermore, the new Poincaré-rank is $r = 1$. Applying the Splitting Lemma will result in two systems of dimension $n = 1$. For each of those systems, the algorithm will continue to apply Case 4 and an additional term-transformation will finally result in systems of the first-kind, and the formal reduction terminates. From the second term-transformation $\text{term}_{1,1,0}$, the system having a Poincaré-rank 1, and taking into account the ramification in the previously computed

hyperexponential part, we now obtain

$$w = \frac{2}{\tau^2} + \frac{1}{\tau}.$$

Note that introducing the ramification prevents the need for computations in $\bar{K}((\tau^{1/s}))$.

Algorithm for q -Difference Systems

When dealing with an irregular singular linear q -differential system, the algorithm Formal_Reduction in Section 5.3 simplifies considerably since after the first reduction stage, in Step 6 (c), the recursive call is not needed. This is due to the relatively simple structure of the irregular part of the formal solutions, as already reported in [31]. Nevertheless, we have not found a published algorithm for this case, and hence the method that results from our approach seems new.

In the q -difference case, we have $\omega = 0$. Assume that $r > 0$. Proposition 5.2.1 then shows that for each eigenvalue $\mu \neq 0$ (with multiplicity m) of the system's leading matrix A_0 , a term transformation with lift $t = r$ and scale $\alpha = 1/\mu$ can be found. Consequently, the new transformed system has Poincaré-rank $\tilde{r} = 0$, and we are done. The formal reduction therefore consists of two different steps:

1. This is the first reduction stage which transforms the input system into a block-diagonal system with each individual system having either $r = 0$, or $r > 0$ and a non-nilpotent leading matrix A_0 with single eigenvalue μ . This will be achieved as discussed earlier, involving Moser-reduction, the Generalised Splitting Lemma and the approach explained in Section 5.3.2 for computing the Katz-invariant.
2. For each of the individual systems having $r > 0$ and $\mu \neq 0$ an eigenvalue of its leading matrix A_0 , carry out a term-transformation with lift $t = r$ and $\alpha = 1/\mu$. The resulting system will have Poincaré-rank $r = 0$ (this is also called a *system of the first kind*) and is hence regular singular. One can show that its indicial polynomial $P(\lambda)$ has only integer roots. Note that we will have $\deg(P) = m \leq n$, a phenomenon which can only arise in the q -difference case for a system of the first kind (see [9] for a definition of P and more details).

In some situations, the algorithm simplifies even further. If the formal solutions do not contain any ramifications, we can compute them without any splitting of the system. The example below illustrates such a case.

Example 5.3.2. Consider the system $\tau^2 \delta Y = A \phi Y$ where $\phi \tau = q \tau$, ($q \neq 1$), $\delta = id - \phi$ and

$$A = \begin{pmatrix} 1 - q^{-2} & \tau^2 \\ \tau & \tau \end{pmatrix}.$$

This system is irregular singular with Poincaré-rank $r = 2$. Its leading matrix A_0 is non-nilpotent, and its non-zero eigenvalue is $\mu = 1 - q^{-2}$. This gives the hyperexponential part

$$w_1 = \frac{1 - q^{-2}}{\tau^2}.$$

Note that the system is Moser-irreducible. Its associated Moser-polynomial is

$$\theta(\lambda) = (1 - q^{-2})(1 - \lambda).$$

We can find the second hyperexponential part from the non-zero root of $\theta(\lambda)$:

$$w_2 = \frac{1}{\tau}.$$

For both hyperexponential parts, we can carry out a term transformation that will lead to new systems of the first kind. We detail this for w_1 : the term transformation $\text{term}_{1-q^{-2}, 1/(1-q^{-2}), 2}$ leads to the system

$$\tilde{A} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} + O(\tau).$$

Up to a constant factor, its indicial polynomial is

$$\varphi(\lambda) = \det(q^\lambda A_0 - (q^\lambda - 1)I) = 1 - q^\lambda.$$

Its degree is less than $n = 2$ and $\mu = 0$ is its only root.

5.4 Computing Formal Solutions

In this section, we will see how to compute all formal solutions of (2.39) from the output of the formal reduction of the system, provided we have explicit formulae for resolving irregular parts (see Section 5.5 for more details on these). We will start with determining the irregular part $\text{irr}(y)$ and regular part $\text{reg}(y)$ of all formal solutions y from the hyperexponential parts that are returned by the formal reduction.

Our implementation of the formal reduction follows the approach in [63, Section 5.3] where we used a variant of the Splitting Lemma (Generalised Splitting Lemma resp.) to block-reduce a system rather than block-diagonalise it. Our algorithm computes a list containing all the hyperexponential parts, their multiplicities and ramifications, as well as corresponding transformation matrices. Each entry of the list is a triple (w, s, T) such that the change of variable $Y = e^*Z$ where e^* satisfies $\text{cert}(e^*) = w$, followed by the substitution

$\tau \rightarrow \tau^s$ and transforming with T , gives a new system matrix

$$B = \begin{pmatrix} B^{11} & 0 \\ B^{21} & B^{22} \end{pmatrix}$$

where $B^{22} = B_0^{22}\tau^\omega + \dots$ with $\text{spec}(B_0^{22}) \subset \mathbb{Z}$ if $q = 1$, and $\text{spec}(B_0^{22}) \subset q^{\mathbb{Z}}$ otherwise. The dimension of B_{22} is the multiplicity of w . Note that one might have $B_0^{22} = 0$.

Using block-reductions simplifies the computation of the transformation matrix T and could be seen as a local factorisation of the input system. For each of the formal solutions of the system $\tau^{-\omega}\delta(Z) = B_{22}\phi(Z)$, one can easily construct a formal solution of the original input system.

5.4.1 The Method

The idea is to determine, for each of the different terms of w , corresponding resolving irregular parts and/or a regular part, following the explicit formulae given in Section 5.5. These are then multiplied together to form $\text{irr}(y)\text{reg}(y)$. Once this has been computed, an appropriate change of variable reduces the problem of computing formal solutions of the original system to that of computing regular solutions of the new system. The formal reduction has implicitly computed e^* with $\text{cert}(e^*) = w$ as a product of canonical resolving irregular parts. We now replace this with an explicit expression.

Note that if $\phi = \text{id}$, the problem is trivial as we have explicit formulae for canonical resolving irregular parts. Hence, assume $\phi \neq \text{id}$. From w , we compute \tilde{e} such that $\text{cert}(e^*) - \text{cert}(\tilde{e}) = o(\tau^\omega)$. This is done by using resolving irregular parts and also regular parts. If

$$w = \frac{a_1}{\tau^{r_1-\omega}} + \dots + \frac{a_l}{\tau^{r_l-\omega}}$$

where $a_i \in \bar{K}$ and $r_i \in \mathbb{Z}$ with $r_1 > \dots > r_l \geq 0$ ($i = 1, \dots, l$), we put

$$\text{irr}(y)\text{reg}(y) = \prod_i e_{k_i, a_i} =: e_y$$

where the e_{k_i, a_i} are resolving irregular parts for $k > 0$ and we have defined $e_{0, \mu} = e_\mu$ as being a regular part. Carrying out the change of variable $Y = e_y/e^*Z$ we replace e^* , for which we do not have an explicit expression in general, we obtain the new system

$$\tau^{r-\omega}\delta Z = \frac{\phi(e_y)}{e_y} \frac{e^*}{\phi(e^*)} A - \left(1 - \frac{\phi(e_y)}{e_y} \frac{e^*}{\phi(e^*)} - w \right) I.$$

using

$$\frac{e^*}{\phi(e^*)} = 1 + \text{cert}(e^*) = 1 + w$$

and

$$\frac{\phi(e_y)}{e_y} = \prod_i \frac{\phi(e_{k_i, a_i})}{e_{k_i, a_i}}.$$

The last task that remains is to compute the series parts, by computing regular formal solutions of a system of the first kind. As we have explained in the previous chapter, rather than applying an adaptation of Wasow's method [74] to pseudo-linear systems, this can be done more efficiently by extending the *monomial-by-monomial* method of [21].

Applying this method to the system

$$\tau^{-\omega} \delta Y = B_{22} \phi Y \tag{5.11}$$

yields regular solutions that are elements of $\bar{K}[[\tau]]^n[u]$. They can be written as

$$y_{ij} = \sum_{j=0}^{s_i} z_{ij} \frac{u^{s_i-j}}{(s_i-j)!} \quad (z_{ij} \in \mathcal{O}^n) \tag{5.12}$$

where u is the pseudo-differential logarithm defined as previously. The algorithm will then compute the different coefficients of the series expansion up to a desired order. This accomplishes the computation of the full set of formal solutions.

5.5 Resolving Irregular Parts of Some Linear Functional Systems

In this section, we will present explicit formulae for resolving irregular parts, for the three types of pseudo-linear systems we have listed in Chapter 2. First, we discuss the differential case with $\phi = \text{id}$ and the derivation $\delta = \frac{d}{dx}$. We then look at difference systems given by the shift automorphism $\phi(\tau) = \frac{\tau}{\tau+1}$. Finally, we consider the automorphism $\phi(\tau) = q\tau$. For all three cases, the computation of formal solutions is then possible as explained in Section 5.4.

5.5.1 Linear Differential Systems

A singular local linear differential system is a system of the form

$$x^{r+1} Y' = A(x) Y \tag{5.13}$$

with $r \geq 0$ where $A \in \mathcal{M}_n(K((x)))$ for a singularity at $x = 0$. This can be represented as a system of pseudo-linear equations of the form (2.31) with $\omega = -1$, $\phi = \text{id}$ and $\delta = \frac{d}{d\tau}$. This system has a formal fundamental matrix solution of the form

$$Y(\tau) = H(\tau)\tau^\Lambda e^{Q(\tau)} \quad (5.14)$$

where $H \in \text{GL}(n, \bar{K}[[\tau]])$, $\Lambda \in \mathcal{M}_n(\bar{K})$,

$$Q(\tau) = \text{diag}(q_1(\tau), \dots, q_n(\tau))$$

and the q_i are polynomials in τ^{-1/s_i} without constant term for convenient positive integers s_i . The presence of a non-trivial block in the Jordan normal form of Λ causes the appearance of logarithmic terms in the formal solutions.

Proposition 5.5.1. *Given the system (5.13) with Poincaré-rank r and leading matrix A_0 with an eigenvalue $\mu \neq 0$, define $e_{r,\mu} = \exp\left(-\frac{\mu}{\tau^r}\right)$. Then $e_{r,\mu}$ is a canonical resolving irregular part of the system with lift 0 and scale 0.*

Proof Since $\phi = \text{id}$, we have $\phi(e_{r,\mu})/e_{r,\mu} = 1$ and since $\frac{d}{d\tau}e_{r,\mu} = e_{r,\mu}\frac{\mu}{\tau^{r+1}}$, we find $\tau^{r-\omega}\delta(e_{r,\mu})/e_{r,\mu} = \mu$. Hence we have $\alpha = 1$ and $t = 0$ as in 5.5 and 5.6, which shows that $e_{r,\mu}$ is a canonical resolving irregular part. \square

5.5.2 Linear Difference Systems

We now address systems of linear difference equations. We will give resolving irregular parts for systems of the form

$$\tau^{r-1}(Y(\tau) - Y(\tau/(\tau + 1))) = A(\tau)Y(\tau/(\tau + 1)) \quad (5.15)$$

with $A \in \mathcal{M}_n(F)$. This is the difference system (ii) as introduced in Chapter 2. It is a local pseudo-linear system with $\omega = 1$, $\phi(\tau) = \frac{\tau}{1+\tau}$ and $\delta = \text{id} - \phi$. In order to expose the full structure of irregular parts, we need to consider (5.15) over the coefficient field $K((\tau^{1/s}))$ ($s > 1$). Using the operation (R3), this can be reduced to a system over the initial field $K((\tau))$ with new automorphism $\phi(\tau) = \frac{\tau}{\sqrt[2]{1+\tau^s}}$. We then have $\omega = s$.

Systems of this form admit a formal fundamental solution [70]

$$Y(\tau) = H(\tau)\tau^\Lambda e^{Q(\tau)}\tau^{R\tau} \quad (5.16)$$

where $H \in \text{GL}(n, \bar{K}[[\tau]])$, $\Lambda \in \mathcal{M}_n(\bar{K})$, R is a diagonal matrix with rational number entries and

$$Q(\tau) = \text{diag}(q_1(\tau), \dots, q_n(\tau))$$

where the q_i are polynomials in τ^{-1/s_i} without constant term.

The structure of resolving irregular parts in this case depends on the precise values of r and s .

Proposition 5.5.2. *Given the system (5.15) with Poincaré-rank r and leading matrix A_0 with eigenvalue $\mu \neq 0$ and additionally, if $r = s$, $\mu \neq -1$. Then*

$$e_{r,\mu} = \begin{cases} \mu^{-\tau^{-s}} \Gamma(\tau^{-s})^{1-r/s} & \text{if } r > s, \\ (\mu + 1)^{-\tau^{-s}} & \text{if } r = s, \\ \exp(-\frac{s\mu}{r} \tau^{-r}) & \text{if } 0 < r < s \end{cases} \quad (5.17)$$

is a resolving irregular part. Furthermore, we have for its lift t and scale α

$$(t, \alpha) = \begin{cases} (r - s, \mu^{-1}) & \text{if } r > s, \\ (0, (1 + \mu)^{-1}) & \text{if } r = s, \\ (0, 1) & \text{if } 0 < r < s. \end{cases}$$

Proof Assume first $r \geq s$ and denote $\tilde{e}_{r,a} = a^{\tau^{-s}} \Gamma(\tau^{-s})^{1-r/s}$ for $a \in \bar{K}$. In order to simplify notation, let us write $\delta_r = \tau^{r-\omega} \delta = \tau^{r-s} \delta$. We compute

$$\phi(\tilde{e}_{r,a})/\tilde{e}_{r,a} = a\tau^{r-s}$$

and

$$\delta_r(\tilde{e}_{r,a})/\tilde{e}_{r,a} = \tau^{r-s}(1 - a\tau^{r-s}).$$

We now further distinguish between $r > s$ and $r = s$. If $r > s$, we set $a = \mu^{-1}$ and obtain

$$\phi(\tilde{e}_{r,\mu})/\tilde{e}_{r,\mu} = \mu^{-1}\tau^{r-s} \quad \text{and} \quad \delta_r(\tilde{e}_{r,\mu})/\tilde{e}_{r,\mu} = \tau^{r-s} + O(\tau^{r-s+1}).$$

Taking into account (5.5) and (5.6), we see that $\tilde{e}_{r,\mu}$ is a resolving irregular part with $\alpha = \mu^{-1}$ and $t = r - s$. If $r = s$, we set $a = (\mu + 1)^{-1}$. The above computations now yield

$$\phi(\tilde{e}_{r,\mu})/\tilde{e}_{r,\mu} = (1 + \mu)^{-1} \quad \text{and} \quad \delta_r(\tilde{e}_{r,\mu})/\tilde{e}_{r,\mu} = \mu/(1 + \mu)$$

so here $\alpha = (1 + \mu)^{-1}$ and $t = 0$ and the claim follows again from (5.5) and (5.6).

If $s > r > 0$, we compute

$$\phi(e_{r,\mu}) = \exp\left(-\frac{s\mu}{r} \tau^{-r} (1 + \tau^s)^{\frac{r}{s}}\right) = e_{r,\mu} \exp(-\mu\tau^{s-r} + \dots)$$

by expanding parts of the exponential as power series. Using further series expansion, we obtain

$$\phi(e_{r,\mu})/e_{r,\mu} = 1 - \mu\tau^{s-r} + \dots \quad \text{and} \quad \delta_r(e_{r,\mu})/e_{r,\mu} = \mu + \dots.$$

Comparing this with (5.5) and (5.6), we see that we can put $\alpha = 1$ and $t = 0$, and the claim follows. \square

Remark 5.5.1. *None of the resolving irregular parts in (5.17) are, in general, canonical.*

5.5.3 Linear q -Difference Systems

Finally, we are interested in the third special case of linear functional systems, those of q -difference systems. We consider a system of the form

$$Y(\tau) - Y(q\tau) = \tau^{-r} A(\tau) Y(q\tau) \quad (5.18)$$

with $r \geq 0$, $q \neq 0$ and q not a root of unity. This corresponds to the system (2.39) with the choices $\phi(\tau) = q\tau$ and $\delta = \text{id} - \phi$ as in (iii) in Chapter 2. The structure of a formal fundamental matrix solution of this system is given e.g. in [31] as:

$$Y(\tau) = H(\tau) \tau^\Lambda q^R \quad (5.19)$$

where $H(\tau) \in \text{GL}(n, \bar{K}[[\tau]])$, $\Lambda \in \mathcal{M}_n(\bar{K})$ and

$$R = \text{diag}\left(\frac{r_1}{2}(t_1^2 - t_1), \dots, \frac{r_n}{2}(t_1^2 - t_1)\right) \quad (5.20)$$

with $s_i \in \mathbb{N}^+$, $r_i \in \mathbb{Q}$ and $t_i = \text{lq}(\tau)/s_i$ for $i = 1, \dots, n$.

Proposition 5.5.3. *Given the system (5.18) with Poincaré-rank $r > 0$ and leading matrix A_0 with eigenvalue μ with $\mu \neq 0$, then*

$$e_{r,\mu} = q^{\frac{r}{2}(\text{lq}(\tau)^2 - \text{lq}(\tau))} \tau^{-\text{lq}\mu} \quad (5.21)$$

is a resolving irregular part of lift $t = r$ and scale $\alpha = 1/\mu$.

Proof We compute

$$\begin{aligned} \phi(e_{r,\mu}) &= q^{\frac{r}{2}((\text{lq}\tau+1)^2 - (\text{lq}\tau+1))} q^{-\text{lq}\mu} \tau^{-\text{lq}\mu} \\ &= q^{\frac{r}{2}(\text{lq}^2\tau - \text{lq}\tau)} q^{r\text{lq}\tau} \mu^{-1} \tau^{-\text{lq}\mu} \\ &= e_{r,\mu} \mu^{-1} \tau^r \end{aligned}$$

hence we obtain

$$\frac{\phi(e_{r,\mu})}{e_{r,\mu}} = \mu^{-1}\tau^r.$$

Consequently,

$$\tau^{r-\omega} \frac{\delta_r(e_{r,\mu})}{e_{r,\mu}} = \tau^r(1 - \mu^{-1}\tau^r) = \tau^r + \dots.$$

Comparing the last two equations with (5.5) and (5.6), we obtain $t = r$ and $\alpha = \mu^{-1}$ which finishes the proof. \square

The presence of the positive integer s_i in the formulae for the t_i in (5.20) is explained by ramifications being introduced during the formal reduction. The operation (R3) then yields a new system with the automorphism $\phi(\tau) = q^{s_i}\tau$. Substituting for this new value in the q -logarithm in (5.21) leads to dividing by s_i in the formula for the t_i in (5.20).

Remark 5.5.2. *The resolving irregular parts as defined in the previous proposition are not canonical ones.*

Chapter 6

Conclusion

In this thesis, we have given a unifying theoretical framework for defining a formal reduction of pseudo-linear systems. This involved adapting various reduction algorithms as well as defining regular, irregular and hyperexponential parts of formal solutions of pseudo-linear systems. We have designed a generic formal reduction algorithm, that can compute a basis of formal solutions for large classes of linear differential, difference and q -difference systems in the neighbourhood of an irregular singular point. We have implemented our algorithm in the Computer Algebra system Maple, as part of the Open Source project ISOLDE [22].

Whereas the generalisation of the Moser-reduction (as well as super-reduction) principle and algorithm, which were the first research objective that we addressed, was fairly straightforward both from a theoretical and implementational point of view, the tasks of defining and computing regular and irregular formal solutions were increasingly challenging. We feel that our theoretical framework is complete, although the terminology necessary for the irregular formal solutions remains somewhat heavy.

Further work concerning the algorithm for computing regular formal solutions in Chapter 4 would mainly focus on improving the computation of the series part of the solutions. Here, our approach effectively truncates the input before it computes the individual monomials, part of the series. It would be desirable to remain with the lazy evaluation approach. However, it is not clear how to achieve this easily, even in the case of scalar n th-order equations. Regarding the formal reduction developed in Chapter 5, we would like to address the fact that the formal reduction method itself could be significantly improved by investigating the efficient computation of ramifications. At present, this seems a difficult problem. One needs to find an efficient way of dealing with the case where the system's Katz-invariant is a rational number. This is a well-known problem in the differential [14] and difference [17] case, and it seems likely that any solution for those systems could then be generalised to pseudo-linear systems. In summary, any satisfactory method would allow for the following

aspects:

- Determine the Katz-invariant $\kappa = p/s$ without having to introduce unnecessary ramifications,
- A splitting of the system *before* introducing the ramification s ,
- To be valid generically, i.e. in the differential, difference and q -difference case.

We believe that, provided a suitable mathematical framework for further exploring the link between the algebraic and differential/difference characteristic polynomial of the system matrix, as commenced in [48], would be a promising line of investigation for the first item above. The results in [64] are formulated for a more general type of systems that we have covered in this thesis: they are established for linear systems of differential-algebraic equations. Nevertheless, they will remain valid for linear differential systems and, together with a solution for the first item, could be generalised to a rational splitting in the pseudo-linear case as required for the second item.

Similarly as in the scalar case, it is clear that the local information that can be computed using our algorithm will be useful for a variety of symbolic tasks for pseudo-linear systems. For example, computing rational and hyperexponential solutions are applications of the formal reduction for computing closed-form solutions. Using our generic approach, we anticipate the design of algorithms for these latter tasks that will work equally well for differential, difference and q -difference systems. This would significantly enhance the way Computer Algebra can handle these systems.

The progress we report in this dissertation has also created new interesting questions. The concept of k -simple systems suggests considering more general types of systems of the form $D\delta Y = N\phi Y$ where we do not necessarily have $\det D \neq 0$. These are systems of pseudo-linear equations that are coupled with purely algebraic conditions, and they have started to be investigated [55, 24]. It seems likely that a formal reduction process for these systems could be devised without major difficulties. On the other hand, efficient implementation using lazy evaluation would not easily be possible merely based on the ISOLDE code base.

The systems considered by us were first-order systems. An interesting and important result would be the generalisation of our methods to higher-order linear functional systems. In particular, a crucial step would be to do this for the results presented in Chapter 3, as the Moser- and super-reduction are at the heart of the formal reduction. This is so far an open problem and its resolution would require an intimate knowledge of both the family of Moser-reduction algorithms and the theory of higher-order linear functional system. Whereas some

work has been done in [8] for linear differential-algebraic equations, it does not seem clear at the moment how to exploit this for advancing Moser-reduction type algorithms. Methods for computing simple formal power series solutions of higher-order systems have been suggested in the difference ([3]) and differential ([26]) case. However a systematic approach based on our ideas would again require the adaptation of the Moser- and super-reduction algorithms to higher order.

Our implementation completely covers the case of differential and difference systems, however the q -difference case remains to be done. Whereas there are no principle difficulties in doing this, we were facing some technical limitations linked to using existing library code of ISOLDE which was not initially designed for an extension to other types of equations.

Designing efficient, symbolic algorithms is one aspect of Computer Algebra. Providing an actual implementation that is reliable, robust and reasonably efficient for typical user input is another one. We have strived to enhance the functionality of the ISOLDE package with an implementation of the methods developed in this dissertation. This has been challenging at times and we had to learn about general software engineering principles that become necessary when contributing to any complex software project.

Chapter 7

List of Terms

Chapter 2

- F, K : fields of characteristic zero, Page 18.
- \bar{K} : algebraic closure of K , Page 18.
- $K(x)$: field of rational functions over K , Page 18.
- $K[[\tau]]$: ring of formal power series with coefficients in K , Page 21.
- $K((\tau))$: field of formal Laurent series with coefficients in K , Page 18.
- $K((\tau^{1/s}))$: field of formal Puiseux series with coefficients in K , Page 18.
- D : differential operator, Page 10.
- Δ : difference operator, Page 10.
- Δ_q : q -difference operator, Page 12.
- $[n]_q$: q -bracket, Page 12.
- ϕ : an automorphism of F , Page 18.
- δ : a pseudo-derivation, Page 18.
- Pseudo-derivation: Definition 2.3.1, Page 18.
- System of pseudo-linear equations: Definition 2.6.1, Page 22.
- γ : field element linking ϕ and δ in case $\phi \neq \text{id}$, Page 18.
- (ϕ, δ) -field: Definition 2.3.2, Page 19.
- Field of constants: Definition 2.3.3, Page 19.

- v : valuation map: Definition 2.5.1, Page 21.
- lc : leading coefficient: Definition 2.5.1, Page 21.
- \mathcal{O} : the valuation ring: Definition 2.5.2, Page 21.
- $\pi: \mathcal{O} \rightarrow K$: evaluation map. Definition 2.5.2, Page 21.
- \mathcal{U} : group of units of \mathcal{O} : Definition 2.5.2, Page 21.
- A, B, \dots : matrices, Page 14.
- $\mathcal{M}_n(F)$: ring of square matrices of dimension n with entries in F , Page 21.
- $GL(n, F)$: invertible matrices of dimension n , Page 21.
- ω : the degree of a pseudo-derivative: Definition 2.5.3, Page 21.
- q : the leading coefficient of $\phi(\tau)$, Page 21.
- c : the leading coefficient of $\delta(\tau)$, Page 21.
- Singular system of pseudo-linear equations: Definition 2.4.1, Page 19.
- r : Poincaré-rank of a system (differential, difference, ...): Definition 2.4.1, Page 19.
- $T_{\delta, \phi}$: Gauge transformation: Definition 2.6.2, Page 22.
- Regular singularity: Definition 2.6.3, Page 23.
- Irregular singularity: Definition 2.6.3, Page 23.

Chapter 3

- $m_{\delta, \phi}$ and $\mu_{\delta, \phi}$: rational numbers associated with a system of pseudo-linear equations, Page 28.
- $\theta(\lambda)$: polynomial associated with a Moser-irreducible system, Page 28.
- Irreducible system: Definition Definition 3.2.1, Page 3.2.1.
- $L(A, \lambda)$: L -matrix of a system, Page 30.
- $m_{\delta, \phi}^k$ and $\mu_{\delta, \phi}^k$: rational numbers associated with a system of pseudo-linear equations, Page 35.
- $\theta_k(\lambda)$: polynomial associated with a super-irreducible system, Page 35.
- k -irreducible, super-irreducible system: Definition 3.4.1, Page 3.4.1.
- $E_{i, j}(\alpha)$: elementary operation matrix, Page 36.

Chapter 4

- System of the first kind: Definition 4.1.1, Page 40.
- $\varphi(\lambda)$: indicial polynomial of a system, Page 41.
- $\sigma_{c,q}$: (c, q) -spectrum, Page 41.
- Good (c, q) -spectrum: Definition 4.1.2, Page 41.
- Regular formal solution: Definition 4.1.4, Page 50.
- e_λ : generalised exponent of a regular formal solution, Page 44.
- Regular part $\text{reg}(y)$ of a formal solution y : Definition 4.1.4, Page 50.
- Simple system: Definition 4.2.1, Page 51.

Chapter 5

- Hyperexponential extension of a (ϕ, δ) -field: Definition 5.2.2, Page 67.
- Irregular part $\text{irr}(y)$ of a formal solution y : Definition 5.2.1, Page 66.
- Hyperexponential part $\text{hyp}(y)$ of a formal solution y : Definition 5.2.3, Page 68.
- Certificate $\text{cert}(h)$ of a hyperexponential element h : Definition 5.2.2, Page 67.
- $\text{term}_{\mu,\alpha,t}$: term transformation, Page 68.
- lift: the lift of a resolving irregular part or term transformation, Page 72.
- scale: the scale of a resolving irregular part or term transformation, Page 72.
- κ : Katz-invariant of a system, Page 70.
- $e_{r,\mu}$: resolving irregular part: Definition 5.2.4, Page 72.

Bibliography

- [1] S.A. Abramov. Rational solutions of linear differential and difference equations with polynomial coefficients. *Compute. Math. Math. Phys.*, 29:1611–1620, 1989.
- [2] S.A. Abramov. Rational solutions of linear difference and q -difference equations with polynomial coefficients. In *Proceedings of ISSAC'95*, pages 303–308. ACM Press, 1995.
- [3] S.A. Abramov. Eg-eliminations. *Journal of Difference Equations and Applications*, 5(4):393–433, 1999.
- [4] S.A. Abramov and M.A. Barkatou. Rational Solutions of First Order Linear Difference Systems. In Oliver Gloor, editor, *Proceedings of ISSAC '98*, pages 124–131, Rostock, Germany, 1998. ACM Press.
- [5] S.A. Abramov, M. Bronstein, and M. Petkovsek. On polynomial solutions of linear differential operator equations. In A.H.M. Levelt, editor, *Proceedings of ISSAC '95*, pages 290–295, Montreal, Canada, 1995. ACM Press.
- [6] Sergei A. Abramov, Peter Paule, and Marko Petkovsek. q -hypergeometric solutions of q -difference equations. *Discrete Math*, 180:3–22, 1998.
- [7] C.R Adams. On the linear ordinary q -difference equation. *Ann. Math.*, 30(2):195–205, 1929.
- [8] M. Barkatou, C. El Bacha, G. Labahn, and E. Pflügel. On simultaneous row and column reduction of higher-order linear differential systems. *Journal of Symbolic Computation*, (19):45–64, 2011.
- [9] M. A. Barkatou, G. Broughton, and E. Pflügel. A monomial-by-monomial method for computing regular solutions of systems of pseudo-linear equations. *Mathematics in Computer Science*, 4(2-3):267–288, 2010.
- [10] M.A. Barkatou. Rational Newton Algorithm for Computing Formal Solutions of Linear Differential Equations. In *Proceedings of ISSAC '88*, pages 183–195, Rome, Italy, 1988. ACM Press.
- [11] M.A. Barkatou. On the reduction of linear systems of difference equations. In *Proceedings of ISSAC '89*, Portland, USA, 1989. ACM Press.
- [12] M.A. Barkatou. Charaterization of regular singular linear systems of difference equations. *Numerical Algorithms*, 1:139–154, 1991.

- [13] M.A. Barkatou. An algorithm for computing a companion block diagonal form for a system of linear differential equations. *Journal of App. Alg. in Eng. Comm. and Comp.*, 4, 1993.
- [14] M.A. Barkatou. An algorithm to compute the exponential part of a formal fundamental matrix solution of a linear differential system. *Journal of App. Alg. in Eng. Comm. and Comp.*, 8(1):1–23, 1997.
- [15] M.A. Barkatou. On Rational Solutions of Systems of Linear Differential Equations. *Journal of Symbolic Computation*, 28:547–567, 1999.
- [16] M.A. Barkatou. Factoring systems of linear functional equations using eigenrings. In *COMPUTER ALGEBRA 2006, Latest Advances in Symbolic Algorithms. Proceedings of the Waterloo Workshop. Ontario, Canada, 2006*.
- [17] M.A. Barkatou and G. Chen. Computing the exponential part of a formal fundamental matrix solution of a linear difference system. *Journal of Difference Equations and Applications*, 5:117–142, 1999.
- [18] M.A. Barkatou and G. Chen. Some formal invariants of linear difference systems and their computations. *Journal fur die reine und angewandte Mathematik*, 553:1–23, 2001.
- [19] M.A. Barkatou and E. Pfügel. A rational algorithm to compute a super-irreducible form of a linear differential system. RR, LMC-IMAG, 1997.
- [20] M.A. Barkatou and E. Pfügel. On the equivalence problem of linear differential systems and its application for factoring completely reducible systems. In *Proceedings of ISSAC '98*, pages 268–275, Rostock, Germany, 1998. ACM Press.
- [21] M.A. Barkatou and E. Pfügel. An algorithm computing the regular formal solutions of a system of linear differential equations. *Journal of Symbolic Computation*, 28:569–588, 1999.
- [22] M.A. Barkatou and E. Pfügel. The ISOLDE package. A SourceForge Open Source project, <http://isolde.sourceforge.net>, 2006.
- [23] M.A. Barkatou and E. Pfügel. Computing super-irreducible forms of systems of linear differential equations via Moser-reduction: A new approach. In *Proceedings of ISSAC '07*, pages 1–8, Waterloo, Canada, 2007. ACM Press.
- [24] Moulay A. Barkatou and Carole El Bacha. On k-simple forms of first-order linear differential systems and their computation. *Journal of Symbolic Computation*, 54(0):36 – 58, 2013.
- [25] Moulay A. Barkatou, Gary Broughton, and Eckhard Pfügel. Regular systems of linear functional equations and applications. In *ISSAC '08: Proceedings of the twenty-first international symposium on Symbolic and algebraic computation*, pages 15–22, New York, NY, USA, 2008. ACM.
- [26] Moulay A. Barkatou, Thomas Cluzeau, and Carole El Bacha. Simple forms of higher-order linear differential systems and their applications in computing regular solutions. *J. Symb. Comput.*, 46:633–658, June 2011.

- [27] Moulay A. Barkatou and Eckhard Pflügel. On the Moser- and super-reduction algorithms of systems of linear differential equations and their complexity. *J. Symb. Comput.*, 44(8):1017–1036, 2009.
- [28] G. Birkhoff. Formal theory of irregular difference equations. *Acta Mathematica*, 54:205–246, 1930.
- [29] G.D. Birkhoff. Singular points of ordinary differential equations. *Trans. Amer. Math. Soc.*, 10:436–470, 1909.
- [30] G.D. Birkhoff. The generalized Riemann problem. *Proc. Am. Acad.*, 49:521–568, 1913.
- [31] G.D. Birkhoff and P.E. Guenther. Note on a canonical form for the linear q -difference system. *Proc. Nat. Acad. Sci.*, 27(4):218–222, 1941.
- [32] M. Bronstein. Linear ordinary differential equations: breaking through the order 2 barrier. In *Proceedings of ISSAC '92*, pages 42–48, Berkeley, USA, 1992. ACM Press.
- [33] M. Bronstein. On solutions of linear ordinary differential equations in their coefficient field. *Journal of Symbolic Computation*, 13:413–439, 1992.
- [34] M. Bronstein. On solutions of linear ordinary difference equations in their coefficient field. *Journal of symbolic computation*, 29(6):841–877, 2002.
- [35] M Bronstein and M. Petkošek. On ore rings, linear operators and factorisation. *Programmirovanie*, 1994.
- [36] M. Bronstein and M. Petkošek. An Introduction to Pseudo-Linear Algebra. *Theoretical Computer Science*, 157(1):3–33, 1996.
- [37] M Bronstein and M. Petkošek. An introduction to pseudo-linear algebra. *Theoretical computer science*, pages 3–33, 1996.
- [38] R.D. Carmichael. The general theory of linear q -difference equations. *American Journal of Mathematics*, 34(2):147–168, 1912.
- [39] Guoting Chen and Abdelmajid Fahim. Formal reduction of linear difference systems. *Pacific J. Math.*, 182:37–54, 1998.
- [40] K.E. Chu. The solution of the matrix equations $AXB - CXD = E$ and $(YA - DZ, YC - BZ) = (E, F)$. *Linear Algebra Appl.*, 93:93–105, 1987.
- [41] R. C. Churchill and J. J. Kovacic. Cyclic vectors. In Newark Rutgers University, editor, *Differential Algebra and Related Topics, Proceedings of the International Workshop*. River Edge, NJ, World Scientific Publishing Co., November 2–3 2002.
- [42] Thomas Cluzeau. Factorization of differential systems in characteristic p . In *Proceedings of ISSAC '03*, pages 58–65, New York, NY, USA, 2003. ACM.
- [43] Thomas Cluzeau and Mark van Hoeij. Computing hypergeometric solutions of linear recurrence equations. *Appl. Algebra Eng. Commun. Comput.*, pages 83–115, 2006.
- [44] P. Deligne. Equations différentielles à points singuliers réguliers. *Lecture Notes in Mathematics*, 163, 1970.

- [45] V. Dietrich. Zur Reduktion von linearen Differentialgleichungssystemen. *Math. Ann.*, 237:79–95, 1978.
- [46] W.A. Harris. Linear systems of difference equations. *Contributions to Differential Equations*, 1:489–518, 1963.
- [47] P. Hartman. *Ordinary Differential Equations*. John Wiley & Sons, Inc., 1964.
- [48] A. Hilali. On the algebraic and differential Newton–Puiseux polygons. *Journal of Symbolic Computation*, 4(3):335–349, December 1987.
- [49] A. Hilali. Solutions formelles de systèmes différentiels linéaires au voisinage d’un point singulier. *Thèse d’Etat, Université de Grenoble*, June 1987.
- [50] A. Hilali and A. Wazner. Formes super-irréductibles des systèmes différentiels linéaires. *Numer. Math.*, 50:429–449, 1987.
- [51] P Horn. *Faktorisierung in Schief-Polynomringen*. PhD thesis, University of Kassel, Kassel, Germany, 2009.
- [52] N. Jacobson. Pseudo-linear transformations. *Annals of Mathematics*, 33(2):484–507, 1937.
- [53] Victor Kac and Pokman Cheung. *Quantum Calculus*. Springer, 2002.
- [54] George Labahn and Ziming Li. Hyperexponential solutions of finite-rank ideals in orthogonal ore rings. In *Proceedings ISSAC*, pages 213–220, 2004.
- [55] E. Pflügel M.A. Barkatou, C.Bacha. Simultaneously row- and column-reduced higher-order linear differential systems. Submitted to ISSAC 2010.
- [56] B. Malgrange. Sur la réduction formelle des équations différentielles à singularités irrégulières. Manuscript, Université Joseph Fourier, 1979.
- [57] J Martinet and J.P. Ramis. Classification analytique des quations différentielles non linaires rsonnantes du premier ordre. *Annales scientifiques de l’ecole normale superieure*, 4(4):571–621, 1983.
- [58] J. Moser. The order of a singularity in Fuchs’ theory. *Math. Z.*, pages 379–398, 1960.
- [59] O. Ore. Formale Theorie der linearen Differentialgleichungen (Zweiter Teil). *J. für die Reine u. angewandte. Mathematik*, 168:233–252, 1932.
- [60] E. Pflügel. An algorithm for computing exponential solutions of first order linear differential systems. In *Proceedings ISSAC ’97*, pages 164–171, Maui, Hawaii, 1997. ACM Press.
- [61] E. Pflügel. On the Latest Version of DESIR. *Theoretical Computer Science*, 187:81–86, 1997.
- [62] E. Pflügel. *Résolution symbolique des systèmes différentiels linéaires*. PhD thesis, LMC-IMAG, 1998.

- [63] E. Pflügel. Effective formal reduction of linear differential systems. *Applicable Algebra in Engineering, Communication and Computation*, 10(2):153–187, 2000.
- [64] E. Pflügel. A rational decomposition-lemma for systems of linear differential-algebraic equations. In *ISSAC '08: Proceedings of the twenty-first international symposium on Symbolic and algebraic computation*, pages 231–238, New York, NY, USA, 2008. ACM.
- [65] J. Sauloy. Systemes aux q -différences singuliers réguliers: classification, matric de connexion et monodromy. *Annales De L'Institut Fourier*, 50(4):1021–1071, 2000.
- [66] E. Tournier. *Solutions formelles d'équations différentielles*. Thèse d'Etat, Université de Grenoble, 1987.
- [67] A. Tovbis. Normal forms of "near similarity" transformations and linear matrix equations. *Linear Algebra and its Applications*, (317):13–40, 2000.
- [68] W.J. Trjitzinsky. Analytic theory of linear q -difference equations. *ACTA Mathematica*, 61(1), 1933.
- [69] H.L. Turritin. Convergent solutions of ordinary linear homogeneous differential equations in the neighborhood of an irregular singular point. *Acta Math.*, 93:27–66, 1955.
- [70] H.L. Turritin. The formal theory of systems of irregular homogeneous linear difference equations. *Bolitin de la Soc. Mexicana*, 5:255–264, 1960.
- [71] M. van Hoeij. Factorization of differential operators with rational function coefficients. *Journal of Symbolic Computation*, 24:537–561, 1997.
- [72] M. van Hoeij. Formal solutions and factorization of differential operators with power series coefficients. *Journal of Symbolic Computation*, 24:1–30, 1997.
- [73] M. van Hoeij and J.-A. Weil. An algorithm for computing invariants of linear differential Galois groups. *Pure Appl. Algebra*, 117,118:353–379, 1997.
- [74] W. Wasow. *Asymptotic Expansions for Ordinary Differential Equations*. Robert E. Krieger Publishing, 1967.
- [75] B. Zürcher. Rationale Normalformen von pseudo-linearen Abbildungen. Master's thesis, ETH Zürich, 1994.