GEOMETRIC SYMBOLIC-NUMERIC METHODS FOR DIFFERENTIAL AND ALGEBRAIC SYSTEMS

(Spine title: New Progress in Numerical Jet Geometry)

by

Wenyuan Wu

Graduate Program
in
Department of Applied Mathematics

Submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Faculty of Graduate Studies
The University of Western Ontario
London, Ontario
July, 2007

THE UNIVERSITY OF WESTERN ONTARIO
FACULTY OF GRADUATE STUDIES

CERTIFICATE OF EXAMINATION

Chief Advisor

Co-Supervisor

Examining Board

The thesis by
Wenyuan Wu

entitled
GEOMETRIC SYMBOLIC-NUMERIC METHODS FOR DIFFERENTIAL AND ALGEBRAIC SYSTEMS

(Spine title: New Progress in Numerical Jet Geometry)

is accepted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Date

Chairman of Examining Board

ABSTRACT

General (e.g. under and over-determined) systems of polynomially nonlinear partial differential equations (PDE) arise frequently in diverse applications. Analyzing such systems requires differentiations and eliminations to detect and include all their integrability conditions. Existing symbolic differential elimination algorithms for this purpose can be prohibitively expensive and only apply to exact systems of PDE and do not stably apply to the approximate systems occurring in applications.

The main contributions of this Thesis are to provide the first practical and stable methods to address the above problems for approximate PDE; and to establish mathematical foundations for this area. These contributions build on a proposal by Reid et al, to extend tools from Numerical Algebraic Geometry to general systems of PDE in the framework of the geometric theory of PDE (Jet Geometry).

Differentiating systems yields PDE systems that are always linear in their highest derivatives. Two methods are given to exploit this linearity. One is a hybrid method that applies to exact input systems. The other applies to approximate systems. For a class of PDE appearing in applications, we give an efficient method that only requires differentiations with respect to one independent variable.

As in Numerical Algebraic Geometry, in Numerical Jet Geometry, the components of PDE are geometrically represented by certain approximate witness points, cut out by intersection of random linear spaces with the components. Such witness points can be efficiently and stably computed by numerical homotopy continuation methods.

Keywords: Jet Geometry; Involution; Formal Integrability; Cartan Kuranishi Algorithm; Numerical Algebraic Geometry; Homotopy Continuation; Approximate Triangular Decomposition; Polynomial Matrix; Riquier Bases.

Co-Authorship Statement

Chapter 2 – 6 of this thesis consist of the following papers:

- **Chapter 2** Greg Reid, Jan Verschelde, Allan Wittkopf and Wenyuan Wu: *Symbolic-Numeric Completion of Differential Systems by Homotopy Continuation.* Proceedings of ISSAC'05, pages 269-276, ACM 2005.

- **Chapter 3** Wenyuan Wu and Greg Reid: *Application of Numerical Algebraic Geometry and Numerical Linear Algebra to PDE.* Proceedings of ISSAC'06, pages 345-353, ACM 2006.

- **Chapter 4** Wenyuan Wu and Greg Reid: *Symbolic-numeric Computation of Implicit Riquier Bases for PDE.* Proceedings of ISSAC'07, pages 377-385, ACM 2007.

- **Chapter 5** Marc Moreno Maza, Greg Reid, Robin Scott and Wenyuan Wu: *On Approximate Triangular Decompositions in Dimension Zero.* Journal of Symbolic Computation, Vol 42(7), page 693-716, 2007.

- **Chapter 6** Wenyuan Wu: *Computing the Rank and Null-space of Polynomial Matrices*, preprint.

As the leading author and organizer of the published papers in Chapter 3 and 4, I developed the mathematical theory, did the implementations and experiments and wrote the major part of the papers.

For the published paper in Chapter 2, I assembled the contributions of other authors, designed and implemented the main algorithm and wrote the central sections.

I developed and wrote the technical part of the paper given in Chapter 5, including the statistical tools for stability analysis of our methods. Also I did all the implementations and experiments.

To my lovely wife Chunyan Ou,

for her love and sacrifice.

To my mother Yulian Wang, and my father Junzhi Wu,

for their encouragement and support.

ACKNOWLEDGEMENTS

First of all, I wish to express my deepest gratitude to my Ph.D. thesis supervisor, Dr. Greg Reid. During the past four years, I got lots of invaluable advice, kind guidance and selfless help from him. He introduced me into the world of the formal theory of partial differential equations. Motivated by his insight about the relation between the formal theory of PDE and Numerical Algebraic Geometry, a completely new area, so-called Numerical Jet Geometry is initiated. My work about the approximate polynomial algebra also follows his idea. I also thank him for giving me a lot of support and help for my research, teaching and also generously improving the living quality of my family during my Ph.D. study.

Secondly, I should appreciate my lovely wife Chunyan Ou for her unselfish support. She has sacrificed her youth, her job, her interests and her energy to my four years Ph.D. study in Canada.

Moreover, I would like to thank Drs. Robert M. Corless, Dr. Marc Moreno Maza and Dr. Eric Schost in ORCCA for their kind guidance for my course projects in their courses. From them, I learned the right attitude and way to do research, which will benefit me throughout my career.

The collaboration with Drs. Jan Verschelde and Allan Wittkopf on the HybridRif project gave a good start to my study.

I acknowledge IMA for the prestigious NSF supported visit to the Program "Applications of Algebraic Geometry" in Minneapolis.

I thank the China Scholar Council for awarding me the Chinese Government Award for Outstanding Students and acknowledge Lihong Zhi for support for a visit to the MMRC at the Chinese Academy of Sciences in Beijing.

I thank Ms. Pat Malone, Audrey Kager and Karen Foullong in our Applied Mathematics department for their conscientious work and valuable help.

There are lots of friends in our department, and even though we are from different backgrounds, we help each other like brothers and sisters. I would like to thank all of you: Fei Xu, Hui Ding, officemate Ramin Nowbakht Ghalati, Shudan Liu, Guangzhi Zhao and roommate Jichao Zhao.

Contents

Certificate of Examination ii

Abstract iii

Co-Authorship iv

Dedication vi

Acknowledgments vii

1 Introduction **1**

1.1 Background and Motivation 2

1.2 Formal Theory of PDE 3

1.2.1 Jet Bundles 5

1.2.2 Differential Functions and Differential Equations 6

1.2.3 Formal Power Series Solutions 7

1.2.4 Involution 8

1.2.5 Spencer Cohomology 10

1.2.6 Cartan Kuranishi Completion 13

1.2.7 Other General Methods for PDE 14

1.3 Numerical Algebraic Geometry 17

1.3.1 Homotopy Continuation Methods 17

1.3.2 Polytope Structure 20
1.3.3 Positive Dimensional Systems 27
1.4 Organization and Comments 31
1.4.1 Comments on [39] (Chapter 2) 32
1.4.2 Comments on [62] (Chapter 3) 32
1.4.3 Comments on [63] (Chapter 4) 33
1.4.4 Comments on [30] (Chapter 5) 33
1.4.5 Comments on [66] (Chapter 6) 34

2 Symbolic-Numeric Completion of Differential Systems by Homotopy Continuation 40

2.1 Introduction 40
2.2 Symbolic Differential Elimination 41
2.3 Numerical Algebraic Geometry 43
2.4 Symbolic-Numeric Completion Algorithm 44
2.4.1 Using Witness Sets 44
2.4.2 Specification of rifsimp 45
2.4.3 The main algorithm 45
2.4.4 Termination Conditions 46
2.5 Optimizations 47
2.6 Examples 48
2.6.1 Illustrative Example 48
2.6.2 System for Discrete Symmetries 50
2.6.3 Random first order ODE 53
2.7 Discussion 55

3 Application of Numerical Algebraic Geometry and Numerical Linear Algebra to PDE 60

3.1 Introduction . . . 60

3.2 PDE in Jet Space . . . 62

3.2.1 Jet Space and Jet variety of a PDE . . . 62

3.2.2 Prolongation and Projection . . . 62

3.2.3 Formally Integrable and Involutive Systems . . . 63

3.2.4 Cartan-Kuranishi Completion . . . 64

3.3 Polynomial Matrix . . . 64

3.3.1 Rank of Polynomial Matrix . . . 65

3.3.2 Computing the Null-space . . . 66

3.4 Numerical Completion Methods . . . 67

3.4.1 Using Witness Points . . . 68

3.4.2 Numerical Differential Elimination . . . 68

3.5 Simple Examples . . . 69

3.6 Physical Example . . . 70

3.7 Random PDE Examples . . . 71

3.8 Experiments with Approximate Ideal Membership Testing . . . 72

3.9 Discussion . . . 75

4 Symbolic-numeric Computation of Implicit Riquier Bases for PDE 81

4.1 Introduction . . . 81

4.2 Zero Set of PDE . . . 83

4.3 Rankings of Derivatives . . . 84

4.4 Signature Matrix of t-Dominated Systems using Rankings . . . 85
4.5 Generalizing Pryce's Prolongation Method to PDE . . . 88
4.6 The Formal Riquier Existence Theorem . . . 90
4.6.1 Implicit Riquier Existence Theorem . . . 92
4.7 Approximating Points on Zero Sets of PDE . . . 95
4.8 Examples . . . 96
4.9 Discussion . . . 99

5 On Approximate Triangular Decompositions in Dimension Zero 104
5.1 Introduction . . . 104
5.2 Triangular decompositions . . . 105
5.3 Approximate Equiprojectable Decomposition in Dimension Zero . . . 110
5.4 Stability Analysis . . . 116
5.5 An illustrative example . . . 122
5.6 Experimental Results . . . 123
5.6.1 Normal distribution test . . . 126
5.6.2 Exact triangular decomposition . . . 126
5.6.3 Approximate triangular sets . . . 127
5.7 Discussion . . . 127

6 Computing the Rank and Null-space of Polynomial Matrices 135
6.1 Introduction . . . 135
6.2 The Rank of a Matrix . . . 136
6.3 Rank of Approximate Polynomial Matrices . . . 138

6.4 Null-space and Syzygy Module 141
6.5 Generalized Sylvester Method 143
6.5.1 Sylvester Matrices and the Algorithm 143
6.5.2 Algorithmic Analysis 148
6.6 Applications 148
6.6.1 Approximate GCD of two multivariate polynomials 148
6.6.2 Projection of the Variety of Quasi-linear Polynomial Systems . 149
6.7 Discussion 151

7 Conclusion and Future Work **156**
7.1 Conclusion and Main Results 156
7.2 Future Research Directions 158

VITA **163**

Chapter 1

Introduction

The mathematical theory of differential equations has developed together with the sciences where the equations originate and where the results find applications. Diverse scientific fields often give rise to initial and boundary value problems for the same differential equations. Applications in modeling and geometry have led to the need to consider general approximate systems of PDE. Any method for such general systems must determine the obstructions to their solvability in the form of integrability conditions by differentiating (prolonging) the systems. Although vital, existing prolongation methods are prohibitively expensive and numerically unstable.

A central task of this Thesis is to provide numerically stable and practical methods to study polynomially nonlinear partial differential equations which are unnecessary to be square systems (such systems are call *general* PDE systems in this Thesis). On the one hand geometric approaches giving an intrinsic and stable way to view the systems are difficult to implement on computers. On the other hand algorithms are closer to algebra, but sometimes we lose geometric insight after a sequence of algebraic operations. To keep geometric information in the algorithms is the main philosophy of the Thesis. Another task is to develop efficient approaches, since realistic problems are often large scale and to find all integrability conditions using full prolongation (differentiation with respect to all independent variables) leads to impractically huge systems.

The main tools to address these problems in the Thesis are the geometric techniques of the formal theory of PDE (which are coordinate independent) and numerical algebraic geometry, which works directly on geometric objects and is based on homotopy continuation techniques.

In this chapter we will give a brief introduction to geometric techniques and the concept of *involution* and show that it is an important and useful concept for PDE. In the central part of this Thesis – the author's series of publications [39, 62, 63] – completion to involution is our main goal. In these papers, we mainly focus on

polynomially nonlinear PDE and completion of such systems to equivalent involutive forms for which power series solutions may be constructed order by order. So in Jet Space, we can consider such differential systems as polynomial systems. This is the key observation enabling the introduction of algebraic geometry methods and computer algebra tools into the study of differential equations.

One of the most important problems in algebraic geometry is to solve polynomial systems. Recently a new area called "Numerical Algebraic Geometry", developed by Sommese, Wampler, and Verschelde [51, 47, 48, 49, 50, 52], provides numerical methods to compute approximation of all isolated (complex) roots of such systems. For positive dimensional systems, they reduce such systems to zero dimensional ones by slicing with random linear equations. In particular the positive dimensional components are represented by using witness points (the solutions of the reduced zero dimensional systems) together with those linear equations. Necessarily, in this chapter we will discuss some important tools and theoretical results in this new area which are crucial for this Thesis.

1.1 Background and Motivation

In this section we discuss the need to study general systems of the type considered in this Thesis.

After the development of calculus, initially only scalar linear PDE of order 1 or 2 were studied. Motivated by applications, existence and uniqueness results for solutions of a wide class of nonlinear determined systems (i.e. # equations = # unknowns) were given as in Cauchy-Kovalevskaya Theory in the 1800's. Symmetry and equivalence applications in the classical work of Lie and Cartan led to the consideration of over-determined systems of PDE (see Olver's book [33] for a historical discussion). Over the last few decades applications in control theory [35] have led to the consideration of under-determined PDE.

Furthermore, in recent decades it was discovered that many applications lead to general systems of differential and algebraic equations. Indeed the applications are so wide-spread and the systems describing them are so complicated that general computer modeling environments have been implemented (e.g. Dynaflex [9]) for automatically producing the systems.

We now discuss existing approaches for general systems of PDE.

Ritt (1950) and Kolchin (1973) started a new field, called *Differential Algebra*, which provides fundamental algebraic theory for general PDE. Riquier (1910) initiated the analytic study and Cartan (1904) introduced geometrical methods for the general systems of PDE expressed as *exterior differential systems*. All these theories constitute an area called *the formal theory of* PDE.

However these different approaches share a common feature: for a system of PDE, to apply the above approaches, we need to differentiate the system with respect to its independent variables to cover all the system's integrability conditions. In a

subsequent elimination steps such integrability conditions need to be simplified to determine if they are satisfied identically, or need to be appended as genuinely new conditions. This process is called *differential elimination*. Unfortunately, the number of new equations after differentiation can grow rapidly, so a computer implementable efficient differential elimination method is vital.

Much progress has been made in exact differential elimination methods, theory and algorithms for nonlinear systems of PDE. For example see Boulier et al. [3], Chen and Gao [5], Hubert [17], Mansfield [28], Seiler [45], Reid, Rust et al. [41, 44, 60, 61], Wu [65]. Such methods enable the identification of all the hidden integrability conditions (or equivalently constraints) for a system of PDE and the automatic statement of an existence and uniqueness theorem for its solutions. They give a geometrical view of its solution space [41, 45] and enable the determination of its symmetry properties. They can ease the difficulty of numerical solution of Differential Algebraic Equations [55] and enable the computation of initial data and associated formal power series solutions in the neighborhood of a point. Algorithmic membership tests (specifically in the radical of a differential ideal) can be given [3, 17].

Despite the considerable progresses above, there are two significant obstacles to such differential elimination techniques. Their inherent complexity puts many problems in applications out of reach. Moreover, the exact methods don't apply stably to systems with approximate coefficients.

This Thesis develops new stable and efficient symbolic-numeric techniques for general systems of PDE in the framework of geometric theory which will be introduced in the next section.

1.2 Formal Theory of PDE

In keeping with our emphasis on general systems, consider systems of PDE with independent variables $(x_1, ..., x_n)$ and dependent variables $(u^1, ..., u^m)$. Let $x \in X$ and $(x, u) \in \mathcal{E}$, where X and $\mathcal{E}$ are manifolds with $\dim X = n$ and $\dim \mathcal{E} = m + n$. Here X is the space of independent variables and $\mathcal{E}$ is the space of independent and dependent variables.

Let $\boldsymbol{\pi} : \mathcal{E} \to X$ be a surjective submersion. See [6] for introductory material on differential geometry. Let $\mathbb{F}$ be a field (usually $\mathbb{C}$ or $\mathbb{R}$).

Definition 1.2.1. *We say that $\mathcal{E}$ is a fibred manifold over X with projection $\boldsymbol{\pi}$, if for any point of $\mathcal{E}$ there exists a coordinate neighborhood $\mathcal{U}$ of this point in $\mathcal{E}$, a local chart $\{\mathcal{U}, \Phi\}$ of $\mathcal{E}$ and a local chart $\{U, \phi\}$ of X, with $U = \boldsymbol{\pi}(\mathcal{U})$, such that the diagram:*

$$\begin{array}{ccc} \mathcal{U} & \xrightarrow{\Phi} & \mathbb{F}^n \times \mathbb{F}^m \\ {\scriptstyle \boldsymbol{\pi}}\downarrow & & \downarrow{\scriptstyle proj} \\ U & \xrightarrow{\phi} & \mathbb{F}^n \end{array} \tag{1.2.1}$$

commutes. Here proj is the natural projection onto the first n-coordinates.

We denote a point of $\mathcal{E}$ by its local coordinates (x, u), where $x = (x_1, ..., x_n), u = (u^1, ..., u^m)$ and denote the projected point in X by (x). The coordinate transformation of $\mathcal{E}$ on $\mathcal{U}_\alpha \cap \mathcal{U}_\beta$ and the transformation of X on $U_\alpha \cap U_\beta$ have the following forms respectively:

$$u^k_\beta = \psi^k_{\alpha\beta}(x_\alpha, u_\alpha) \tag{1.2.2}$$

$$x^i_\beta = \varphi^i_{\alpha\beta}(x_\alpha) \tag{1.2.3}$$

When $\mathcal{E}$ and X are differentiable manifolds and $\boldsymbol{\pi}$ is a differentiable map, we call $\mathcal{E}$ a *differentiable* fibred manifold. In the sequel we assume that all the manifolds and maps are differentiable and specifically that X is a differentiable, connected, paracompact manifold. The most important feature of paracompact Hausdorff spaces is that they admit partitions of unity which enables us to introduce integral on smooth manifolds.

Definition 1.2.2. *A local section of $\mathcal{E}$ over an open set $U \subset X$ is a map $f : U \to \mathcal{E}$, such that for any $x \in U$,$\boldsymbol{\pi} \circ f(x) = x$ (that is $\boldsymbol{\pi} \circ f = id_U$). We call U the domain of f, denoted by* $\mathrm{dom} f$. *In particular, if* $\mathrm{dom} f = X$, *then f is called a global section of $\mathcal{E}$ over X.*

Definition 1.2.3. *For any $x \in X$, $\mathcal{E}_x := \boldsymbol{\pi}^{-1}(x)$ is a closed sub-manifold of $\mathcal{E}$ called the fiber over x.*

Now let Y be another manifold with $\dim Y = m$.

Definition 1.2.4. *A fibred manifold $\mathcal{E}$ over X with $\boldsymbol{\pi} : \mathcal{E} \to X$ is called a bundle over X with fiber Y, if for any open covering $\{U_\alpha\}$ of X, there exist homeomorphisms $\Phi_\alpha : \boldsymbol{\pi}^{-1}(U_\alpha) \to U_\alpha \times Y$, such that the following diagram is commutative:*

$$\begin{array}{ccc} \boldsymbol{\pi}^{-1}(U_\alpha) & \xrightarrow{\Phi_\alpha} & U_\alpha \times Y \\ {\scriptstyle \boldsymbol{\pi}}\big\downarrow & \swarrow_{proj} & \\ U_\alpha & & \end{array} \tag{1.2.4}$$

It is clear that $\mathcal{E} = \bigcup_\alpha \boldsymbol{\pi}^{-1}(U_\alpha)$ and that locally it is homeomorphic to $X \times Y$. If $\mathcal{E} = X \times Y$, it is a bundle over X and it is called a *trivial bundle.*

Remark 1.2.1. *Note that a fibred manifold is not necessary to be a bundle. Let X be a segment $\{(x, 0, 0) \in \mathbb{R}^3 : x \in [-1, 1]\}$ (1 dimensional manifold with boundary) and $\mathcal{E}$ be a sphere $x^2 + y^2 + z^2 = 1$. And the projection $\boldsymbol{\pi} : \mathcal{E} \to X$ sends (x, y, z) to x. When $x \neq 1, -1$, the fiber Y is a circle with dimension 1. But when $x = 1$, $\boldsymbol{\pi}^{-1}(x) = (1, 0, 0)$, which is not homeomorphic to a circle.*

1.2.1 Jet Bundles

Let f and g be two sections of a fibred manifold $\mathcal{E}$ with $\boldsymbol{\pi} : \mathcal{E} \to X$ and let x be a point in $\mathrm{dom}f \cap \mathrm{dom}g$.

We define a multi-index α as a n-tuple $(\alpha_1, \alpha_2, ..., \alpha_n)$ with $\alpha_i \in \mathbb{N}$. The order of the multi-index α, denoted $|\alpha|$ is given by the sum of the α_i.

Definition 1.2.5. *For any integer $q \geq 0$, we say the sections f and g on X are q-equivalent at x if $f(x) = g(x)$ and $\partial_\alpha f(x) = \partial_\alpha g(x)$ for any $1 \leq |\alpha| \leq q$. The equivalence class of f is called the q-jet of f at x and is denoted by $j_q(f)_x$. We define the set $J_q(\mathcal{E})_x$ to be the set of all the q-jets at x of the sections of $\mathcal{E}$ and define $J_q(\mathcal{E}) := \bigcup_{x\in X} J_q(\mathcal{E})_x$.*

The set $J_q(\mathcal{E})$ can be considered as a fibred manifold both over X and $\mathcal{E}$, which is called the bundle of q-jets over $\mathcal{E}$.

It is well known that the total number of derivatives of order q with n independent variables x_i and m dependent variables u^j is $m\binom{n+q-1}{q}$ and $\dim J_q(\mathcal{E}) = n + m\binom{n+q}{q}$.

We can introduce jet variables with order q which have one-to-one correspondence with the derivatives of the dependent variables of order q. The set of all jet variables is defined to be $\Omega = \{u^j_\alpha \ : \ \alpha = (\alpha_1, \cdots, \alpha_n) \in \mathbb{N}^n, j = 1, \cdots, m\}$ where each member of Ω corresponds to a partial derivative by:

$$u^j_\alpha \leftrightarrow (\partial_{x_n})^{\alpha_n} \cdots (\partial_{x_1})^{\alpha_1} u^j(x_1, \cdots, x_n) \ . \tag{1.2.5}$$

For convenience, we use $\underset{q}{u}$ to denote all the q-th order jet variables.

EXAMPLE 1.2.1. *Let $n = 2$ and $m = 1$. Label the independent variables x and y and the dependent variable u. Then the first order jet bundle, $J_1(\mathcal{E})$, has coordinates (x, y, u, u_x, u_y) and $J_2(\mathcal{E})$ has coordinates $(x, y, u, u_x, u_y, u_{xx}, u_{xy}, u_{yy})$ and so on. Here $\underset{0}{u} = u$, $\underset{1}{u} = (u_x, u_y)$, $\underset{2}{u} = (u_{xx}, u_{xy}, u_{yy})$ etc.*

Proposition 1.2.1. *Let $r, s \in \mathbb{N}$ and $s > 1$. If we consider $J_{r+s}(\mathcal{E})$, $J_r(J_s(\mathcal{E}))$ and $J_{r+1}(J_{s-1}(\mathcal{E}))$ as fibred sub-manifolds of $J_1^{r+s}(\mathcal{E}) := \underbrace{J_1(J_1(\cdots J_1(\mathcal{E})))}_{r+s \text{ times}}$ over X in a natural way, we have the relation:*

$$J_{r+s}(\mathcal{E}) = J_r(J_s(\mathcal{E})) \cap J_{r+1}(J_{s-1}(\mathcal{E})). \tag{1.2.6}$$

See Chapter 1 of reference [34] for the proof of this proposition. The basic idea is very simple. Suppose there is only dependent variable. If we use a vector $(a_1, a_2, ..., a_{r+s})$ to represent a variable in $J_1^{r+s}(\mathcal{E})$, where $a_i \in \{1, ...n\}$. And a jet variable in $J_q(\mathcal{E})$ corresponds to $(a_1, a_2, ..., a_q)$ with $a_1 \leq \cdots \leq a_q$ (because of symmetry of partial derivatives). So a variable in $J_r(J_s(\mathcal{E}))$ corresponds to a vector $(a_1, a_2, ..., a_s, a_{s+1}, ..., a_{s+r})$ with $a_1 \leq \cdots \leq a_s$ and $a_{s+1} \leq \cdots \leq a_{s+r}$. If this variable

is also in $J_{r+1}(J_{s-1}(\mathcal{E}))$ then it must satisfy $a_1 \leq \cdots \leq a_{s-1}$ and $a_s \leq \cdots \leq a_{s+r}$ which implies $a_1 \leq \cdots \leq a_{s+r}$. So it is a jet variable in $J_{r+s}(\mathcal{E})$. Conversely we can split one ascending sequence into two.

We denote by $\boldsymbol{\pi}_q^{q+r} : J_{q+r}(\mathcal{E}) \to J_q(\mathcal{E})$ the map sending a $(q+r)$-jet $j_{q+r}(f)$ to the q-jet $j_q(f)$ and we identify $J_0(\mathcal{E})$ with $\mathcal{E}$. For notational brevity we will often omit $\mathcal{E}$ from $J_q(\mathcal{E})$, writing J_q.

1.2.2 Differential Functions and Differential Equations

Locally $\mathcal{E}$ is homeomorphic to $\mathbb{F}^n \times \mathbb{F}^m$, so $J_q = J_q(\mathbb{F}^n, \mathbb{F}^m)$. We say a *differential function* on J_q is an analytic, $\mathbb{F}$-valued function with variables $\{x, u, ..., \underset{q}{u}\}$. The set of all differential functions on J_q is denoted by $\mathcal{A}_q$. For any differential function f, we define its differential order to be $\min\{q : f \in \mathcal{A}_q\}$, denoted by $\mathrm{ord}(f)$. We say a system of differential functions F has (differential) order q, if $\max\{\mathrm{ord}(f) : f \in F\} = q$.

We introduce the *Formal Total Derivative* operator for each independent variable x_i to act on members of Ω by a unit increment of the i-th index of their vector subscript: $\mathbf{D}_i u^j_\alpha := u^j_{\alpha+1_i}$ where $\alpha + 1_i = (\alpha_1, ..., \alpha_i + 1, ..., \alpha_n)$.

The action of $\mathbf{D}_i$ to a differential function is defined to be:

$$\mathbf{D}_i = \frac{\partial}{\partial x_i} + \sum_{u^j_\alpha \in \Omega} \mathbf{D}_i u^j_\alpha \frac{\partial}{\partial u^j_\alpha}$$

So $\mathbf{D}_i : \mathcal{A}_q \to \mathcal{A}_{q+1}$. It is convenient to extend the multi-index notation for formal jet variables to formal total derivatives:

$$\mathbf{D}^\alpha := (\mathbf{D}_1)^{\alpha_1} \cdots (\mathbf{D}_n)^{\alpha_n} \tag{1.2.7}$$

Clearly, if $f \in \mathcal{A}_q$ then $\mathbf{D}^\alpha f \in \mathcal{A}_{q+|\alpha|}$. We also introduce another convenient notation: $\mathbf{D}^r f := \{\mathbf{D}^\alpha f : |\alpha| = r\}$ and define $\mathbf{D}^0 f := f$. Naturally, we can apply this notation to a system $F \subset \mathcal{A}_q$ by defining $\mathbf{D}^r F := \{\mathbf{D}^r f : f \in F\}$.

Differential equations R of order q are often stated as the kernel of a set of differential functions in $\mathcal{A}_q$ with order q:

$$Z(R) := \{(x, u^j_\alpha) \in J_q(\mathbb{F}^n, \mathbb{F}^m) : R_k(x, u^j_\alpha) = 0, k = 1, ..., \ell\} \tag{1.2.8}$$

A *solution* of R over an open set $U \subset X$ are m analytic functions $f^j(x)$, $j = 1, ..., m$, such that for each point $x \in U$, $(x, u^j_\alpha) \in Z(R)$, where $u^j_\alpha = \mathbf{D}^\alpha f^j$.

Remark 1.2.2. *If we intend to view systems of* PDE *as geometric objects, we also call a fibred sub-manifold $Z(R)$ of $J_q(\mathcal{E})$ a* PDE *of order q on $\mathcal{E}$. In the view of geometry, a solution of $Z(R)$ is a local section f of $\mathcal{E}$ over an open set $U \subset X$ such that $j_q(f)_x \in Z(R)$, for any $x \in U$.*

1.2.3 Formal Power Series Solutions

An analytic solution of R in a neighborhood of x^0 can be written as the power series:

$$u^j = f^j(x) = \sum_{|\alpha|=0}^{\infty} \frac{c^j_\alpha}{\alpha!}(x - x^0)^\alpha \tag{1.2.9}$$

The coefficient c^j_α is equal to the value of $\mathbf{D}^\alpha f^j$ at the point x^0 corresponding to the jet variable u^j_α. So we can consider the power series of a solution up to order q as a point in $Z(R)$.

The inverse question is: for each point in $Z(R)$, can we construct a power series solution at this point? Generally, this is not true.

EXAMPLE 1.2.2. *Let us consider a system R, where $\ell = 2, m = 2, n = 2, q = 1$, given by*

$$u_x - v = 0, u_y - x = 0 \tag{1.2.10}$$

At first glance we might expect that all the points in $Z(R)$ are consistent points at which we can construct power series solutions. However there is a hidden constraint in J_1: $\boldsymbol{D}_y(u_x - v) - \boldsymbol{D}_x(u_y - x) = -v_y + 1 = 0$. Thus some points in $Z(R)$, which do not satisfy this hidden constraint, are not consistent.

Thus the construction of a power series solution of a PDE system R order by order can be performed only if R contains all its integrability conditions. The systems having such properties are called *formally integrable* systems. To study these systems we need to introduce two basic operators on Jet Space.

Definition 1.2.6. *[Prolongation] Let R be a system of* PDE *with order q. Its r-th prolongation is defined to be:*

$$R^{(r)} := \{R, \boldsymbol{D}R, \boldsymbol{D}^2R, ..., \boldsymbol{D}^rR\} \tag{1.2.11}$$

Remark 1.2.3. *If the equations of R have different differential order, we first need to prolong each equation up to order q, which is given by:*

$$\{\boldsymbol{D}^pR_k : p = q - \mathrm{ord}(R_k), k = 1, ..., \ell\}$$

Applying (1.2.8) to $R^{(r)}$, we have the zero set of a prolonged differential system. For example, let $R = u_x^2 + u_x - u = 0$. Then applying the formal total derivatives $\mathbf{D}_x$ and $\mathbf{D}_y$ gives:

$$Z(R^{(1)}) = \{(x, y, u, \underset{1}{u}, \underset{2}{u}) \in J_2 : u_x^2 + u_x - u = 2u_xu_{xx} + u_{xx} - u_x = 2u_xu_{xy} + u_{xy} - u_y = 0\}$$

Prolongation lifts the locus of a PDE system from lower order Jet Space to higher order Jet Space. An inverse operation, so-called *projection*, maps the locus from higher order Jet Space to lower order Jet Space.

Definition 1.2.7. *[Projection] Given a* PDE R *in* J_{q+r}*, the projection of* R *from* J_{q+r} *to* J_q *is:*

$$\boldsymbol{\pi}_q^{q+r} Z(R) := \{(x, u, \underset{1}{u}, \ldots, \underset{q}{u}) \in J_q : \exists (x, u, \underset{1}{u}, ..., \underset{q}{u}, ..., \underset{q+r}{u}) \in Z(R)\}.$$

This geometric operator cannot be algorithmized directly. But it could be translated to the analogous algebraic versions. Suppose R is a system in J_q and $R^{(1)}$ is its prolonged system. Then after we eliminate the $(q+1)$-order jet variables in $R^{(1)}$, we have an algebraic representation for the projection, denoted by $\boldsymbol{\alpha} R^{(1)}$.

It is easy to demonstrate that projecting the prolongation of a differential system R in J_q may not return the original system but a subset thereof:

$$\boldsymbol{\pi}_q^{q+r} Z(R^{(r)}) \subseteq Z(R), \text{ for any } r \in \mathbb{N} \tag{1.2.12}$$

If it is only a proper subset of $Z(R)$, then there are extra constraints, which we call *integrability conditions*. They are differential rather than algebraic consequences of the original system. If for some system we cannot find any new constraints by differentiation, then naturally we introduce the following concept:

Definition 1.2.8. *[Formally Integrable System] A differential system* R *with order* q *is formally integrable, if* $\boldsymbol{\pi}_{q+r}^{q+r+1} Z(R^{(r+1)}) = Z(R^{(r)})$ *for any* $r \in \mathbb{N}$.

This definition requires that for any r, the projections and prolongations will not produce any new constraints. However verifying formal integrability by direct use of Definition 1.2.8 requires checking infinitely many conditions. For finite implementation, the geometric approach needs to be complemented by some algebraic tools. To produce a finite test, we now briefly describe involution and Spencer Cohomology theory.

1.2.4 Involution

We now turn to the consideration of a subset of formally integrable systems known as involutive systems. Two facts make this class of systems interesting and useful. Firstly, it is possible to determine whether a given system is involutive using only a finite number of operations. Secondly, for any system it is possible to produce an involutive form with the same solution space using only a finite number of operations.

Now let us consider a single prolongation of a system of PDE R with order q. First we study the local structure of $Z(R^{(1)})$ by looking at the tangent space at a point $p \in Z(R^{(1)})$. The local dimension of $Z(R^{(1)})$ is given by the dimension of the tangent space, which is the null-space of the Jacobian matrix at p. This Jacobian matrix can be divided into four blocks:

$$\begin{pmatrix} \frac{\partial(\mathbf{D}R)}{\partial\ u_{q+1}} & \frac{\partial(\mathbf{D}R)}{\partial u_s} \\ 0 & \frac{\partial R}{\partial u_s} \end{pmatrix}, \tag{1.2.13}$$

where $0 \leq s \leq q$.

Definition 1.2.9. *[Symbol] Consider a* PDE *system R of order q. The matrix $\left(\frac{\partial(\boldsymbol{D}R)}{\partial\ u_{q+1}}\right)$ is called the symbol matrix of $R^{(1)}$, and is denoted by $\mathcal{S}^{(1)}$. The kernel of the symbol matrix is called the symbol of $R^{(1)}$, and is denoted by g_{q+1}.*

Similarly we define the symbol matrix of R to be $\left(\frac{\partial R}{\partial u_q}\right)$ and prolonged symbols are defined similarly. To avoid introducing laborious notation, we denote the symbol and the symbol matrix of R by g_q and $\mathcal{S}$ respectively.

The Symbol can be regarded as a way to test for the existence of integrability conditions. We first introduce Cartan's test, a straightforward method, for checking involutivity of the symbol. This method depends on local coordinates.

The multi-index $\alpha = (\alpha_1, ..., \alpha_n) \in \mathbb{N}^n$ is said to be of *class k* if the first non-zero entry of α is α_k. We know each column of the symbol matrix $\mathcal{S}$ corresponds to a jet variable u^j_α. So we order the columns of with higher class so that they appear to the left of those with lower class and transform $\mathcal{S}$ to row echelon form, then define

$$\beta^k := \text{number of rows of class } k \tag{1.2.14}$$

appearing in the echelon form.

These numbers depend on the local coordinates. But we can choose a coordinate system in which the sums $\sum_{i=k}^n \beta^i$, $k = n, .., 1$ take their maximum values successively. Such a coordinate system is called *δ-regular* (almost all coordinate systems are δ-regular).

Now we apply prolongation only to the pivot equations. If the pivot is class k, then we apply partial differentiation with respect to $x_1, ..., x_k$. Then the new equations denoted by R' are a subset of $\mathbf{D}R$. And the rank of symbol matrix of R' is $\sum_{k=1}^n k\beta^k$. Since the prolongation of the whole system should contain more equations than R', in general we have $\text{rank}\mathcal{S}^{(1)} \geq \sum_{k=1}^n k\beta^k$.

If

$$\text{rank}\mathcal{S}^{(1)} = \sum_{k=1}^{n} k\beta^k, \tag{1.2.15}$$

then the coordinate system is *δ-regular* and $\mathcal{S}$ is said to be *involutive* [34, 45]. The test (1.2.15) is called the *Cartan test*. Obviously, this definition depends on local coordinates. Fortunately a generic set of coordinates is δ-regular, so the Cartan test

can be applied after (potentially very expensive) generic linear change of coordinates [34].

1.2.5 Spencer Cohomology

The famous Cartan test introduced above for involution of the Symbol requires checking a condition involving some integers β_k, which unfortunately are coordinate-dependent. In the 1960s, Spencer introduced an intrinsic definition of involutivity of the symbol. His definition was expressed in terms of the exactness of the so-called δ operator on certain sequences involving the symbols of the system, which employed concepts from differential geometry and homological algebra. Later, the theory was thoroughly studied by Spencer [53], Quillen [37] and Goldschmidt [11]. This definition is very formal but it is convenient from a theoretical point of view because of its coordinate-independence.

For a base space X with dimension n, we adopt the standard notations of T^* for the cotangent bundle over X. And we denote $\mathcal{E}$ (with $\dim \mathcal{E} = m + n$) a fibred bundle over X, $\mathcal{V}(\mathcal{E})$ for the vertical jet bundle (see [34] for the definition), $\mathbf{S}^p T^*$ (with $\dim \mathbf{S}^p T^* = \binom{n+p-1}{p}$) and $\mathbf{\Lambda}^r T^*$ (with $\dim \mathbf{\Lambda}^r T^* = \binom{n}{r}$) for the bundle of symmetric tensors and skew-symmetric tensors over X respectively.

A basis of $\mathbf{S}^p T^*$ can be represented by v_α, where $\alpha = (\alpha_1, ..., \alpha_n) \in \mathbb{N}^n$, and $|\alpha| = \sum_i \alpha_i = p$ (corresponding to all the monomials with degree p). For an integer i with $1 \leq i \leq n$, we define $\alpha \pm 1_i := (\alpha_1, ..., \alpha_i \pm 1, ..., \alpha_n)$. When $p < 0$, we define $\mathbf{S}^p T^* := 0$.

The basis elements of $\mathbf{\Lambda}^r T^*$ are denoted by $dx^\mu = dx_{\mu_1} \wedge \cdots \wedge dx_{\mu_r}$, where μ is a sequence of integers $1 \leq \mu_1 < \cdots < \mu_r \leq n$ with $|\mu| = r$ (the number of elements of μ is r). When $r > n$, $\mathbf{\Lambda}^r T^* = 0$.

Consider a PDE system R of order q. Intrinsically, the symbol g_p $(p < q)$ is a family of vector spaces in $\mathbf{S}^p T^* \otimes \mathcal{V}(\mathcal{E})$ defined by:

$$g_p := \begin{cases} 0, & \text{if } p < 0; \\ \mathbf{S}^p T^* \otimes \mathcal{V}(\mathcal{E}), & \text{if } 0 \leq p < q. \end{cases} \tag{1.2.16}$$

In local coordinates, the rth prolongation of the symbol equations is given as

$$\left\{ \sum_{1 \leq j \leq m, |\alpha| = q} \frac{\partial R_k}{\partial u^j_\alpha} v^j_{\alpha+\beta} : |\beta| = r, k = 1, ..., \ell \right\}. \tag{1.2.17}$$

The rth prolonged symbol g_{q+r} is the kernel of these equations.

Definition 1.2.10. *[δ-operator] With $|\mu| = r$ and $|\alpha| = p$ fixed, the δ-operator is a linear map: $\mathbf{\Lambda}^r T^* \otimes \mathbf{S}^{p+1} T^* \to \mathbf{\Lambda}^{r+1} T^* \otimes \mathbf{S}^p T^*$. Let $\omega = \omega_{\mu,\alpha} dx^\mu \otimes v_\alpha \in \mathbf{\Lambda}^r T^* \otimes \mathbf{S}^{p+1} T^*$*

then

$$\delta(\omega) := \sum_{i,\alpha_i>0} \omega_{\mu,\alpha}(dx^\mu \wedge dx_i) \otimes v_{\alpha-1_i} \tag{1.2.18}$$

Proposition 1.2.2. *For any integer k, the δ sequence*

$$0 \longrightarrow \mathbf{S}^p T^* \longrightarrow \mathbf{\Lambda} T^* \otimes \mathbf{S}^{p-1} T^* \longrightarrow \cdots \longrightarrow \mathbf{\Lambda}^n T^* \otimes \mathbf{S}^{p-n} T^* \longrightarrow 0 \tag{1.2.19}$$

is exact.

Since the vector bundle $\mathcal{V}(\mathcal{E})$ is flat, consequently the tensor product retains the exactness and similarly we obtain the exact sequence

$$0 \longrightarrow \mathbf{S}^p T^* \otimes \mathcal{V}(\mathcal{E}) \longrightarrow \mathbf{\Lambda} T^* \otimes \mathbf{S}^{p-1} T^* \otimes \mathcal{V}(\mathcal{E}) \longrightarrow$$

$$\cdots \longrightarrow \mathbf{\Lambda}^{n-1} T^* \otimes \mathbf{S}^{p-n+1} T^* \otimes \mathcal{V}(\mathcal{E}) \longrightarrow \mathbf{\Lambda}^n T^* \otimes \mathbf{S}^{p-n} T^* \otimes \mathcal{V}(\mathcal{E}) \longrightarrow 0$$

Now by [53, 34] we restrict δ to the space $g_{q+r} \subset \mathbf{S}^{q+r} T^* \otimes \mathcal{V}(\mathcal{E})$ obtaining

$$\delta : \mathbf{\Lambda}^p T^* \otimes g_{q+r} \to \mathbf{\Lambda}^{p+1} T^* \otimes g_{q+r-1} \tag{1.2.20}$$

and a complex

$$0 \longrightarrow g_{q+r} \longrightarrow \mathbf{\Lambda} T^* \otimes g_{q+r-1} \longrightarrow \cdots \longrightarrow \mathbf{\Lambda}^n T^* \otimes g_{q+r-n} \longrightarrow 0\,. \tag{1.2.21}$$

In general this complex may not be exact. The cohomology H^p_{q+r-p} which is the quotient space $\ker\delta/\mathrm{im}\delta$ at $\mathbf{\Lambda}^p T^* \otimes g_{q+r-p}$, is called *Spencer Cohomology.*

Definition 1.2.11. *[Involutive Symbol] We say the symbol g_q is involutive if $H^p_{q+r-p} = 0$ for all $0 \le p \le n$ and all $r \ge p$.*

Remark 1.2.4. *Spencer Cohomology groups are dual to the homology groups of a Koszul complex [13, 11]. Serre showed that this intrinsic definition is equivalent to Cartan's definition in a letter to Guillemin and Sternberg [13]. Following Serre's idea, Singer and Sternberg gave a rigorous proof in [46]. Other discussion about involutivity can be found in Pommaret's book [34]. Thus the operational and intrinsic definitions of involutivity agree.*

We use an example due to Mansfield [29] to show how to perform the involutivity test by using these two definitions. She also showed the connection between involutivity and *Differential Gröbner Bases* in [29].

EXAMPLE 1.2.3. *Let R be a* PDE *with one dependent variable and three independent variables:*

$$\begin{cases} u_{yz} - u_{xx} = 0 \\ u_{zz} - u_{xz} = 0 \end{cases} \tag{1.2.22}$$

The symbol equations are $\{v_{yz} - v_{xx} = 0, v_{zz} - v_{xz} = 0\}$*, which can be equivalently expressed as the symbol matrix times a vector of monomial basis. The kernel can be expressed as the following basis*

$$g_2 = \langle v_{yz} + v_{xx}, v_{zz} + v_{xz}, v_{xy}, v_{yy} \rangle \tag{1.2.23}$$

Similarly, we compute a basis of g_3

$$g_3 = \langle v_{yyz} + v_{xxy}, v_{xyy}, v_{yyy}, v_{xxx} + v_{xyz} + v_{xxz} + v_{yzz} + v_{xzz} + v_{zzz} \rangle \tag{1.2.24}$$

Using the above basis and $\mathbf{S}T^* = \{v_x, v_y, v_z\}$*,* $\mathbf{\Lambda}T^* = \{dx, dy, dz\}$*,* $\mathbf{\Lambda}^2T^* = \{dx \wedge dy, dx \wedge dz, dy \wedge dz\}$ *and* $\mathbf{\Lambda}^3T^* = \{dx \wedge dy \wedge dz\}$*, we compute the cohomology at* $\mathbf{\Lambda}^2T^* \otimes g_2$*:*

$$\mathbf{\Lambda}T^* \otimes g_3 \longrightarrow \mathbf{\Lambda}^2T^* \otimes g_2 \longrightarrow \mathbf{\Lambda}^3T^* \otimes \mathbf{S}T^* \longrightarrow 0 \tag{1.2.25}$$

Computing the image of $\delta(\mathbf{\Lambda}T^* \otimes g_3)$ *yields an* 8*-dimensional space. On the other hand, the dimension of* $\mathbf{\Lambda}^3T^* \otimes \mathbf{S}T^*$ *is* 3*, and the dimension of* $\mathbf{\Lambda}^2T^* \otimes g_2$ *is* 12*. Since* $\delta(\mathbf{\Lambda}^2T^* \otimes g_2)$ *is surjective, the kernel has dimension* 9*. Hence* $H^2_{2+2-2} \neq 0$ *and* g_2 *is not involutive.*

Now let us apply Cartan's test (1.2.15) to this example: $\beta^3 = 1, \beta^2 = 1, \beta^1 = 0$*, so* $\sum_{k=1}^{3} k\beta^k = 5$*, which is less than* 6*, the rank of* $\mathcal{S}^{(1)}$*. So these two results are consistent.*

Apparently Cartan's test is more convenient. However, if the coordinates of a given system are not δ-regular, then Cartan's test may fail.

EXAMPLE 1.2.4. *Let us consider a single involutive* PDE $u_{xy} = 0$ *with one dependent variable and two independent variables. It is easy to check that* $\beta^1 = 1, \beta^2 = 0$ *and* $\sum_{k=1}^{2} k\beta^k = 1$ *but* $\operatorname{rank}\mathcal{S}^{(1)} = 2$*, so Cartan's test fails. The reader can check that after a generic linear change of coordinates:* $(x, y) \mapsto (ax + by, cx + dy)$*, it succeeds.*

Applying the Spencer Cohomology test, we have $\dim g_{2+r} = 2$*, for* $r \geq 0$*. The complex:*

$$0 \longrightarrow g_{2+r} \longrightarrow \mathbf{\Lambda}T^* \otimes g_{2+r-1} \longrightarrow \mathbf{\Lambda}^2T^* \otimes g_{2+r-2} \longrightarrow 0 \tag{1.2.26}$$

is exact, since $\dim(g_{2+r}) + \dim(\mathbf{\Lambda}^2T^* \otimes g_{2+r-2}) = 2 + 2 = \dim(\mathbf{\Lambda}T^* \otimes g_{2+r-1}) = 2 \times 2$*. This indicates that* g_2 *is involutive.*

Remark 1.2.5. *We briefly discuss a combination of these two methods for testing of involutivity of the symbol. We know Cartan's test gives a sufficient condition. On the other hand it is impossible to check the exactness of all* δ *sequences in Spencer's Test. But the exactness of each* δ *sequence is a necessary condition.*

We first use Cartan's test. If it fails then there are two possibilities: either the symbol is not involutive, or the symbol is involutive but the coordinates are not δ*-regular. Then we choose one* δ *sequence from (1.2.21) and check its exactness. If*

it is not exact, then we know the symbol is not involutive, and further prolongations are needed. Otherwise, a random change of coordinates can be launched to restore δ-regularity.

Theorem 1.2.1. *[δ-Poincare Lemma] Let g_q be the symbol of a q-th order system R in $J_q(\mathcal{E})$ with m dependent variables and n independent variables. Then there exists an integer $q' = \theta(m, n, q)$ such that $g_{q'}$ is involutive.*

Remark 1.2.6. *A bound can be calculated by the following recursive formula:*

$$\begin{cases} \theta(m,0,1) & = 0; \\ \theta(m,n,1) & = m \cdot \binom{a+n}{n-1} + a + 1, \ \textit{where } a = \theta(m, n-1, 1); \\ \theta(m,n,q) & = \theta(m,b,1), \ \textit{where } b = \sum_{i=0}^{q} \binom{n+i-1}{n-1} \cdot m. \end{cases} \tag{1.2.27}$$

For a proof see Sweeney [54]. Although this bound is impractically large, it states that one can produce an involutive symbol for any PDE *after finitely many steps of prolongations.*

Definition 1.2.12. *[Involutive System] A differential system R is said to be involutive, if it is formally integrable and its symbol is involutive.*

Apparently, "involutivity" is a stronger concept than formal integrability. However it is easier to test involutivity by using the following criterion.

Theorem 1.2.2. *[Criterion for Involution] A differential system R of order q is in involution if and only if its symbol is involutive and $\boldsymbol{\pi} Z(R^{(1)}) = Z(R)$.*

This criterion was formulated precisely by D. Quillen in his PhD Thesis [37] for linear systems of PDE and H. Goldschmidt [12] extended it to nonlinear PDE using Spencer Cohomology theory.

1.2.6 Cartan Kuranishi Completion

By the Cartan-Kähler theorem [67], involutive systems are locally solvable and allow an existence and uniqueness theorem. The Cartan-Kuranishi prolongation theorem states, roughly, that given an exterior differential system after finitely many prolongations, it becomes either involutive or incompatible. The finiteness of the following famous completion procedure was first conjectured by Cartan, and finally proved by Kuranishi [22]. See Malgrange's work [27] for a recent discussion.

Procedure 1.2.1. $R = \boldsymbol{CK}(R)$
Input R
Repeat
  while $\mathcal{S}$ *is not involutive* ***repeat*** $R := R^{(1)}$
  while $Z(R) \neq \boldsymbol{\pi} Z(R^{(1)})$ ***repeat*** $R := \boldsymbol{\alpha} R^{(1)}$

end loop
Output R

The first loop is the prolongation stage to make the symbol involutive, and will be finite by Theorem 1.2.1.

The second loop is the projection stage to check if there are new constraints. At a point of $Z(R)$, the symbol equations can be considered as A-module, a submodule of A^m, where A is the polynomial ring $\mathbb{F}[x_1, ..., x_n]$ and m is the number of dependent variables. When we find a new constraint after proper steps of prolongations it will add a new generator to the (symbol) module. Subsequently the symbol modules consist of an ascending chain. Because A^m is Noetherian, the chain will terminate. Hence the second loop will stop after finitely many steps.

Combining the two loops together we can complete R to an involutive form in finitely many steps by Theorem 1.2.2.

1.2.7 Other General Methods for PDE

The spirit of this thesis is to develop the theory and algorithms for Numerical Jet Geometry in the framework of Cartan's geometric study of PDE. However it is necessary to give brief introduction to other methods since they are still very useful for our algorithm in [39] and our theory in [63].

Differential algebra was introduced by Ritt [42] and developed by Kolchin [21], and is a generalization of classical commutative algebra. After completing the ring structure with a set of commutative derivations $\Delta = \{\delta_1, ..., \delta_n\}$, we can define differential polynomials, ideals, fields, modules and algebras in a straightforward way. A natural attempt is to develop differential analogues of Buchberger's algorithm for systems of polynomially nonlinear PDE. Carra-Ferro [4] and Ollivier [32] gave definitions of such Differential Gröbner Bases. However these bases could be infinite (unlike the case for polynomial algebraic equations and linear PDE systems). In her PhD Thesis, Mansfield [28] gave an algorithm which used pseudo-reduction instead of reduction to attempt to construct Differential Gröbner Bases. It has proved very useful in applications [29]. In a breakthrough work by Boulier et. al. [3], they gave an algorithm which performs binary splitting on the *initials* and *separants* and rigorously proved that the resulting system of cases gives a representation of the radical of the differential ideal generated by the system (but not the differential ideal).

Besides the algebraic methods mentioned above, there are some analytic approaches. For example Rust et al. [43, 44] give analytic differential elimination methods to complete analytic systems of PDE to desired forms. The desired forms are *Riquier Bases* and *reduced involutive forms*, which state the existence and uniqueness of formal power series solutions. Such methods are more related to our work in Chapter 2 and 4, so we show some details here.

First we introduce the concept of ranking, which is vital in all the symbolic differential elimination methods. A *positive ranking* [44] $\prec$ of all the partial derivatives Ω is a total ordering on Ω which satisfies:

$$v_\alpha \prec v_\beta \Rightarrow \frac{\partial v_\alpha}{\partial x_i} \prec \frac{\partial v_\beta}{\partial x_i}; \tag{1.2.28}$$

$$v_\alpha \prec \frac{\partial v_\alpha}{\partial x_i}, \tag{1.2.29}$$

for any independent variable x_i and $v_\alpha, v_\beta \in \Omega$.

Let us consider an example with one dependent variable u and two independent variables x, y. By the positivity of ranking (1.2.29), we have $u \prec u_x \prec u_{xx} \prec \cdots$. If we let $u_x \prec u_y$ and use the total degree, then a ranking is determined by the condition (1.2.28):

$$u \prec u_x \prec u_y \prec u_{xx} \prec u_{xy} \prec u_{yy} \prec \cdots \tag{1.2.30}$$

It can be checked that this satisfies all the axioms of rankings. For a theory and classification of rankings see Rust [44].

Let $\text{HD}f$ denote the highest derivative of f in Ω with respect the ranking $\prec$. We say that f is *leading linear with respect to a ranking* $\prec$ if f has the form $f = h \cdot \text{HD}f + g$, with $\text{HD}g \prec \text{HD}f$ and $\text{HD}h \prec \text{HD}f$. Otherwise we say f is *leading nonlinear with respect to a ranking* $\prec$. In addition, we say that f is $\prec$*-monic with respect to a ranking* $\prec$ if f is leading linear and $h = 1$.

The *principal derivatives* of a finite set $\mathcal{M}$ of $\prec$-monic analytic functions are defined as

$$\text{Prin}\mathcal{M} := \{v \in \Omega | \exists f \in \mathcal{M} \text{ and } \alpha \in \mathbb{N}^n \text{ with } v = \text{HD}\mathbf{D}^\alpha f\} \tag{1.2.31}$$

The *parametric derivatives* of $\mathcal{M}$ are those derivatives that are not principal, denoted by $\text{Par}\mathcal{M}$.

The parametric and principal derivatives enable us to specify initial data, that will be important in the Existence and Uniqueness Theorem 1.2.3.

We define a *specification of initial data* for $\mathcal{M}$ to be a map

$$\phi : \{x\} \cup \text{ Par } \mathcal{M} \to \mathbb{F}$$

For $x^0 \in \mathbb{F}^m$, we say that ϕ is a specification at x^0 if

$$\phi(x) := (\phi(x_1), \phi(x_2), \cdots, \phi(x_m)) = x^0.$$

For an analytic function g on Jet Space, let $\phi(g)$ be the function of the principal derivatives obtained from g by evaluating x and the parametric derivatives using ϕ:

$$\phi(g) := g(\phi(x), (\phi(u))_{u \in Par\mathcal{M}}).$$

Given a ranking $\prec$ of partial derivatives, Riquier bases are in solved form with respect to their leading derivatives (a set of $\prec$-monic analytic functions). They are determined by successively including integrability conditions and performing eliminations on the resulting systems. The solved form requirement means that in the exact case they are essentially restricted to PDE which are linear in their highest derivatives.

Definition 1.2.13 (Riquier Basis)**.** *$\mathcal{M}$ is called a Riquier Basis if for all $\alpha, \alpha' \in \mathbb{N}^m$ and $f, f' \in \mathcal{M}$ with $\mathrm{HD}\boldsymbol{D}^{\alpha} f = \mathrm{HD}\boldsymbol{D}^{\alpha'} f'$, the integrability condition $\boldsymbol{D}^{\alpha} f - \boldsymbol{D}^{\alpha'} f'$ is reduced to zero by a sequence of one-step reductions by members of $\mathcal{M}$.*

See [43] for the definition of one-step reduction used above. A fundamental property of Riquier Bases is:

Theorem 1.2.3. *[Formal Riquier Existence Theorem] Let $\mathcal{M}$ be a Riquier Basis such that each $f \in \mathcal{M}$ is polynomial in the principal derivatives. For $x^0 \in \mathbb{F}^n$, let ϕ be a specification of initial data for $\mathcal{M}$ at x^0 such that $\phi(f)$ is well-defined for all $f \in \mathcal{M}$. Then there is formal power series solution $u(x) \in \mathbb{F}[[x - x^0]]^n$ to $\mathcal{M}$ at x^0 such that $\boldsymbol{D}^{\alpha} u^i(x^0) = \phi(u^i_{\alpha})$ for all $u^i_{\alpha} \in Par\mathcal{M}$. Furthermore, every formal power series solution to $\mathcal{M}$ at x^0 may be obtained in this way for some ϕ.*

The solved form requirement of Riquier Bases means that they cannot be directly applied to general nonlinear systems. But we know that any PDE is either linear or nonlinear in its leading derivative with respect to a ranking. Furthermore any leading nonlinear PDE after differentiation with respect to any independent variable becomes leading linear in its leading derivative. A general algorithm to compute all the integrability conditions, developed by Reid et. al. [40, 61], called **rifsimp**, performs linear eliminations amongst the leading linear PDE in the same way as the standard form algorithm in the case of linear systems. The **rifsimp** algorithm terminates when no new equations are generated relative to the given system. We say an equation is *new* if it lowers the dimension of the existing system regarded as a submanifold of its Jet Space.

The output of **rifsimp** is called *reduced involutive form* (rif) which is related to the concept of involution as we introduced before.

Significant results have been achieved by algebraic and analytic approaches for general exact PDE. However, we will not use such rewriting techniques directly in this Thesis because of the inherent instability caused by rankings on approximate systems. The geometric methods we have discussed are our main tools. Paradoxically we develop our theory by using rankings and Riquier Bases in Chapter 4. However no rankings and no eliminations appear in our algorithm and a Riquier Basis is obtained in an implicit form. Thus the method in Chapter 4 is different from those symbolic methods discussed above.

The main theme of this Thesis is dominated by a rhythm of geometry. In the next section on Numerical Algebraic Geometry, this theme will again be highly emphasized.

1.3 Numerical Algebraic Geometry

The most basic nonlinear functions are polynomials. They provide a better approximation of nonlinear phenomena than linear equations. For systems of PDE if the differential equations are polynomials with respect to the jet variables, we say they are *polynomially* nonlinear differential equations. If we choose $\mathbb{C}$ as our underlying field, the zero sets defined by such systems in Jet Space are jet varieties. Naturally we can use the computational techniques of algebraic geometry to study these jet varieties.

While much of algebraic geometry is concerned with abstract and general statements about varieties, methods for effective computation with concretely-given polynomials have also been developed. A very important class of such techniques are provided by Buchberger's Algorithm (see Buchberger's Thesis 1965), which transforms polynomial systems to the form of Gröbner bases (a generalization of the Gauss Algorithm for row reduced form).

Today Buchberger's algorithm and many improved versions are employed in most computer algebra systems. But it cannot be applied to approximate systems directly. The main reason is that Gröbner bases are discontinuous with the input and they depend on an ordering which can cause numerical instability (the instability problem caused by Gaussian Elimination is a special case). In addition, the worst case complexity for computing Gröbner bases is double exponential.

Recently, a new area, "Numerical Algebraic Geometry", was initiated by Andrew Sommese and Charles Wampler and is developing rapidly. It bears the same relation to "Algebraic Geometry" that "Numerical Linear Algebra" bears to "Linear Algebra". This Thesis mainly aims to develop numerically stable methods for general PDE. So we give a brief introduction to this new area in this section. An elegant and introductory description of this area can be found in Andrew Sommese and Charles Wampler's 2005 book [52].

1.3.1 Homotopy Continuation Methods

Homotopy continuation methods play a fundamental role in Numerical Algebraic Geometry and provide an efficient and stable way to compute all isolated roots of polynomial systems. Verschelde implemented these methods in his software package PHCpack [56].

The basic idea is to embed the target system into a family of systems continuously depending on parameters. Then each point in the parameter space corresponds to a set of solutions. Suppose we know the solutions at a point. Then we can track

the solutions from this starting point to the point representing the target system we want to solve.

First let us look at the simplest case: a univariate polynomial $f(z)$ with degree d. We know that $f(z)$ has d roots in $\mathbb{C}$ (counting multiplicities). Of course we can embed $f(z)$ into the family $a_d z^d + a_{d-1} z^{d-1} + \cdots + a_0$, where the a_i are parameters. Now choose a start point corresponding to $z^d - 1$ in this parameter space, whose roots are

$$z_k^0 = e^{2k\pi\sqrt{-1}/d},\ k = 0, 1, ..., d-1 \tag{1.3.1}$$

Then we use a real straight line in the parameter space to connect $z^d - 1$ with $f(z)$:

$$H(z,t) := tf(z) + (1-t)(z^d - 1). \tag{1.3.2}$$

This form is a subclass of the family depending on only one real parameter $t \in [0,1]$.

When $t = 0$ we have the start system $H(z,0) = z^d - 1$ and when $t = 1$ we have our target system $H(z,1) = f(z)$. An important question is to show how to track individual solutions as t changes from 0 to 1. Let us look at the tracking of the solution z_k (the k-th root of $f(z)$). When t changes from 0 to 1, it describes a curve, which is function of t, denoted by $z_k = z_k(t)$. So $H(z_k(t), t) \equiv 0$ for all $t \in [0,1]$. Consequently, we have

$$0 \equiv \frac{dH(z_k(t),t)}{dt} = \frac{\partial H(z,t)}{\partial z}\frac{dz_k(t)}{dt} + \frac{\partial H(z,t)}{\partial t}. \tag{1.3.3}$$

This problem is reduced to an ODE for the unknown function $z_k(t)$ together with an algebraic constraint $H(z_k(t), t) \equiv 0$. The initial condition is the start solution $z_k(0) = z_k^0$ and $z_k(1)$ is a solution of our target problem $f(z) = 0$.

Remark 1.3.1. *In the book [2], Blum, Smale et al. show that on average an approximate root of a generic polynomial system can be found in polynomial time. Also application of the polynomial cost method for numerically solving differential algebraic equations [18] gives polynomial cost method for solving homotopies.*

But there is a prerequisite for the continuous tracking: $\frac{\partial H(z,t)}{\partial z} \neq 0$ along the curve $z = z_k(t)$. If the equations $z - z_k(t) = 0$ and $tf'(z) + d(1-t)z^{d-1} = 0$ have intersection at some point $(t, z_k(t))$, then we cannot continue the tracking. There is way to avoid this singular case, called the "gamma trick" that was first introduced in [31]. We know two complex curves almost always have intersections at complex points, but here t must be real. So if we introduce a random complex transformation to the second curve, the intersection points will become complex points and such a singularity will not appear when $t \in [0,1)$. Let us introduce a random angle $\theta \in [-\pi, \pi]$ and modify the homotopy (1.3.2) to

$$H(z,t) := tf(z) + e^{i\theta}(1-t)(z^d - 1). \tag{1.3.4}$$

It is easy to show that the k-th starting solution is still z_k^0 in (1.3.1) and that $z_k(1)$ is still a root of $f(z)$.

Genericity and Probability One

In an idealized model where paths are tracked exactly and the random angle can be generated to infinite precision, the homotopy (1.3.4) can be proved to succeed "with probability one". To clarify this statement, it is necessary to use a fundamental concept in algebraic geometry: *genericity*.

Definition 1.3.1 (Generic)**.** *Let X be an irreducible algebraic variety. We say a property P holds **generically** on X, if the set of points of X that do not satisfy P are contained in a proper subvariety Y of X. The points in $X\backslash Y$ are called **generic** points.*

The set $X\backslash Y$ is called a Zariski open set of X. Roughly speaking, if Y is a proper subvariety of an irreducible variety X and p is a random point on X with uniform probability distribution, then the probability that $p \notin Y$ is one. So we can consider a random point as a generic point on X without a precise description of Y. Many of the desirable behaviors of homotopy continuation methods rely on this fact.

Coefficient-Parameter Homotopy

There are several versions of the Coefficient-Parameter theorem in [52]. Here we only state the basic one.

Theorem 1.3.1. *Let $F(z;q) = \{f_1(z;q), ..., f_n(z;q)\}$ be a polynomial system in n variables z and m parameters q. Let $\mathcal{N}(q)$ denote the number of nonsingular solutions as a function of q:*

$$\mathcal{N}(q) := \#\left\{z \in \mathbb{C}^n : F(z;q) = 0,\ \det\left(\frac{\partial F}{\partial z}(z;q)\right) \neq 0\right\} \tag{1.3.5}$$

Then,

1. *There exist N, such that $\mathcal{N}(q) \leq N$ for any $q \in \mathbb{C}^m$. Also $\{q \in \mathbb{C}^m : \mathcal{N}(q) = N\}$ is a Zariski open set of $\mathbb{C}^m$. The exceptional set $Y = \{q : \mathcal{N}(q) < N\}$ is an affine variety contained in a variety with dimension $m-1$.*

2. *The homotopy $F(z;\phi(t)) = 0$ with $\phi(t) : [0,1) \to \mathbb{C}^m\backslash Y$ has N continuous non-singular solution paths $z(t)$.*

3. *When $t \to 1^-$, the limit of $z_k(t)$, $k = 1, ..., N$ includes all the non-singular roots of $F(z;\phi(1))$.*

An important question is how to choose a homotopy path $\phi(t)$ which can avoid the exceptional set Y. The following lemma [52] gives an easy way to address this problem.

Lemma 1.3.2. *Fix a point q and a proper algebraic set Y in $\mathbb{C}^m$. For a generic point $p \in \mathbb{C}^m$, the one-real-dimensional open line segment $\phi(t) := (1-t)\,p + t\,q, t \in [0,1)$ is contained in $\mathbb{C}^m \backslash Y$.*

We now apply this lemma to Equation (1.3.4), where q represents the target system $f(z)$ and p represents the initial system $e^{i\theta}z^d - e^{i\theta}$ in the parameter space for a random $\theta \in [-\pi, \pi]$. The gamma trick introduces a type of randomization to the choice of p. During the construction of an initial system (even for positive dimensional system solving), we always use randomization techniques to avoid such "bad" situations arising in the numerical computation.

In fact, the "bad set" is not only a lower dimensional variety Y but the numerically difficult region around Y, which has nonzero measure. So in the numerical computation, we should replace "with probability 1" with "with high probability".

1.3.2 Polytope Structure

When we apply homotopy continuation methods to solve polynomial system $F(z) = \{f_1(z), ..., f_n(z)\}$ with degree of f_i equal to d_i, an initial system needs to be solved first. By Lemma 1.3.2 the initial system corresponds to a generic point in the parameter space. Of course we can use total degree to construct the parameter space (all the systems will have $d = \prod_i^n d_i$ nonsingular roots).

However the target system may not be a generic system, and the number of roots can be fewer than the generic case. A well-known example is the eigenvalue problem $A \cdot v = \lambda v, a \cdot v = 1$, where $A \in \mathbb{F}^{n\times n}, a \in \mathbb{F}^n$. This example only has n roots, but the Bezout number of this system is 2^n. This means if we use total degree homotopy, when $t \to 1^-$, there will be many singular paths, which causes the paths tracking to be very inefficient.

If we can embed the target system into a special class of systems, which has much fewer roots, then by Theorem 1.3.1, we can still find all the roots by homotopy continuation. There are many ways to efficiently estimate the number of roots of the target system and construct an initial system [25, 31, 26, 58, 16].

Newton Polytopes and Mixed Volume

Here we introduce Newton Polytope techniques which often give a sharp estimate of the number of roots of a given polynomial system.

Let $\mathbb{C}^* := \mathbb{C}\backslash 0$, denote the nonzero complex numbers. A Laurent polynomial in

the variables $x = (x_1, ..., x_n)$ is defined in multi-index notation as

$$f(x) := \sum_{\alpha \in S} c_\alpha x^\alpha \tag{1.3.6}$$

where $S \subset \mathbb{Z}^n$ and each $c_\alpha \in \mathbb{C}^*$. Here S corresponds to the monomial set of f, which is called the *"support"* of f. Embed S into $\mathbb{R}^n$. The convex hull $Q := \text{conv}(S)$ of S is said to be the *"Newton Polytope"* of f, and is also denoted by $Q = \text{conv}(f)$. Note that Laurent monomial x^α allows negative degrees.

Suppose we have two Laurent polynomials f_1 and f_2. A geometric addition operation on polytopes $Q_1 = \text{conv}(f_1)$ and $Q_2 = \text{conv}(f_2)$, called the "Minkowski sum" is of interest to us:

$$Q_1 + Q_2 := \{q_1 + q_2 : q_1 \in Q_1, q_2 \in Q_2\} \tag{1.3.7}$$

where $q_1 + q_2$ is addition in the vector space $\mathbb{R}^n$. It is interesting that the sum is still a convex polytope and it equals $\text{conv}(f_1 \cdot f_2)$.

Suppose the vertices of an n-dimensional polytope Q are $v_0, v_1, ..., v_n$ (if Q has more vertices we can easily decompose it as a union of polytopes and each of them has $n + 1$ vertices). The volume of this polytope is

$$\text{Vol}_n(Q) = \frac{1}{n!} |\det[v_1 - v_0, ..., v_n - v_0]| \tag{1.3.8}$$

Proposition 4.9 in Chapter 7 of reference [7] shows that $\text{Vol}(\lambda_1 Q_1 + \cdots + \lambda_n Q_n)$ is a homogenous polynomial of degree n in λ_i. A certain coefficient of this polynomial has a special meaning.

Definition 1.3.2. *[Mixed Volume] The mixed volume of convex polytopes $Q_1, ..., Q_n$ is defined as the coefficient of the term $\lambda_1 \cdots \lambda_n$ in the homogenous polynomial $\text{Vol}(\lambda_1 Q_1 + \cdots + \lambda_n Q_n)$, and is denoted by $M_n(Q_1, ..., Q_n)$.*

Mixed volume is a very important invariant of a polynomial system. Firstly, it is a symmetric function of $Q_1, ..., Q_n$. And, it is an invariant under a shift of polytopes (e.g. $M_n(Q_1, ..., Q_n) = M_n(Q_1 + \mathbf{v}, ..., Q_n)$, for any $\mathbf{v} \in \mathbb{R}^n$). See [57] for other properties of mixed volumes.

One of the most important applications of Mixed Volume is the following theorem.

Theorem 1.3.3. *[Bernstein Theorem] Let $F = \{f_1, ..., f_n\}$ be a system of polynomials. Then the number of roots of $F = 0$ (counting multiplicities) in $(\mathbb{C}^*)^n$ is bounded by the mixed volume $M_n(Q_1, ..., Q_n)$ where $Q_i = \text{conv}(f_i)$. Moreover if the coefficients of F are generic then the mixed volume gives the exact number of roots of $F = 0$ in $(\mathbb{C}^*)^n$.*

This bound is also called the "BKK bound" in recognition of the contributions of Bernstein (1975), Kushnirenko (1976) and Khovanski (1978).

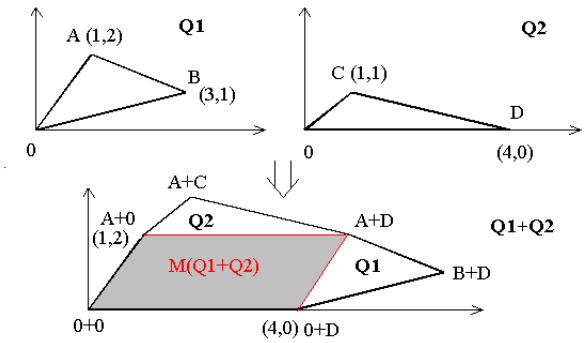

Figure 1.3.1: Compute Mixed Volume by Minkowski sum

This bound only involves the information on monomials appearing in the polynomial system F. Consequently it should be cheaper to compute the bound than to solve the system. Let us begin with a simple case to see how to compute the mixed volume and subsequently this bound.

We know $\mathrm{Vol}_n(\lambda_1 Q_1 + \lambda_2 Q_2)$ is a homogenous quadratic polynomial in λ_1, λ_2. Suppose it has the form: $p(\lambda_1, \lambda_2) = c_1\lambda_1^2 + c_{11}\lambda_1\lambda_2 + c_2\lambda_2^2$. So we have $p(1,1) - p(1,0) - p(0,1) = c_1 + c_{11} + c_2 - c_1 - c_2 = c_{11}$, which means

$$M_2(Q_1, Q_2) = \mathrm{Vol}_n(Q_1 + Q_2) - \mathrm{Vol}_n(Q_1) - \mathrm{Vol}_n(Q_2) \tag{1.3.9}$$

To show the main idea of this section, we choose a simple system given in Sommese et al [50] as our running example.

EXAMPLE 1.3.1. *Suppose* $f_1 = ax^3y + bxy^2 + 1, f_2 = cx^4 + dxy + 1$. *Let* Q_1, Q_2 *be the Newton Polytopes of* f_1, f_2 *respectively.*

The mixed volume is the area of the grey area of Figure 1.3.1, which is equal to the Minkowski sum of Q_1, Q_2 *minus the area of* Q_1 *and* Q_2.

We can generalize this formula to arbitrary n by induction [7]:

$$M_n(Q_1, ..., Q_n) = \sum_{i=1}^{n} (-1)^{n-i} \sum_{I \subset \{1,...,n\}, |I|=i} \mathrm{Vol}_n(\Sigma_{j \in I} Q_j) \tag{1.3.10}$$

Polyhedral Homotopies

Note that the formula (1.3.10) is not an efficient way to compute the mixed volume of a given polynomial system with many equations and many variables. First we introduce some concepts to show how to improve the efficiency of this computation. Here we restrict to polytopes spanned by integer vertices.

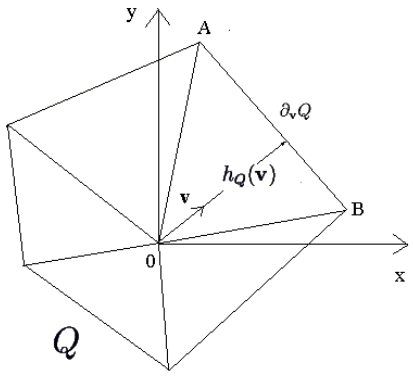

Figure 1.3.2: Recursive Computation of Volume

Definition 1.3.3. *Let $Q = \mathrm{conv}(S)$, spanned by $S \subset \mathbb{Z}^n$. The support function $h : \mathbb{R}^n \to \mathbb{R}$ on S is defined by*

$$h_S(\mathbf{v}) := \max_{\mathbf{a}\in S} \langle \mathbf{a}, \mathbf{v} \rangle, \tag{1.3.11}$$

where $\langle , \rangle$ is the Euclidian inner produce in $\mathbb{R}^n$.

Definition 1.3.4. *Let $Q = \mathrm{conv}(S), S \subset \mathbb{Z}^n$ and $\mathbf{v}$ be a vector in $\mathbb{R}^n$. Let $\partial_{\mathbf{v}} S = \{\mathbf{a} \in S : \langle \mathbf{a}, \mathbf{v} \rangle = h_S(\mathbf{v})\}$. Then the face of the polytope Q in the direction $\mathbf{v}$ is defined as the convex hull of $\partial_{\mathbf{v}} S$ and is denoted by $\partial_{\mathbf{v}} Q = \mathrm{conv}(\partial_{\mathbf{v}} S)$.*

Now we can compute the volume of a polytope in $\mathbb{Z}^n$ in a recursive way [58]:

$$\mathrm{Vol}_n(Q) = \frac{1}{n} \sum_{||\mathbf{v}||=1} h_Q(\mathbf{v}) \mathrm{Vol}_{n-1}(\partial_{\mathbf{v}} Q), \tag{1.3.12}$$

where $\mathbf{v}$ ranges over all unit vectors in $\mathbb{R}^n$.

We now illustrate the intuitive geometric idea behind this formula by Figure 1.3.2.

Note that there are only finitely many normalized outer normals $\mathbf{v}$ of Q for which $\partial_{\mathbf{v}} Q \neq 0$. For each facet of Q, there is a unique normalized outer normal $\mathbf{v}$. In Figure 1.3.2, we can consider $h_Q(\mathbf{v})$ and $\partial_{\mathbf{v}} Q$ as the height and the base of the triangle $\triangle ABO$. So the area of $\triangle ABO$ is equal to $1/2 \cdot h_Q(\mathbf{v}) \mathrm{Vol}_1(\partial_{\mathbf{v}} Q)$. It is easy to see that this can be generalized to the n-dimensional case.

Combining Formula (1.3.12) with Formula (1.3.10), we can obtain a recursive formula to compute the mixed volume of an n-tuple of polytopes:

$$M_n(Q_1, ..., Q_n) = \sum_{||\mathbf{v}||=1} h_{Q_1}(\mathbf{v}) M_{n-1}(\partial_{\mathbf{v}} Q_2, ..., \partial_{\mathbf{v}} Q_n). \tag{1.3.13}$$

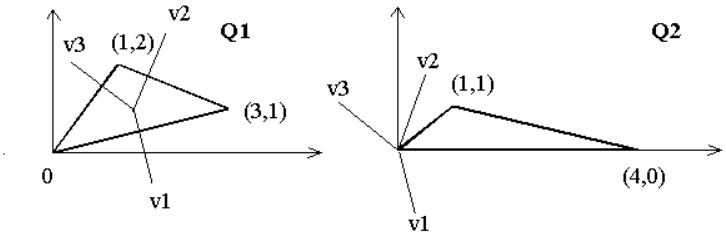

Figure 1.3.3: Recursive Computation of Mixed Volume

Here to illustrate the main idea we only look at a simple case (for more details see [7]). Specifically, the mixed volume of two polytopes is:

$$\begin{aligned}
M_2(P,Q) &= \mathrm{Vol}_2(P,Q) - \mathrm{Vol}_2(P) - \mathrm{Vol}_2(Q) \\
&= 1/2 \sum_{||\mathbf{v}||=1} h_{P+Q}(\mathbf{v})\mathrm{Vol}_1(\partial_{\mathbf{v}}(P+Q)) \\
&\quad -1/2 \sum_{||\mathbf{v}||=1} h_P(\mathbf{v})\mathrm{Vol}_1(\partial_{\mathbf{v}}P) - 1/2 \sum_{||\mathbf{v}||=1} h_Q(\mathbf{v})\mathrm{Vol}_1(\partial_{\mathbf{v}}Q) \\
&= 1/2 \sum_{||\mathbf{v}||=1} [(h_P(\mathbf{v}) + h_Q(\mathbf{v}))(\mathrm{Vol}_1(\partial_{\mathbf{v}}P) \\
&\quad +\mathrm{Vol}_1(\partial_{\mathbf{v}}Q)) - h_P(\mathbf{v})\mathrm{Vol}_1(\partial_{\mathbf{v}}P) - h_Q(\mathbf{v})\mathrm{Vol}_1(\partial_{\mathbf{v}}Q)] \\
&= 1/2 \sum_{||\mathbf{v}||=1} (h_P(\mathbf{v})\mathrm{Vol}_1(\partial_{\mathbf{v}}Q) + h_Q(\mathbf{v})\mathrm{Vol}_1(\partial_{\mathbf{v}}P)) \\
&= \sum_{||\mathbf{v}||=1} h_P(\mathbf{v}) M_1(\partial_{\mathbf{v}}Q).
\end{aligned}$$

EXAMPLE 1.3.2. *We apply Formula (1.3.13) to the previous Example 1.3.1. There are only* 3 *normalized outer normals* $\mathbf{v}$ *of* Q_1 *for which* $\mathrm{Vol}_1(\partial_{\mathbf{v}}Q_1) \neq 0$ *(as shown in Figure 1.3.3). So*

$$\begin{aligned}
M_2(Q_1,Q_2) &= \sum_{v=v1,v2,v3} h_{Q_2}(\mathbf{v}) M_1(\partial_{\mathbf{v}}Q_1) \\
&= \frac{4}{\sqrt{10}} \cdot \sqrt{10} + \frac{4}{\sqrt{5}} \cdot \sqrt{5} + 0 \cdot \sqrt{5} \\
&= 4 + 4 = 8.
\end{aligned}$$

which is consistent with the result obtained using Formula (1.3.9).

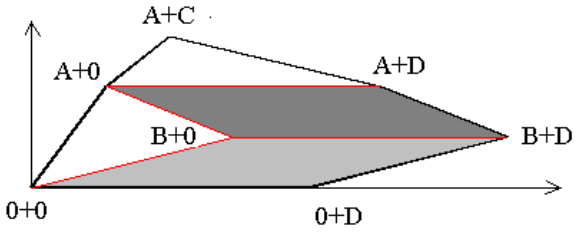

Figure 1.3.4: A mixed subdivision of $Q_1 + Q_2$

In practice, computing the mixed volume of many polytopes using Formula (1.3.13) can be complicated and inefficient. A better way, due to Huber and Sturmfels is given in [16] by using a mixed subdivision of the Minkowski sum of the polytopes.

We only show the main idea using Example 1.3.1. In Figure 1.3.1, the Minkowski sum of Q_1, Q_2 consists of three parts: $\{(A+0, A+C, A+D), (A+D, A+B, A+0), (0+0, A+0, A+D, 0+D)\}$. Only the grey area is a "mixed cell" which is a parallelogram spanned by an edge of Q_1 and an edge of Q_2. Huber and Sturmfels proved that the mixed volume is equal to the sum of volumes of such "mixed cells". T. Y. Li and his team have developed an efficient implementation of this approach in their software [10].

Note that such a mixed subdivision is not unique. For example, Figure 1.3.4 gives another mixed subdivision of $Q_1 + Q_2$. The upper "mixed cell" is the parallelogram spanned by $\{AB, OD\}$ and the lower "mixed cell" the parallelogram spanned by $\{BO, OD\}$. The sum of the areas is still 8.

We next consider how to "break" the Minkowski sum of $Q_1 + \cdots + Q_n$ into pieces to obtain a mixed subdivision. The key idea is to lift the polytopes Q_i to $\hat{Q}_i$ in a higher dimensional space and look at the facets of the lower hull of $\hat{Q}_1 + \cdots + \hat{Q}_n$. Since the union of the projections of these facets is the Minkowski sum of the original polytopes, the lifting may induce a subdivision of $Q_1 + \cdots + Q_n$. Huber and Sturmfels showed that we can always construct such a mixed subdivision by a sufficiently random (integer) lifting. Furthermore, this lifting can induce a polyhedral homotopy.

Rather than delve deeper into such techniques and theories, we simply illustrate the idea using the running Example 1.3.1.

EXAMPLE 1.3.3. *First we choose an integer lifting:* $\hat{Q}_1 = \{(3,1,1), (1,2,0), (0,0,0)\}$ *and* $\hat{Q}_2 = \{(4,0,0), (1,1,1), (0,0,0)\}$. *The lower hull of* $\hat{Q}_1 + \hat{Q}_2$ *is shown in Figure 1.3.5, which induces a mixed subdivision of* $Q_1 + Q_2$.

We transform the lifted polytopes to polynomials (using t *for the extra coordinate):*

$$H(x, y; t) = \begin{cases} a\, x^3yt + b\, xy^2 + 1 = 0 \\ c\, x^4 + d\, xyt + 1 = 0 \end{cases} \tag{1.3.14}$$

When $t = 1$, *we have* $H(x, y; 1) = F(x, y)$. *The roots of* $H(x, y; t) = 0$ *are alge-*

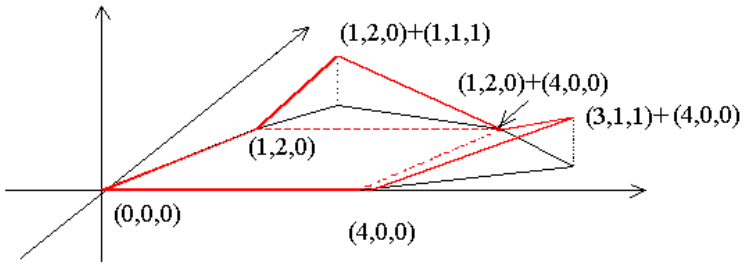

Figure 1.3.5: Lower Hull of $\hat{Q}_1 + \hat{Q}_2$

braic functions of the parameter t (and may have many branches). The branches of solutions given by Puiseux series have the form

$$(x(t), y(t)) = (x_0 t^{\gamma_1}, y_0 t^{\gamma_2}) + \textit{higher order terms}, \tag{1.3.15}$$

where $(\gamma_1, \gamma_2, 1)$ is an inner normal of the mixed cell of the lower hull of $\hat{Q}_1 + \hat{Q}_2$.

There is only one mixed cell and its inner normal is $(0, 0, 1)$. So by substituting $x = x_0, y = y_0, t = 0$ into $H(x, y; t)$ we obtain the initial system

$$\begin{cases} b\, x_0 y_0^2 + 1 = 0 \\ c\, x_0^4 + 1 = 0 \end{cases} \tag{1.3.16}$$

Obviously, if the coefficients are chosen randomly, then there are 8 roots of the initial system. This number is exactly equal to the volume of this mixed cell.

Remark 1.3.2. *To determine the values of γ_1, γ_2 in Equation (1.3.15), we can consider the lowest degree of each polynomial of Equation (1.3.14) with respect to t. If an equation has solutions when $t \to 0^+$, then its lowest degree must be greater than or equal to zero (constant numbers have degree 0). So we have*

$$\begin{cases} \min(0 + 3\gamma_1 + \gamma_2 + 1, 0 + \gamma_1 + 2\gamma_2, 0) \geq 0 \\ \min(0 + 4\gamma_1, 0 + \gamma_1 + \gamma_2 + 1, 0) \geq 0 \end{cases} \tag{1.3.17}$$

which can be simplified to

$$\begin{cases} \min(3\gamma_1 + \gamma_2 + 1, \gamma_1 + 2\gamma_2) = 0 \\ \min(4\gamma_1, \gamma_1 + \gamma_2 + 1) = 0 \end{cases} \tag{1.3.18}$$

The zero set of each equation is a piecewise line and we can check the intersection is $(0, 0)$. The theoretical study of such objects is called Tropical Algebraic Geometry, which is a relatively new area started in the late nineties [19].

1.3.3 Positive Dimensional Systems

When we consider positive dimensional systems, a significant obstacle to algorithms is giving a finite description of the infinite sets of points that are their positive dimensional components. One way is to use parametric representations (e.g. $(x, y) = (a, b)t + (x^0, y^0), t \in \mathbb{F}$ for a straight line). Another way to describe positive dimensional varieties is by using algebraic approaches, such as Gröbner Bases or Triangular Sets.

Numerical algebraic geometry provides us with a more geometric approach. It uses a certain nice finite subset of points of a component, called a *witness set*, to represent the whole component.

Linear Slicing and Witness Sets

Given a polynomial system F, we first suppose the algebraic variety $V(F)$ consists only of pure k-dimensional components. From Harris [15], we know that intersecting the components by a generic hyperplane (i.e. by appending a linear equation to F) will always drop the dimension of the components by 1. So by appending k generic linear equations $L^{[k]}$ to F, the dimension of the new system $G = \{F, L^{[k]}\}$ will be 0. Using homotopy continuation methods, we can compute all the isolated roots $A = V(G)$, which are generic points on $V(F)$ restricted to $L^{[k]}$. We put these three ingredients together and call it a *Witness Set* of $V(F)$, denoted by $W = (A, F, L^{[k]})$.

For a mixed dimensional algebraic set, we adopt the following recursive definition.

Definition 1.3.5. *Let $Z \subset \mathbb{C}^n$ be an affine algebraic set. Then a witness set for Z is a collection of witness sets of Z_k, $k = 0, ..., \dim(Z)$, where Z_k denotes the pure k-dimensional components of Z.*

We summarize the good properties of linear slicing in the following Slicing Theorem [52].

Theorem 1.3.4. *Let $X \subset \mathbb{C}^n$ be a pure k-dimensional affine algebraic set. There is a Zariski open set $U \subset \mathbb{P}^n$ such that for any $a \in U$ and $L(x; a)$:*
(1) if $k = 0$, then $V(L) \cap X = \emptyset$;
(2) if $k > 0$, then $V(L) \cap X$ is $k-1$-dimensional and $deg(V(L) \cap X) = deg(X)$;
(3) if $k > 1$ and X is irreducible, then $V(L) \cap X$ is irreducible.

Remark 1.3.3. *Witness sets are equivalent to the symbolic method of lifting fibers in a geometric resolution [24]. This idea of cutting with hyperplanes to determine the dimensions of solution components appeared in Guisti and Heintz [14].*

The approach of describing positive dimensional systems by using witness sets has many computational advantages.

1. It is cheaper than computing defining equations of varieties (a set of generators of a radical ideal).

Figure 1.3.6: Equi-Dimensional Decomposition of $V(F)$

2. It is numerically stable, consuming much less memory (see our comparison results in Chapter 2), and is suitable for parallel computation.

3. The witness sets can be used to construct an approximation of defining equations [30].

A witness set $W = (A, F, L^{[k]})$ encodes geometric invariants and gives us a more direct and explicit description of the components of an algebraic set. For example, the number of equations of $L^{[k]}$ is equal to the *dimension* of the component and the number of generic points in A is the *degree* of the components.

Also the witness points can be used as an inclusion test for varieties and a radical ideal membership test for ideals. Given a radical ideal I and a polynomial f, we know that $f \in I \Leftrightarrow V(f) \supset V(I)$ by the ideal-variety correspondence. So the radical ideal membership test can be reduced to a special case of the inclusion test of varieties.

Proposition 1.3.1. *Let f be a polynomial in $\mathbb{C}[x_1, ..., x_n]$ and W be the witness set of a variety $\hat{V} \subset \mathbb{C}^n$. If $f(p) = 0$, for all the witness points of W, then $\hat{V} \subset V(f)$ equivalently $f \in I(\hat{V})$.*

This test is very efficient and in particular we will use it to remove the redundant equations of a (differential) polynomial system [39, 62]. It leads immediately to an equality test for two varieties.

Decomposition of Components

A main task of numerical algebraic geometry is the so-called numerical equi-dimensional decomposition and irreducible decomposition of algebraic sets.

We start from a simple but visualizable example.

EXAMPLE 1.3.4. *Consider a polynomial system $F = [f_1, f_2]$ with variables x, y, z given by*

$$F = \begin{bmatrix} (z - y^2)(x^2 + y^2 + z^2 - 1) \\ x(z - x^3)(x^2 + y^2 + z^2 - 1) \end{bmatrix} = 0 \tag{1.3.19}$$

In Figure 1.3.6, visually, the variety of F consists of a sphere and some algebraic curves.

Figure 1.3.7: Irreducible Decomposition of $V(F)$

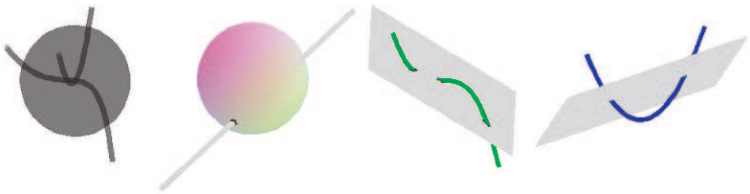

Figure 1.3.8: Irreducible Decomposition of $V(F)$

If we do the numerical irreducible decomposition, we can see in Figure 1.3.7 that there is one 2-dimensional component (a sphere) and two 1-dimensional components (two curves).

Now we use witness sets to represent $V(F)$. For the 2-dimensional component, cutting by two random hyperplanes, (or equivalently by a random line), we obtain 2 witness points. Using one random hyperplane to cut the 1-dimensional components, we obtain 2 witness points for one curve and 3 witness points for the other, as shown in Figure 1.3.8. These results yield the dimension and degree of each component.

By Definition 1.3.5, to compute the equi-dimensional decomposition of an algebraic set is equivalent to finding its witness set. Since we use the linear slicing method to find the witness points, it is easier to "hit" the points on higher dimensional components. The Slicing Theorem 1.3.4 shows that the intersection of a generic co-dimension k linear space and a component with dimension lower than k is always empty. So we can start by searching for top dimensional components by slicing with $n-1$ hyperplanes then peeling off one hyperplane at a time and descending from $n-1$ dimensional to 0 dimensional components. This is called the "Cascade Algorithm" [47].

When we append $n-1$ random linear equations to the original system, we often obtain an over-determined system (unless the original system has only one equation). We still use the system in Example 1.3.4. After appending two random linear equations, we have a new over-determined system $G_2 = [f_1, f_2, L_1, L_2]$.

In order to use homotopy continuation methods, we have to transform G_2 into a square system. A naive way to do this is to choose three equations to solve first, then substitute the solutions into the fourth one and finally check the roots of the

resulting system. Unfortunately, sometimes this approach may fail. For example, the solutions of $[x(x+y-1)=0, y(x+y-1)=0, xy=0]$ are $\{(0,0),(0,1),(1,0)\}$, but any two equations of the system intersect at a line rather than a finite set.

One good way to proceed is to embed G_2 into a 4-dimensional space by adding one slack variable s and choosing the coefficients $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ randomly:

$$G_2' = [f_1 + \alpha_1 s, f_2 + \alpha_2 s, L_1 + \alpha_3 s, L_2 + \alpha_4 s] \tag{1.3.20}$$

So (x^0, y^0, z^0) is a solution of G_2 if and only if $(x^0, y^0, z^0, 0)$ is a solution of G_2'. Consequently, we can apply homotopy continuation to the square system G_2' to find all the isolated solutions of G_2' and check if the last coordinate is zero or not.

Another natural procedure to obtain a square systems is by taking random linear combinations of the original equations, denoted by $M^{n\times\ell}\cdot F^{\ell\times 1}$, where ℓ is the number of original equations. If we choose the matrix M randomly, then any positive dimensional component of $V(F)$ is irreducible if and only if it is an irreducible component of $V(M\cdot F)$ (see [52]).

By the methods discussed above we can show G_2 has finitely many isolated solutions, the top dimensional component of $V(F)$ is 2 dimensional and these solutions are witness points of the 2-dimensional component.

Now we remove one hyperplane to compute the witness sets of 1 dimensional components of $V(F)$. However L_1 intersects with the curves of $V(F)$ and also inescapably meets with the 2-dimensional component of F (the surface). So the solutions of $G_1 = [f_1, f_2, L_1]$ will contain some generic points on the curves and some generic points on the surface. So they are not a witness set of 1-dimensional components, but a superset of them, which is called a *Witness Superset*.

To obtain a witness set from a witness superset it is necessary to remove the excess (so-called *junk*) points on the higher dimensional components. Note that the witness superset of the top dimensional components is a witness set, and the nonsingular points in the witness supersets of lower dimensional components are true witness points. For the singular points, there are two possibilities: they could be witness points with multiplicities greater than 1 or they belong to higher dimensional components. So any method for computing the local dimension of a variety at a point can be used to remove the junk points.

Another way to remove junk points involves a technique called "homotopy membership testing", which is also important in the numerical irreducible decomposition. Suppose we have the witness set $(A, F, L^{[k]})$ of the pure k-dimensional components of $V(F)$ and a point $p \in V(F)$. We want to know if p is contained in these components. The key idea is to choose a generic linear space L_2 passing through p with codimension k. We then construct a homotopy $H(x,t) = [F, (1-t)L^{[k]} + tL_2]$ and track the paths starting at $t=0$ from A and the ending at $t=1$ with the point B (t varies from 0 to 1). If $p \in B$, then p is a point of the pure k-dimensional components of $V(F)$, otherwise it is not.

In conclusion, by the techniques discussed above, we can compute a witness set of an algebraic set, which gives a representation of the equi-dimensional decomposition of $V(F)$.

A numerical irreducible decomposition method was first given in [48], and is not used in this Thesis. Roughly speaking, it is the further decomposition of a pure dimensional solution set into irreducible components by using monodromy loops, certified by linear traces. For the details of the algorithm, we refer the readers to the book by Sommese and Wampler [52].

Diagonal Homotopy

In Chapter 2 and Chapter 3, we will see that the algebraic equations appear one by one after the iteration of differential eliminations. The new equations will be appended to the old system to obtain a larger system, which geometrically corresponds to the intersection of algebraic sets.

Reuse of the witness set of the old system is very important for the efficiency of the whole algorithm. A recent technique called "diagonal homotopy" in numerical algebraic geometry fits this purpose perfectly.

Here we give a brief introduction to the diagonal intersection algorithm. Suppose we intersect two solution components of equations $F(x) = 0$ and $G(x) = 0$ in X space, where $x = (x_1, ..., x_n) \in X$. Suppose we know the witness sets (A, F, L_F) and (B, G, L_G) already. First we embed the two systems into $X \times X$ space: $\{F(x) = 0, G(y) = 0\}$ and $(x, y) \in X \times X$, where $y = (y_1, ..., y_n)$.

Now consider the homotopy

$$H(x, y, t) = \begin{bmatrix} F(x) \\ G(y) \\ L_F(x) \\ L_G(y) \end{bmatrix} (1 - t) + \begin{bmatrix} F(x) \\ G(y) \\ x - y \\ L(x) \end{bmatrix} t = 0 \tag{1.3.21}$$

We know the degree of the intersection is bounded by the product of degrees. So, at $t = 0$, we start at the points belonging to the product of the two witness sets $A \times B$. When $t = 1$, $H(x, y, 1)$ is at the diagonal $x - y = 0$, which is equivalent to $\{F(x) = 0, G(x) = 0, L(x) = 0\}$. When we change the codimension of the generic linear space L using the Cascade Algorithm, we can compute a witness set of all components (after removing the junk points). For a detailed description of diagonal homotopies see [49].

1.4 Organization and Comments

This Thesis is presented in an integrated-article format. Chapters from 2 to 6 treat discrete but related problems and are the author's publications [39, 62, 63, 30, 66] respectively. The first three papers focus on using a new area called *Numerical Jet*

Geometry to study the geometric structure of differential equations in Jet Space by using numerical methods. The last two papers focus on approximate computation in polynomial algebra.

1.4.1 Comments on [39] (Chapter 2)

This paper is a continuation of symbolic-numeric methods for differential systems begun in [60, 38]. The main contribution is to process the leading linear and nonlinear parts by using symbolic and numeric methods separately.

The higher efficiency is due to executing (radical) ideal membership testing by a numeric method which only computes a part of information of the nonlinear system rather than the complete ideal theoretic information given by the symbolic methods (e.g. Gröbner Bases and Triangular Decomposition). In particular radical ideal membership testing is reduced to substituting witness points that are efficiently computed by homotopy continuation.

Another key factor is to exploit structured information such as mixed volume to dramatically reduce the number of the paths followed by homotopy continuation. The new technique Diagonal Homotopy introduced in [49] plays the important role of reusing the existing information and provides a powerful tool to analyze the large systems.

1.4.2 Comments on [62] (Chapter 3)

The Hybrid method given in Paper [39] only applies to exact input systems. If the input is approximate then this method can be numerically unstable because the leading linear part is processed by some algorithms, closely related to Gaussian Elimination, which compute a solved form subject to a given ordering. The ordering can force pivoting on small quantities and hence induce instability.

Paper [62] aims to replace the Gaussian type Differential Elimination algorithms by stable numerical methods. The philosophy is to use geometric approaches to study PDE, e.g. the Cartan-Kuranishi method combined with numerical algebraic geometry. It originates from the idea in [38]. But the method given in that paper causes large nonlinear systems with extremely large Bezout numbers. One of the key contributions of Paper [62] is to exploit the linearity which always appears after prolongation.

The second contribution is to replace the membership test in Paper [39] by a rank test. The advantage is to detect the existence of the new constraints before we compute them and this provides a much cheaper criterion for termination of the method.

The third contribution is the construction of the projected constraints by computing the null-space of polynomial matrices.

1.4.3 Comments on [63] (Chapter 4)

Pryce gave an efficient method in [36] for square Differential Algebraic Equations (DAE) which are essentially ODE with algebraic constraints. The most interesting aspect of his approach is that it only involves differentiation and no elimination. Paper [63] generalizes this idea to square systems of PDE and gives an efficient prolongation method which only requires differentiation with respect to one of the independent variables. In this framework, Pryce's method is a special case.

Genericity statements in this paper show that our fast prolongation method has a high probability of success and it can be applied to a wide class of PDE.

There is an interesting analogue between our fast prolongation method for differential systems and mixed volume techniques for polynomial systems. First they are both concerned with structural information for square systems (ignoring the coefficients). Secondly, generic choices of the coefficients can guarantee the success for both methods. Finally, both methods use combinatorial and linear programming techniques to compute the results. The integer linear programming problem given in this paper is dual to an assignment problem. Eric Schost pointed out such problems can be solved in polynomial time by Hungarian Method (Harold W. Kuhn, 1955). We also show in our paper such problems can be solved very efficiently in practice.

1.4.4 Comments on [30] (Chapter 5)

Triangular decomposition techniques [64, 23, 20, 68, 59, 1, 8] give desirable algebraic representations for varieties of exact polynomial systems. Our study on differential equations always focuses on polynomially nonlinear systems, so approximate computation in polynomial algebra is a subfield of Numerical Jet Geometry.

In this paper we give the first method to construct approximate triangular decomposition from geometric objects: isolated points. The advantage is that geometric objects are more stable than their algebraic representations. The symbolic computation of triangular decomposition is a long sequence of manipulations of algebraic equations which often cause very large accumulation errors in the coefficients. On the other hand, the construction from geometric objects requires only three steps, for which we can easily provide an estimate on cumulative errors by using statistical tools. The isolated roots are computed by using homotopy continuation methods. The condition number at each root, a key factor of our error analysis, delivers the information on the"quality" of the approximation.

In this paper we use monomial basis for the interpolation. Dr. Corless suggested us to consider the other basis which may lead to a more stable result. We gave the forward error analysis in this paper, but backward error is also very important for a numerical algorithm. As Corless suggested, we can consider an approximate triangular decomposition as an exact triangular decomposition of some variety which is close to the given one. Such backward error analysis deserves a careful study in the future work.

1.4.5 Comments on [66] (Chapter 6)

This unpublished paper gives the details about the theory and algorithms for polynomial matrices. This theory plays an important role in Paper [62].

One of the motivations is to exploit the linearity of a prolonged differential system. The key information is contained in the Symbol matrix, which is a polynomial matrix, if we only consider polynomially nonlinear PDE.

Another goal is to explore the relation between the syzygy module and the nullspace of a polynomial matrix. The paper also lays some foundations for interpreting the approximate computation as a "nearby problem".

When we reduce polynomial algebra to linear algebra, the matrices appearing in the computation always have structure which plays an important role. As we can see in Example 6.2.1, Dr. Corless pointed out the significant difference between structured and unstructured matrices.

Bibliography

[1] P. Aubry, D. Lazard, and M. Moreno Maza. On the theories of triangular sets. J. Symb. Comp., 28(1,2):45–124, 1999.

[2] L. Blum, F. Cucker, M. Shub, and S. Smale. Complexity and Real Computation. Springer–Verlag, 1998.

[3] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Representation for the radical of a finitely generated differential ideal. Proc. ISSAC'95, 158–166, 1995.

[4] G. Carra-Ferro. Gröbner Bases and Differential Algebra. Lecture Notes in Comp. Sci. 356 128-140, 1987.

[5] Y. Chen and X.-S. Gao. Involutive Bases of Algebraic Partial Differential Equation Systems. Science in China (A), 33(2), page 97–113, 2003.

[6] S. S. Chern, W. H. Chen and K. S. Lam. Lectures on Differential Geometry. World Scientific Press, 1999.

[7] D. Cox, J. Little, and D.O'Shea. Using Algebraic Geometry. Springer-Verlag, New York, 1998. Graduate Texts in Mathematics.

[8] X. Dahan, M. Moreno Maza, É. Schost, W. Wu, and Y. Xie. Lifting techniques for triangular decompositions. In ISSAC'05, pages 108-115, ACM Press 2005.

[9] P. Shi and J. McPhee. DynaFlex User's Guide. Available at http://real.uwaterloo.ca/∼ dynaflex/manual.pdf

[10] T. Gao, T. Y. Li and M. Wu. Algorithm 846: MixedVol: a software package for mixed-volume computation. ACM Transactions on Mathematical Software, Vol 31(4), Pages 555-560, 2005

[11] H. Goldschmidt. Existence theorems for analytic linear partial differential equations. Ann. Math. Vol 86, pages 246-270, 1967.

[12] H. Goldschmidt. Integrability criteria for systems of non linear partial differential equations. J. of Differential Geometry, Vol 1, pages 269-307, 1967.

[13] V. W. Guillemin and S. Sternberg. An algebraic model of transitive diferential geometry. Bull. Amer. Math. Sac., 70 (1964), 16-47.

[14] M. Guisti and J. Heintz. La détermination de la dimension et des points isolées d'une variété algébrique peuvent s'effectuer en temps polynomial. Computational Algebraic Geometry and Commutative Algebra, Cortona 1991, vol. XXXIV of Symposia Mathematica, pages 216–256. Camb. Univ. Press, 1993.

[15] J. Harris. Algebraic Geometry, a First Course. Springer-Verlag, 1992.

[16] B. Huber and B. Sturmfels. A polyhedral method for solving sparse polynomial systems. Math. Comp., Vol 64(212), pages 1541-1555, 1995.

[17] E. Hubert. Detecting degenerate cases in non-linear differential equations of first order. Theoretical Computer Science 187(1-2): 7–25, 1997.

[18] S. Ilie, R. M. Corless and G. Reid. Numerical solutions of index-1 differential algebraic equations can be computed in polynomial time. Numerical Algorithms, Vol 42(2), pages 161-171, 2006.

[19] I. Itenberg, G. Mikhalkin and E. I. Shustin. Tropical Algebraic Geometry. Birkhauser Basel, 1st edition, 2007.

[20] M. Kalkbrenner. A generalized euclidean algorithm for computing triangular representations of algebraic varieties. J. Symb. Comp., 15:143–167, 1993.

[21] E.R. Kolchin. Differential Algebra and Algebraic Groups. Academic Press, 1973.

[22] M. Kuranishi. On E. Cartan's Prolongation Theorem of Exterior Differential Systems. Amer. J. Math, Vol 79, pages 1-47, 1957.

[23] D. Lazard. Solving zero-dimensional algebraic systems. J. Symb. Comp., 13:117–133, 1992.

[24] G. Lecerf. Computing the equidimensional decomposition of an algebraic closed set by means of lifting fibers. J. Complexity 19(4):564–596, 2003.

[25] T. Y. Li and T. Sauer and J. A. Yorke. The random product homotopy and deficient polynomial systems. Numer. Math. Vol 51(5), pages 481-500, 1987.

[26] T. Y. Li and T. Sauer and J. A. Yorke. The cheater's homotopy: an efficient procedure for solving systems of polynomial equations. SIAM J. Numer. Anal., Vol 26(5), pages 1241-1251, 1989.

[27] B. Malgrange. Systèmes différentiels involutifs. Panoramas et Synthèses 19, 2005.

[28] E. Mansfield. Differential Gröbner Bases. Ph.D. thesis, Univ. of Sydney, 1991.

[29] E. Mansfield. A simple criterion for involutivity. J. London Math. Soc., Vol 54, pages 323-345, 1996.

[30] M. Moreno Maza, G. Reid, R. Scott and W. Wu. On Approximate Triangular Decompositions in Dimension Zero. J. Symbolic Comp., 42(7), 693-716, 2007.

[31] A. Morgan and A. Sommese. A homotopy for solving general polynomial systems that respects m-homogeneous structures. Appl. Math. Comput., Vol 24(2), pages 101-113, 1987.

[32] F. Ollivier. Standard bases of differential ideals, Lecture Notes in Comp. Sci., Vol 508, pages 304-321, 1991.

[33] Peter J. Olver. Equivalence, Invariants and Symmetry (London Mathematical Society Lecture Notes). Cambridge University Press, 1995.

[34] J.F. Pommaret. Systems of Partial Differential Equations and Lie Pseudogroups. Gordon and Breach Science Publishers, Inc. 1978.

[35] J.F. Pommaret. Partial Differential Control Theory. Series: Mathematics and Its Applications, Vol. 530, Springer, 2001.

[36] J.D. Pryce. A Simple Structure Analysis Method for DAEs. BIT, vol 41, No. 2, pp. 364-394, 2001.

[37] D. G. Quillen. Formal properties of over-determined systems of linear partial differential equations. PhD Thesis, Harvard University, 1964.

[38] G. Reid, C. Smith, and J. Verschelde. Geometric completion of differential systems using numeric-symbolic continuation. SIGSAM Bulletin 36(2):1–17, 2002.

[39] G. Reid, J. Verschelde, A.D. Wittkopf and W. Wu. Symbolic-Numeric Completion of Differential Systems by Homotopy Continuation. Proc. ISSAC 2005. ACM Press. 269–276, 2005.

[40] G.J. Reid, A.D. Wittkopf and A. Boulton. Reduction of systems of nonlinear partial differential equations to simplified involutive forms. Eur. J. of Appl. Math. 7: 604–635, 1996.

[41] G.J. Reid, P. Lin, and A.D. Wittkopf. Differential elimination-completion algorithms for DAE and PDAE. Studies in Applied Math. 106(1): 1–45, 2001.

[42] J.F. Ritt. Differential Algebra, Amer. Math. Soc. Colloq. Publns. 13 (A.M.S., New York), 1950.

[43] C.J. Rust, G.J. Reid, and A.D. Wittkopf. Existence and uniqueness theorems for formal power series solutions of analytic differential systems. Proc. ISSAC 99. ACM Press. 105-112, 1999.

[44] C.J. Rust, *Rankings of derivatives for elimination algorithms and formal solvability of analytic partial differential equations*, Ph.D. Thesis, University of Chicago, 1998.

[45] W.M. Seiler. Involution - The formal theory of differential equations and its applications in computer algebra and numerical analysis. Habilitation Thesis, Univ. of Mannheim, 2002.

[46] I. M. Singer and S. Sternberg. On the infinite groups of Lie and Cartan I. J. d'Analyse Mathematiques, vol XV, pp 1-114, 1965.

[47] A.J. Sommese and J. Verschelde. Numerical homotopies to compute generic points on positive dimensional algebraic sets. Journal of Complexity 16(3):572-602, 2000.

[48] A.J. Sommese, J. Verschelde, and C.W. Wampler. Numerical decomposition of the solution sets of polynomial systems into irreducible components. SIAM J. Numer. Anal. 38(6):2022–2046, 2001.

[49] A.J. Sommese, J. Verschelde, and C.W. Wampler. Homotopies for intersecting solution components of polynomial systems. SIAM J. Numer. Anal. 42(4):1552–1571, 2004.

[50] A.J. Sommese, J. Verschelde, and C.W. Wampler. Introduction to Numerical Algebraic Geometry. Algorithms and Computation in Mathematics, Vol 14, pages 339-392. Springer-Verlag 2005.

[51] A.J. Sommese and C.W. Wampler. Numerical Algebraic Geometry. In The Mathematics of Numerical Analysis, Volume 32 of Lectures in Applied Mathematics, 749–763, 1996. Proceedings of the AMS-SIAM Summer Seminar in Applied Mathematics, Utah, 1995.

[52] A.J. Sommese and C.W. Wampler. The Numerical solution of systems of polynomials arising in engineering and science. World Scientific Press, Singapore, 2005.

[53] D. C. Spencer. Overdetermined systems of linear partial differential equations. Bull. Amer. Math. Soc. Vol 75, pages 179-239, 1969.

[54] W. J. Sweeney. The δ-Poincare estimate. Pacific J. Math., Vol 20(3), pages 559-570, 1967.

[55] J. Tuomela and T. Arponen. On the numerical solution of involutive ordinary differential systems. IMA J. Numer. Anal. 20: 561–599, 2000.

[56] J. Verschelde. Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. ACM Transactions on Mathematical Software 25(2): 251–276, 1999. Software available at `http://www.math.uic.edu/~jan`.

[57] J. Verschelde. Homotopy Continuation Methods for Solving Polynomial Systems. PhD Thesis, Katholieke University, 1996.

[58] J. Verschelde, P. Verlinden and R. Cools. Homotopies exploiting Newton polytopes for solving sparse polynomial systems. SIAM J. Numer. Anal., Vol 31(3), pages 915-930, 1994.

[59] D. M. Wang. Elimination Methods. Springer, Wein, New York, 2000.

[60] A. Wittkopf and G.J. Reid. Fast differential elimination in C: The CDiffElim environment. Computer Physics Communications, 139: 192–217, 2001.

[61] A. Wittkopf. *Algorithms and Implementations for Differential Elimination.* Ph.D. Thesis, Simon Fraser University, 2004.

[62] Wenyuan Wu and Greg Reid. Application of Numerical Algebraic Geometry and Numerical Linear Algebra to PDE. Proc. of ISSAC'06, pages 345-352, ACM 2006.

[63] Wenyuan Wu and Greg Reid. Symbolic-numeric Computation of Implicit Riquier Bases for PDE. Proc. of ISSAC'07, pages 377-385, ACM 2007.

[64] W. T. Wu. A zero structure theorem for polynomial equations solving. MM Research Preprints, 1:2–12, 1987.

[65] W. T. Wu. On the foundations of algebraic differential geometry. *Mathematics-Mechanization Research Preprint* No. 3, pages 1–26, 1989.

[66] Wenyuan Wu. Computing the Rank and Null-space of Polynomial Matrices. Preprint, 2006.

[67] K. Yang. Exterior Differential Systems and Equivalent Problems. Kluwer Academic Publishers, 1992.

[68] L. Yang and J. Zhang. Searching dependency between algebraic equations: an algorithm applied to automated reasoning. In Johnson, J., McKee, S. and Vella, A., eds, Artificial Intelligence in Mathematics, pp.147-156. Oxford, Oxford University Press.

Chapter 2

Symbolic-Numeric Completion of Differential Systems by Homotopy Continuation

Two ideas are combined to construct a hybrid symbolic-numeric differential-elimination method for identifying and including missing constraints arising in differential systems. First we exploit the fact that a system once differentiated becomes linear in its highest derivatives. Then we apply diagonal homotopies to incrementally process new constraints, one at a time. The method is illustrated on several examples, combining symbolic differential elimination (using **rifsimp**) with numerical homotopy continuation (using **phc**).

2.1 Introduction

Over and under-determined systems of ODE and PDE arise in applications such as constrained multibody mechanics and control systems (e.g. differential-algebraic equations (DAE) arise in constrained Lagrangian mechanics [20]).

Much progress has been made in exact differential elimination methods, theory and algorithms for nonlinear systems of PDE. For example see Boulier et al. [3], Chen and Gao [5], Hubert [9], Mansfield [12], Seiler [20], Reid, Rust et al. [18], Wu [30]. Such methods enable the identification of all the hidden constraints for a system of PDE and the automatic statement of an existence and uniqueness theorem for its solutions. They give a geometrical view of its solution space [17, 20] and enable the determination of its symmetry properties. They enable the computation of initial data and associated formal power series solutions in the neighborhood of a point. Algorithmic membership tests (specifically in the radical of a differential ideal) can be given [3, 9]. They can ease the difficulty of numerical solution of DAE systems [26].

This paper is a sequel to [14] and [7] in which we develop theory and methods

for using numerical homotopy continuation methods in the differential elimination process to identify missing constraints for systems of differential equations. In [14] such methods were first introduced by combining the Cartan-Kuranishni approach with homotopy methods. During the application of that approach all equations are differentiated up to the current highest derivative order, resulting in potentially large numbers of PDE. These PDE are treated as polynomial equations in jet space, and their large number implies that the number of continuation paths that must be tracked can be impractically large.

In this paper we process the leading linear PDE using the **rifsimp** algorithm [29] and the leading nonlinear PDE using PHCpack [27], applying diagonal homotopies [23]. The correctness of our results can be certified if the constraints are free from multiplicities and the numerical representations are well conditioned.

This paper is organized as follows. In Section 2.2 we introduce some basic material for symbolic differential elimination, and in Section 2.3 we give a short overview on recent developments in homotopy methods. In Section 2.4 we present our method and briefly outline some optimizations in Section 2.5. Examples are given in Section 2.6 and concluding remarks in Section 4.9.

2.2 Symbolic Differential Elimination

Consider a polynomially nonlinear system of PDE $R = (R^1, \ldots, R^l) = 0$ with independent variables $x = (x_1, \ldots, x_n)$ and dependent variables $u = (u^1, \ldots, u^m)$ over $\mathbb{C}$ with coefficients from some computable extension of $\mathbb{Q}$. As in [3, 18, 20] solutions and derivatives are replaced by formal (jet) variables, allowing manipulation of equations without first assuming that solutions exist [13]. In particular, denoting the p-th order jet variables corresponding to derivatives as $\underset{p}{u}$, the jet variety of a qth order system in $J^q = \mathbb{C}^{n_q}$ is

$$V(R) := \{(x, u, \underset{1}{u}, \ldots, \underset{q}{u}) \in J^q : R(x, u, \underset{1}{u}, ..., \underset{q}{u}) = 0\}. \tag{2.2.1}$$

Here $n_q = n + m\binom{n+q}{q}$ is the number of independent variables, dependent variables and derivatives of order less than or equal to q. We restrict to the subset of the variables of J^q that actually appear in the given system.

EXAMPLE 2.2.1. *Throughout this article we use the following running example, first introduced in [16], see also [7]:*

$$\frac{\partial^2 u(x,y)}{\partial y^2} - \frac{\partial^2 u(x,y)}{\partial x \partial y} = 0, \ \left(\frac{\partial u(x,y)}{\partial x}\right)^r + \frac{\partial u(x,y)}{\partial x} - u(x,y) = 0. \tag{2.2.2}$$

For the case $r = 2$, this is a differential polynomial system $R = (u_{yy} - u_{xy}, u_x^2 + u_x - u) = 0$ in the jet space of second order $J^2 = \mathbb{C}^8$ and has jet variety $V(R) = \{(x,\ y,\ u,\ u_x,\ u_y,\ u_{xx},\ u_{xy},\ u_{yy}) \in J^2$: $u_{yy} - u_{xy} = 0,\ u_x^2 + u_x - u = 0\}$.

The **rifsimp** algorithm [18, 29] takes on input a ranking of partial derivatives. A ranking of derivatives [18] is a total ordering on the set of all jet variables (excluding independent variables) that is invariant under differentiation and satisfies a positivity condition.

Definition 2.2.1 (Highest Derivative)**.** *The highest derivative of a* PDE ψ *is the largest derivative appearing in the* PDE *in the ranking. A* PDE *is classified as leading linear or leading nonlinear with respect to a ranking according as whether it is linear or not in its highest derivative.*

EXAMPLE 2.2.2. *Consider the ranking of partial derivatives:*

$$u \prec u_x \prec u_y \prec u_{xx} \prec u_{xy} \prec \cdots \tag{2.2.3}$$

Then (3.2.2) has $\text{HD}(u_{yy} - u_{xy}) = u_{yy}$ *and* $\text{HD}(u_x^2 + u_x - u) = u_x$*. Hence* $u_{yy} - u_{xy}$ *is leading linear and* $u_x^2 + u_x - u$ *is leading nonlinear with respect to the ranking.*

Definition 2.2.2 (Formal Total Derivative)**.** *The formal total derivative is* $D_{x_j} = \frac{\partial}{\partial x_j} + \Sigma_{\ell=1}^{m} u_{x_j}^{\ell} \frac{\partial}{\partial u^{\ell}} + \cdots$ *Given a list of equations* $N = 0$*,* $D(N)$ *is the list of first order total derivatives of all equations of* N *with respect to all independent variables (i.e.* $D(N) := [D_{x_j} N_k]$*) and forms a single prolongation of* N*.*

For Example 2.2.1 with $N = u_x^2 + u_x - u = 0$ and formal total derivatives D_x and D_y we have:

$$D(N) = [2u_x u_{xx} + u_{xx} - u_x, 2u_x u_{xy} + u_{xy} - u_y].$$

Implementations of symbolic differential elimination algorithms have devoted much attention to decreasing the order of such prolongations, while still determining all the obstructions to the existence of formal power series solutions (e.g. see [20, 29]).

As input the **rifsimp** algorithm takes a polynomially nonlinear system of PDE, together with a ranking of derivatives [18]. It partitions the system into two disjoint sets: PDE which are linear in their highest derivatives with respect to the ranking, and its complement the leading nonlinear PDE. During its execution these two sets are continually updated until they satisfy certain theoretical conditions for termination [18, 29]. One condition is that the integrability conditions, after reduction with respect to the leading nonlinear PDE, should lie in the algebraic ideal generated by the leading nonlinear PDE. Also the once-differentiated set of the leading nonlinear PDE, after reduction with respect to the leading linear PDE, should lie in that ideal. Violation of these conditions gives new equations which are appended to the system, and the process above is repeated. In existing implementations [29] the membership is symbolically tested using Gröbner Bases and Triangular Set methods.

2.3 Numerical Algebraic Geometry

Our tool to numerically solve polynomial systems is homotopy continuation. Homotopy methods define families of systems, embedding a system to be solved in a homotopy, connecting it to a start system whose solutions are known. Continuation methods are then applied to track the paths defined by the homotopy, leading to the solutions. By random choices of constants in the homotopy one can prove that, except for an algebraic set of bad choices of constants, singularities and diverging paths can only occur at the end of the paths, when the system to be solved has singular solutions or fewer solutions than the generic root count.

In [24], a new field "Numerical Algebraic Geometry" was described which led to the development of homotopies to describe all irreducible components (all meaning: for all dimensions) of the solution set of a polynomial system. We briefly list key ingredients of this developing field, see also [25]:

1) Witness Sets: are the key data in a numerical irreducible decomposition. A witness set for a k-dimensional solution component consists of k random hyperplanes and all isolated solutions in the intersection of the component with those hyperplanes. The degree of the solution component equals the number of witness points. Witness sets are equivalent to lifting fibers in a geometric resolution [10].

2) Cascade of Homotopies: Candidate witness points are computed efficiently using a cascade of homotopies, peeling off the hyperplanes in going from high to lower dimensional solution components. This idea of cutting with hyperplanes to determine the dimensions of solution components appeared in Giusti and Heintz [6].

3) Factorization. Using monodromy loops, certified by linear traces, a pure dimensional solution set is factored into irreducibles. See e.g. [4] for the related approximate multivariate polynomial factorization.

4) Diagonal Homotopies. We intersect two solution components: A and B. Let A be defined by polynomial equations $f(u) = 0$, cut by hyperplanes $K(u) = 0$, and B similarly defined by $g(v) = 0$ and $L(v) = 0$. Consider the homotopy

$$H(u,v,t) = \begin{bmatrix} f(u) = 0 \\ g(v) = 0 \\ K(u) = 0 \\ L(v) = 0 \end{bmatrix} (1-t) + \begin{bmatrix} f(u) = 0 \\ g(v) = 0 \\ u - v = 0 \\ M(u) = 0 \end{bmatrix} t. \qquad (2.3.1)$$

For $t = 0$, we start at the product of the two witness sets $A \times B$. At $t = 1$, we are at the diagonal $u - v = 0$. The above homotopy is just a simple description of diagonal homotopies, see [23]. Intersecting components is done symbolically in the geometric resolution, see [10].

These methods have been implemented in PHCpack [27], see [22] for a description of some of its added capabilities. An interface to PHCpack within Maple is described in [11].

2.4 Symbolic-Numeric Completion Algorithm

We will present pseudo-code to describe our main algorithm **HybridRif** and its subroutines to find missing constraints.

2.4.1 Using Witness Sets

A basic step of our method is to detect whether a new constraint lowers the dimension of some component of the current system. As we append constraints, the general membership test of [22] simplifies to substituting the witness points of the current constraints into the presumed new constraint.

Algorithm 2.4.1. W^{new} = ***MembershipTest***(W, p, ϵ)
Input: *Witness set W; a polynomial p; a tolerance ϵ.*
Output: $W^{new} = \{w \in W : |p(w)| > \epsilon\}$.

For correctness of our methods we need to test whether the constraints are free from singularities. Numerically this is done by setting thresholds on the condition numbers of the witness points. While ill-conditioned points are not necessarily points with with multiplicity > 1 (extreme values of coefficients can lead to a bad scaling), for our current homotopy methods there is no difference in practice.

Algorithm 2.4.2. *boolean* = ***IsReduced***(W, ϵ)
Input: *Witness set W;*
ϵ tolerance on inverse condition number.
Output: *true if all points in W have good condition;*
false otherwise.

A witness set for a hypersurface (defined by one multivariate polynomial) is obtained by cutting the hypersurface by a general (random) line and applying a univariate root finder.

Algorithm 2.4.3. W = ***WitnessSet***(p, ϵ)
Input: *A polynomial p ; a tolerance ϵ.*
Output: *Witness set W for p.*

Diagonal homotopies [23] are used to compute a numerical representation of the intersection of two solution components given by witness sets.

Algorithm 2.4.4. W = ***Intersect***(W_1, W_2)
Input: *Witness sets W_1, W_2 representing A, B resp.*
Output: *Witness W representing $A \cap B$.*

2.4.2 Specification of rifsimp

In brief the **rifsimp** algorithm described in [18, 29] has the following input and output characteristics:

Algorithm 2.4.5. $[N, L, P] = \boldsymbol{rifsimp}(S, \prec)$

Input: *S, a system of polynomially nonlinear* PDE *and inequations over $\mathbb{Q}$, and a ranking $\prec$.*
Output: *[N,L,P], where*

- *L is a set of leading linear* PDE *in solved form with respect to its highest derivatives in the ranking $\prec$, where* $\text{HD}(L)$ *are all distinct, and no member of* $\text{HD}(L)$ *is a derivative of any other member;*
- *N is a set of leading nonlinear* PDE *with no dependence on* $\text{HD}(L)$ *or derivatives of* $\text{HD}(L)$*;*
- *P is a set of inequations (the pivots) with no dependence on on* $\text{HD}(L)$ *or derivatives of* $\text{HD}(L)$*;*
- *the integrability conditions of the leading linear* PDE *after reduction wrt L are in $\langle N \rangle$;*
- *$D(N)$ after reduction wrt L is in $\langle N \rangle$.*

In addition, an existence and uniqueness theorem is provided for its output [18]. We work with a modified version of the **rifsimp** algorithm, called **riflin**, which works with only the leading linear part so no membership tests are performed and $D(N)$ is not computed. In contrast to **rifsimp** only a subset of all constraints are determined and an existence uniqueness result can not be stated.

2.4.3 The main algorithm

The symbolic algebraic equation manipulation in **rifsimp** is replaced with the numerical diagonal homotopy method.

Algorithm 2.4.6. $[N, P, L, fail] = \boldsymbol{HybridRif}(S, \prec, \epsilon)$

Input : *S, a polynomially nonlinear system of differential equations and inequations over $\mathbb{Q}$ and a ranking $\prec$; a tolerance ϵ.*
Output: *$[N, P, L, fail]$, where*
N: leading nonlinear part;
P: pivots (inequations);
L: leading linear part;
fail: true if witness sets are not reduced.

$[N^{(0)}, P^{(0)}, L^{(0)}] :=$ ***riflin***$(S, \prec)$
$W^{(0)} :=$ ***WitnessSet***$(N^{(0)}, \epsilon)$
Repeat from $j = 1$
  $Q^{(j)} := N^{(j-1)} \cup P^{(j-1)} \cup L^{(j-1)} \cup D(N^{(j-1)})$
  $[N^{(j)}, P^{(j)}, L^{(j)}] :=$ ***riflin***$(Q^{(j)}, \prec)$
  if $\text{HD}(L^{(j-1)}) \neq \text{HD}(L^{(j)})$ *then*
    $[N^{(j-1)}, P^{(j-1)}, L^{(j-1)}] := [N^{(j)}, P^{(j)}, L^{(j)}]$;
  else
    $W^{(j)} := W^{(j-1)}$;
    Repeat p *in* $N^{(j)} \backslash N^{(j-1)}$
      $W^{new} :=$ ***MembershipTest***$(W^{(j)}, p, \epsilon)$;
      if $W^{new} \neq \emptyset$ *then*
        $W^{(j)} :=$ ***WitnessSet***(p, ϵ);
        $W^{(j)} :=$ ***Intersect***$(W^{new}, W^{(j)})$;
      end if;
    end loop;
    if $W^{(j)} = W^{(j-1)}$ *then*
      $fail :=$*not* ***IsReduced***$(W^{(j-1)}, \epsilon)$;
      return $[N^{(j-1)}, P^{(j-1)}, L^{(j-1)}, fail]$;
    end if;
    $j := j + 1$;
  end if;
end loop.

In practice **HybridRif** will abort reporting failure as soon as a witness set shows intolerably high condition numbers.

2.4.4 Termination Conditions

Algorithm **IsReduced** is implemented using estimates for the inverse condition number of the Jacobian matrix at the witness points.

A standard Noetherian argument, which is a minor variation of that in [18], shows that the linear part $L^{(j)}$ must eventually stabilize. Further it is easily shown that

$$\text{HD}(L^{(j)}) = \text{HD}(L^{(j-1)}) \Rightarrow P^{(j)} = P^{(j-1)}. \tag{2.4.1}$$

The condition **MembershipTest**$(W^{(j-1)}, N^{(j)}, \epsilon) = \emptyset$ used to terminate **HybridRif** corresponds to the symbolic test involving the difference of two varieties:

$$V(N^{(j)}) \backslash V(P^{(j-1)}) \supseteq V(N^{(j-1)}) \backslash V(P^{(j-1)}). \tag{2.4.2}$$

As **HybridRif** will fail when it encounters singularities or ill-conditioned representations, its termination is not absolute as is the case when the symbolic condi-

tions (2.4.1) and (2.4.2) are applied. However, when **HybridRif** terminates without failure, the final witness set can be certified as follows: every witness point is an approximate zero in the sense of [2].

2.5 Optimizations

It will be advantageous, but not theoretically necessary, to remove redundant equations. A polynomial is redundant if after its removal the geometry of the solution set has not changed. By repeated calls to Algorithm 2.4.1, we can implement the following.

Algorithm 2.5.1. N^{new} = ***Shrink***(N, W, ϵ)
Input: N *a set of polynomials;*
W *witness set representing* $N^{-1}(0)$*;*
ϵ *is tolerance for Algorithm 2.4.1.*
Output: N^{new} *cuts out same solution set as* N*.*

The number of paths followed by homotopy methods is perhaps the most important aspect of their computational cost. In the case of dense polynomial square systems, this number is given by the Bézout degree. As a system is prolonged (differentiated) the number of equations can grow dramatically and the product of the degrees of these equations (their Bézout degree) can grow even more explosively. Thus methods for decreasing this number are a priority in the development of homotopy methods for PDE systems.

It is also advantageous to fix the value of the independent variables to random numbers: $x = \hat{x}$ where x belongs to the space of independent variables X. Extended graphs of solutions of PDE belong to components which are fibred over X. This condition is ensured for each component of $V(N) \subseteq J^q$ that is fibred over X. Let π_X denote the projection onto X, that is $\pi_X(x, v) = x$ where $v = (u, ..., \underset{q}{u})$. Thus at neighborhoods $\mathcal{O}(\hat{x}, \hat{v})$ of regular points $(\hat{x}, \hat{v}) \in V(N)$ we have $\dim(\pi_X \mathcal{O}(\hat{x}, \hat{v}) \cap V(N)) = \dim X$. Equivalently we have $\dim \pi_X T_{(\hat{x},\hat{v})} V(N) = \dim X = n$ where $T_{(\hat{x},\hat{v})} V(N)$ is the tangent space to $V(N)$ at $(\hat{x}, \hat{v})$.

Suppose we are given a system with variety C and a hyper-surface S. We can already test $S \supseteq C$, by use of the algorithm **MembershipTest** and substitution of generic points, but the number of continuation paths can be impractically large. Instead we set $x = \hat{x}$, $C_{\hat{x}} = \{(x, v) \in C : x = \hat{x}\} = \ell_{\hat{x}} \cap C$ and $S_{\hat{x}} = \{(x, v) \in S : x = \hat{x}\} = \ell_{\hat{x}} \cap S$ where $\ell_{\hat{x}} = \{(\hat{x}, v)\}$ is a linear space. By application of **MembershipTest** and substitution of generic points we can determine if $S_{\hat{x}} \supseteq C_{\hat{x}}$ by following far fewer continuation paths. In general however this does not necessarily imply $S \supseteq C$. For example consider $S = \{(x, u) : (x - 3)(u - 1) = 0\}$ and $C = \{(x, u) : (u - 2) = 0\}$ then with $\hat{x} = 3$, $S_{\hat{x}} \supseteq C_{\hat{x}}$, but $S \not\supseteq C$. But note that components of form such as $x - 3 = 0$ are not fibred over X and are not of interest

for PDE, since they imply that the 'independent variables' are instead dependent on each other. Such non-fibred components are avoided, with high probability, by setting $x = \hat{x}$.

Assume C has only one component fibred over X and $S_{\hat{x}} \supseteq C_{\hat{x}}$. If $S \not\supseteq C$, then $C \cap S$ is a proper algebraic subset of C [21] which means $\dim C > \dim (C \cap S)$ and since C is fibred over X, $C_{\hat{x}}$ is not empty. Therefore $\dim C_{\hat{x}} > \dim (C \cap S \cap \ell_{\hat{x}}) = \dim (C_{\hat{x}} \cap S_{\hat{x}})$, contradicting $S_{\hat{x}} \supseteq C_{\hat{x}}$, so S must contain the whole component of C. This technique can often dramatically decrease the Bézout bound of the system and number of paths for the witness set by homotopies in **MembershipTest**. Note that the degree d of a PDE, when the independent variables are fixed to constants, is invariant under prolongation. Hence the Bézout degree of the prolongation of a single PDE, is d^N where N is the number of PDE in the prolongation.

2.6 Examples

2.6.1 Illustrative Example

The simple illustrative system (3.2.2) with $r = 2$ has

$$S := [u_{yy} - u_{xy} = 0,\ u_x^2 + u_x - u = 0] \tag{2.6.1}$$

on entry into **HybridRif**. We assume the ranking is given by (2.2.3). Since the independent variables x, y do not appear explicitly they are not used in dimension counts.

At the first iteration, applying **riflin** to S yields the single leading linear PDE in the solved form $L^{(0)}$, and a single leading nonlinear PDE $N^{(0)}$:

$$\begin{array}{lll} N^{(0)} & P^{(0)} & L^{(0)} \\ u_x^2 + u_x - u = 0 & \emptyset & u_{yy} = u_{xy} \end{array} \tag{2.6.2}$$

We first calculate $D(N^{(0)}) = [2u_x u_{xx} + u_{xx} - u_x, 2u_x u_{xy} + u_{xy} - u_y]$ then

$$Q^{(1)} = L^{(0)} \cup P^{(0)} \cup N^{(0)} \cup D(N^{(0)}) \tag{2.6.3}$$

and apply **riflin** to $Q^{(1)}$ to obtain

$$\begin{array}{lll} N^{(1)} & P^{(1)} & L^{(1)} \\ u_x^2 + u_x - u = 0 & (2u_x + 1) \neq 0 & u_{yy} = \frac{u_y}{2u_x+1} \\ u_y^2 - u_y u_x = 0 & & u_{xy} = \frac{u_y}{2u_x+1} \\ u_y u_x - u_y^2 = 0 & & u_{xx} = \frac{u_x}{2u_x+1} \end{array} \tag{2.6.4}$$

We remove the obvious duplicate equation in $N^{(1)}$ by a simple implementation of Algorithm 2.5.1 although this is not necessary for the correctness and termination of

HybridRif. Next we check whether the leading linear part is stable or not. Since $\text{HD}(L^{(1)}) = [u_{yy}, u_{xy}, u_{xx}] \neq \text{HD}(L^{(0)}) = [u_{yy}]$ we return to the beginning of the major loop. We first compute

$$\begin{aligned} D(N^{(1)}) = [&2u_xu_{xx} + u_{xx} - u_x, 2u_xu_{xy} + u_{xy} - u_y, \\ &-u_{xy}(u_x - u_y) - u_y(u_{xx} - u_{xy}), \\ &-u_{yy}(u_x - u_y) - u_y(u_{xy} - u_{yy})] \end{aligned} \tag{2.6.5}$$

then $Q^{(2)} = L^{(1)} \cup P^{(1)} \cup N^{(1)} \cup D(N^{(1)})$. Next **riflin** is applied to $Q^{(2)}$ which after removing redundant equations gives:

$$\begin{array}{lll} N^{(2)} & P^{(2)} & L^{(2)} \\ u_x^2 + u_x - u = 0 & (2u_x + 1) \neq 0 & u_{yy} = \frac{u_y}{2u_x+1} \\ u_yu_x - u_y^2 = 0 & & u_{xy} = \frac{u_y}{2u_x+1} \\ & & u_{xx} = \frac{u_x}{2u_x+1} \end{array} \tag{2.6.6}$$

Here $\text{HD}(L^{(2)}) = \text{HD}(L^{(1)})$, so the membership test is applied to $N^{(2)}$ to test $V(N^{(2)})\backslash V(P^{(2)}) \supseteq V(N^{(1)})\backslash V(P^{(1)})$.

First we compute the witness set of each polynomial in $N^{(1)}$ in (u, u_x, u_y)-space by **WitnessSet**. There are two paths to be followed for each polynomial. Then the witness set $W^{(1)}$ for $N^{(1)}$ is computed by **Intersect**, yielding four witness points resulting from tracking 4 paths. During the application of **MembershipTest** points in $W^{(1)}$ are evaluated in the system $N^{(2)}$. Since **IsReduced** $(W^{(1)}, \epsilon) = \mathit{true}$ and **MembershipTest**$(W^{(1)}, N^{(2)}, \epsilon) = \emptyset$, the termination conditions are met and the algorithm returns $[N^{(1)}, P^{(1)}, L^{(1)}]$.

For this example it can be checked that the outputs of **HybridRif** and the fully symbolic algorithm **rifsimp** are the same. In Section 2.6.2 an example is given where the outputs of **HybridRif** and **rifsimp** differ.

Comparison with a Numerical Geometrical Completion Method: Here we compare **HybridRif** with a numerical geometrical completion method [1, 7, 14] which is a variation of the symbolic Cartan-Kuranishi method [13, 20]. In [7] the first application of the interpolation-free method of [14] is given to the example system above. The method when applied to an input system R involves computing $\dim \pi^\ell D^k R$ where $\pi : J^q \to J^{q-1}$ is the usual projection until the criteria of projected involution [1] are satisfied. The output of the method of [7] consists of

$$\begin{aligned} &\phi^1 = 0, \phi^2 = 0, \\ &D_x(\phi^1) = 0, D_y(\phi^1) = 0, D_x(\phi^2) = 0, D_y(\phi^2) = 0, \\ &D_{xx}(\phi^2) = 0, D_{xy}(\phi^2) = 0, D_{yy}(\phi^2) = 0 \end{aligned} \tag{2.6.7}$$

where $R = [\phi^1 = u_{yy} - u_{xy} = 0,\ \phi^2 = u_x^2 + u_x - u = 0]$ is the input system above. In

[7] the following dimensions are computed using homotopy continuation:

$$\begin{aligned} \dim(R) &= 2 \quad \dim(DR) = 1 \quad \dim(D^2R) = 1 \\ \dim \pi(DR) &= 1 \quad \dim \pi(D^2R) = 1 \\ \dim \pi^2(D^2R) &= 1 \end{aligned} \tag{2.6.8}$$

and show $\pi(DR)$ is an involutive system. In the computations, the worst Bézout number that appears is 64 which is much bigger than 4, the number of continuation paths that had to be followed in the application of **HybridRif** above.

2.6.2 System for Discrete Symmetries

Reference [15] solves the problem of determining the full diffeomorphism pseudogroup of point transformations $(x, u) \mapsto (\hat{x}, \hat{u})$ of the form $\hat{x} = X(x, u), \hat{u} = U(x, u)$, leaving invariant the ODE

$$u_{xx} = \frac{1}{x} u_x + \frac{4}{x^3} u^2. \tag{2.6.9}$$

Requiring that these transformations leave the ODE invariant leads [15] to a system of nonlinear PDE for the unknown functions X, U:

$$\begin{aligned} &4U^2X_u^3 - X^3X_uU_{uu} + X^3U_uX_{uu} + X^2U_uX_u^2 = 0, \\ &X^2U_xX_u^2 + 2X^3U_uX_{xu} - X^3X_xU_{uu} - 2X^3X_uU_{xu} + 2X^2U_uX_xX_u \\ &\quad +12U^2X_xX_u^2 + X^3U_xX_{uu} = 0, \\ &x^3X^3U_xX_{xx} - 4u^2X^3U_uX_x - x^3X^3U_{xx}X_x + x^3X^2U_xX_x^2 \\ &\quad +4u^2X^3U_xX_u + 4x^3U^2X_x^3 = 0, \\ &2xX^3U_xX_{xu} + X^3U_xX_u - xX^3U_{xx}X_u + 2xX^2U_xX_xX_u - X^3U_uX_x \\ &\quad +xX^2U_uX_x^2 - 2xX^3U_{xu}X_x + xX^3U_uX_{xx} + 12xU^2X_x^2X_u = 0 \end{aligned} \tag{2.6.10}$$

augmented with the condition that the Jacobian of the transformation does not vanish: $X_xU_u - X_uU_x \neq 0$.

Application of the **HybridRif** Algorithm with the ranking graded first by total order of derivative, then with $\partial_u \prec \partial_x$ and finally lexicographically with $U \prec X$, i.e.:

$$U \prec X \prec U_u \prec X_u \prec U_x \prec X_x \prec \cdots \tag{2.6.11}$$

gives the leading linear system

$$\begin{aligned}
X_{xx} &= \frac{6\,X^3 - 5\,x^2 X_x U_u X - x^3 U_u {X_x}^2}{5x^3 X U_u}, \quad X_u = 0 \\
U_{xx} &= \left(20\,x^3 U^2 U_u {X_x}^3 - 20\,u^2 X^3 {U_u}^2 X_x - 5\,x^2 U_u X^3 U_x X_x \right. \\
&\quad \left. +4\,x^3 U_x X^2 U_u {X_x}^2 + 6\,U_x X^5\right)/(5x^3 U_u X_x X^3) \\
U_{xu} &= \frac{2\,x^3 U_u {X_x}^2 - 5\,x^2 X_x U_u X + 3\,X^3}{5x^3 X X_x} \\
U_{uu} &= 0
\end{aligned} \tag{2.6.12}$$

together with the condition $U_u \neq 0, X_x \neq 0$. The constraint leading nonlinear equations found by **HybridRif** are:

$$\begin{aligned}
& x^3 U_u {X_x}^2 - X^3 = 0, \\
& -200\,x^3 u {X_x}^2 X^3 {U_u}^2 + 200\,x^6 {X_x}^4 {U_u}^2 U - 27\,x^3 U_u X^4 {X_x}^2 + 36\,X^7 \\
& \quad -25\,x^4 {U_u}^2 X^3 {X_x}^2 + 16\,x^6 {U_u}^2 X {X_x}^4 = 0, \\
& -200\,x^6 {X_x}^3 {U_u}^2 U_x - 16\,x^6 {U_u}^2 {X_x}^4 + 12\,x^3 U_u X^3 {X_x}^2 + 90\,x X^5 U_u \\
& \quad -1200\,x^3 X^2 U_u {X_x}^2 U + 680\,x^3 u {X_x}^2 X^2 {U_u}^2 + 85\,x^4 {U_u}^2 X^2 {X_x}^2 \\
& \quad +720\,u X^5 U_u - 171\,X^6 - 200\,x^2 u X_x X^3 {U_u}^2 = 0, \\
& 432\,u X^{10} + 792\,x^4 X^7 {X_x}^2 U_u + 204\,x^7 X^4 {X_x}^4 {U_u}^2 - 60\,x^6 X^5 {X_x}^4 U_u \\
& \quad -3600\,x^3 X^7 {X_x}^2 U - 200\,x^9 X^2 {X_x}^5 {U_u}^2 U_x + 200\,x^4 u X^6 {X_x}^2 {U_u}^2 \\
& \quad -1200\,x^2 u X^8 X_x U_u - 2400\,x^6 X^5 {X_x}^3 U_x U_u - 2400\,x^6 X^4 {X_x}^4 U_u U \\
& \quad +1632\,x^6 u X^4 {X_x}^4 {U_u}^2 + 6336\,x^3 u X^7 {X_x}^2 U_u - 800\,x^9 {X_x}^6 {U_u}^2 U^2 \\
& \quad -1400\,x^5 u X^5 {X_x}^3 {U_u}^2 + 800\,x^6 u^2 {X_x}^4 {U_u}^3 X^3 - 990\,x^3 X^8 {X_x}^2 \\
& \quad +54\,x X^{10} = 0
\end{aligned} \tag{2.6.13}$$

Application of the initial data algorithm [29] to the leading linear equations (2.6.12) above yields the following initial data

$$\begin{aligned}
& X(x_0, u_0) = X^0,\ U(x_0, u_0) = U^0, \\
& X_x(x_0, u_0) = X_x^0,\ U_x(x_0, u_0) = U_x^0,\ U_u(x_0, u_0) = U_u^0.
\end{aligned} \tag{2.6.14}$$

Then the existence and uniqueness theorem [18] implies that formal power solutions to the system exist at points where the constants X^0, U^0, X_x^0, U_x^0, U_u^0 satisfy the constraint nonlinear equations (2.6.13).

In this example $N^{(1)}$ consists of the first 3 nonlinear equations in (2.6.13) with degrees 6,13,12 respectively, and the corresponding linear part ($L^{(1)}$) becomes stable. Next the witness set is constructed for $N^{(1)}$. Next $N^{(2)}$ is obtained with all 4 nonlinear equations of (2.6.13) with degrees 6, 13, 12 and 19 respectively. Application of **MembershipTest** shows that the fourth equation is geometrically new so the

witness set of its intersection is computed using **Intersect**.

Because of the high total degree in this example, we use techniques to decrease the number the continuation paths followed by **phc**. The first technique is to specialize the independent variables to random fixed values as discussed in Section 2.5. In particular the degrees of the uncovered constraints (2.6.13) decrease dramatically from 6, 13, 12, 19 to 3, 7, 6, 10. A second key to success, was to use mixed volumes instead of Bézout Bounds. In particular in the application of diagonal homotopies, this decreased the number of paths needing to be followed for $N^{(1)}$ from $126(= 3\cdot 7\cdot 6)$ to 3 and the number of paths for $N^{(2)}$ from $1260(= 3 \cdot 7 \cdot 6 \cdot 10)$ to 4.

Application of diagonal homotopies showed the existence of 1 dimensional components for the constraint nonlinear system (3 dimensional if we include x, u in the dimension count). This agrees with the explicit computations in [15]. Denote by $\mathcal{G}_{\text{lie}}$ the Lie subgroup of symmetries in a connected component of the identity of the full symmetry group $\mathcal{G}$ of the ODE. Our dimensional computation correctly reveals the dimension of $\mathcal{G}_{\text{lie}}$ as 1 as determined by a more conventional linearized calculation in [15]. The degree determined by our calculations is 4 and corresponds to the cardinality of $\mathcal{G}/\mathcal{G}_{\text{lie}}$ which is in agreement with [15] (indeed there it is shown that the factor group is isomorphic to $\mathbb{Z}_4$). Further calculations using **phc** on the full constraint nonlinear system reveals that there are 4 degree one, one dimensional components (fixing x, u to constants) whose equations can be interpolated if desired. These computations are again in agreement with the explicit ones in [15].

Interestingly high degree singular components of natural geometric origin violating the invertibility condition $X_xU_u - X_uU_x \neq 0$ arose in our calculations and initially caused some numerical difficulties. Such components were excluded by inclusion of the invertibility condition. For the system above this is equivalent to $X_xU_u \neq 0$, since $X_u = 0$. Consequently we also have $X \neq 0$ and $U \neq 0$.

Comparison with a Numerical Geometrical Completion Method: We compare **HybridRif** with a numerical geometrical completion method [1, 7, 14] which is a variation of the symbolic Cartan-Kuranishi method [13, 20]. The method when applied to an input system R (2.6.13) involves computing $\dim \pi^\ell D^k R$ until the criteria of projected involution [1] are satisfied. The system R has Bézout number 12288 which is reduced to 1875 after substituting random values for the independent variables. The prolongation of DR which has 18 equations with Bézout number 50096498540544. After specializing the independent variables it reduces to 177978515625 which was still too high.

Comparison with the rifsimp symbolic algorithm: Application of **rifsimp**

with the ranking (2.6.11) yielded the leading linear system:

$$\begin{aligned} U_{x,x} &= (-16x^2uU^2 - 4x^3U^2 - 128xu^2U^2 + \\ &\quad -384u^2x^2U_xU - 16x^4U_xU - 160x^3uU_xU + \\ &\quad +4x^2uXU + 16xu^2XU + 128u^3XU - 3x^4XU_x + \\ &\quad -80u^2x^2U_xX - 32x^3uU_xX)(x^3(4u+x)^2(X+8U)) \\ X_x &= (U^2(4U+X)(8u+x))/ \\ &\quad (u(32uU^2 + 4xU^2 + 8x^2U_xU + 32uxU_xU \\ &\quad +8uXU + xXU + 4xuXU_x + x^2U_xX)) \\ X_u &= 0 \\ U_u &= \frac{X+8U}{8u+x} \end{aligned} \tag{2.6.15}$$

with the leading nonlinear equations

$$\begin{aligned} 32u^3XU - 64u^2x^2U_xU - 32xu^2U^2 - 8u^2x^2U_xX \\ +8xu^2XU - 32u^2x^3U_x^2 - 2x^3uU_xX - 16x^3uU_xU \\ +x^2uXU - 12ux^4U_x^2 - 8x^2uU^2 - x^3U^2 - x^5U_x^2 = 0, \\ (uX - xU)(xX + 4uX + 4xU) = 0 \end{aligned} \tag{2.6.16}$$

and the inequations $X + 8U \neq 0, X \neq 0, U \neq 0$.

Unlike the example of Section 2.6.1, this differs from the result obtained by **HybridRif**. This discrepancy is resolved by noting that both systems define the the same locus in jet space.

Finally we note that Hydon [8] gives an elegant and efficient method which exploits the knowledge of the Lie group $\mathcal{G}_{\text{lie}}$ to considerably ease computation of the full group $\mathcal{G}$.

2.6.3 Random first order ODE

In this section we apply our symbolic-numeric approach to a class of random ODE $R(u_x, u) = 0$ for a single dependent variable u. The efficiency of this approach is compared with that of using the symbolic **rifsimp** algorithm. Differentiation of $R(u_x, u) = 0$ gives $R_{u_x}u_{xx} + R_uu_x = 0$. The following cases are easily obtained:

$$\begin{aligned} &\text{Case 1: } u_{xx} = -\tfrac{R_u u_x}{R_{u_x}}, R_{u_x} \neq 0,\ R(u_x, u) = 0 \\ &\text{Case 2: } S_2 = \{R = 0, R_u = 0, R_{u_x} = 0\},\ u_x \neq 0 \\ &\text{Case 3: } S_3 = \{R(0,u) = 0, R_{u_x}(0,u) = 0\},\ u_x = 0 \end{aligned} \tag{2.6.17}$$

For random differential polynomials R, system S_2 in (2.6.17) consists of two random polynomials in one variable and system S_3 in (2.6.17) consists of three random poly-

d	2	3	4	5	6	7	8	9
RAM:$\mathbb{Z}$	0.24	0.6	1.4	2.5	8.3	16.5	128.7	INC
RAM:$\mathbb{C}$	3.40	6.8	11.2	20.7	62.2	INC	INC	INC

Table 2.1: rifsimp memory consumption (MB) applied to a class of random polynomial ODE $\mathbf{R}(\mathbf{u_x}, \mathbf{u}) = \mathbf{0}$ with integer coefficients, and a class with complex rational coefficients. Here $d = \text{degree}(R)$. RAM=INC indicates the memory of machine was exhausted.

d	2	5	8	11	14	17	20
RAM:$\mathbb{Z}$	1.5	1.8	2.0	2.1	2.2	2.3	2.5
RAM:$\mathbb{C}$	1.5	1.8	2.0	2.0	2.2	2.3	2.6

Table 2.2: phc memory consumption (MB) applied to a class of random polynomial ODE $\mathbf{R}(\mathbf{u_x}, \mathbf{u}) = \mathbf{0}$ with integer coefficients, and a class with complex rational coefficients. Here $d = \text{degree}(R)$.

nomials in two variables. For random R systems S_2 and S_3 will be inconsistent with high probability and Case 1 will be the only consistent case. A full analysis of all the singular cases for such ODE is given in the classic work of Hubert [9].

Two subclasses of random ODE with degrees d from 1 to 20 were considered. One subclass had random coefficients consisting of integers between -99 and 99, and the other random subclass had random coefficients consisting of complex numbers of the form $(a+bi)/(\max\{|a|,|b|\}+1)$ where a, b are random integers between -10 and 10.

The computations were carried out using Maple 9, and **phc** (release 2.3 beta) on a 1.5 GHZ Pentium M, with 512 MB of RAM, running under Windows XP. As shown in Table 2.1 the RAM was exhausted at relatively low degree $d = 9$, and this was dramatically worse for complex coefficients where exhaustion occurred at $d = 7$. As seen in Table 2.2 RAM usage by **phc** was dramatically lower and more stable than that of **rifsimp**. While changing from random integer to complex coefficients barely affected the RAM consumed by **phc**, it dramatically increased RAM usage by **rifsimp**.

Degree-time statistics for **rifsimp** and **phc** are shown in Figure 2.6.1. The positive concavity of the two curves for **rifsimp** indicates its complexity is more than polynomial. The approximately linear curves for **phc** in Figure 2.6.1 on the log-log scale is typical for a polynomial-time method. However the worst case complexity of **HybridRif** is at least exponential, considering its application to systems of linear homogeneous PDE in a single dependent variable. In that case its output is isomorphic to a Gröbner Basis. Groundbreaking work on reducing the complexity to polynomial time for ODE was done by Sedglovacic [19]. The memory usage statistics show the discrepancy between **rifsimp** and **phc** growing with increasing degree, and when changing from integer to random complex coefficients. The symbolic differential elimination program **Rosenfeld_Groebner** had similar memory and time behavior to **rifsimp** on the random class of ODE.

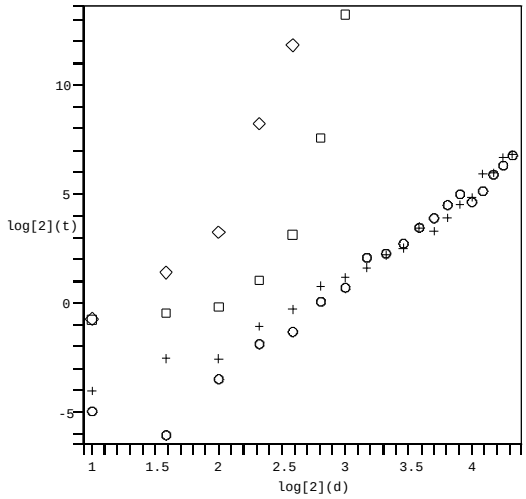

Figure 2.6.1: Time-Degree Statistics for a Random polynomial ODE $\mathbf{R}(\mathbf{u_x}, \mathbf{u}) = \mathbf{0}$ plotted on a $\log - \log$ scale. $d = \text{degree}(R)$ and t is the time to apply the differential elimination process using: rifsimp (comp. coeffs. $\Diamond$); rifsimp (integ. coeffs. $\Box$); phc (comp. coeffs. $+$); phc (integ. coeffs. o).

2.7 Discussion

Our method applies to intrinsic (exact) systems of polynomially nonlinear PDE and relies on splitting the system into a leading linear subsystem and its complement. Well-developed (linear) symbolic methods are applied to the leading linear part of the system. The success of this strategy enables the shrinking of the number of genuinely nonlinear equations that are dealt with by the numerical continuation methods. The use of diagonal homotopies allows handling the constraints incrementally, exploiting the structure of the leading nonlinear systems, and leads to a further decrease in the number of paths to be followed. Note that one could – at least in theory – replace the use of witness sets and diagonal homotopies in **HybridRif** by lifting fibers and using geometric resolutions [10]. In contrast to Gröbner methods, the fact that only geometrically new constraints are used means that generally fewer constraints need to be stored than would be required to represent the ideal. In addition, the maintenance of the constraints in their introduced form helps to preserve sparsity, and reduce equation and coefficient growth typical of Gröbner methods. It also allows flexibility in using alternative and sparse methods to control expression swell. Such methods include encoding the constraints by straight line programs, or using memory management based on ordered storage strategies [29] or directed acyclic graph structures as used by Lecerf in his implementation of the algorithms in [10].

The methods were applied to a number of examples starting with an easy illustrative example in Section 2.6.1. Secondly a system for discrete symmetries of moderate difficulty for symbolic methods was considered. Although the output was implicit, it illustrated that useful features of the symmetries could be extracted by the new

hybrid methods (such as the number of discrete symmetries, and the degree of the components of the group). On this example, **HybridRif** was compared with a geometrical approach based on a numerical version of the Cartan-Kuranishi algorithm. It demonstrated that far fewer continuation paths were needed by **HybridRif** than the numerical geometrical method developed in earlier work.

Finally in Section 2.6.3 we considered a class of random first order ODE. On systems which are denser and of higher degree, numerical methods have an advantage while symbolic methods can perform better on lower degree, highly structured sparse systems. We caution that the sample size is too small to make emphatic statements. Certainly it indicates that there is scope to improve **rifsimp**'s algebraic processing by using alternative symbolic and numeric algorithms.

This paper belongs to a series initiated in [28], continued in [14] and [7], aimed at developing "Numerical Jet Geometry", as a subfield of "Numerical Algebraic Geometry". Ultimately, this development will lead to methods enabling the practical processing of approximate input systems.

Bibliography

[1] J. Bonasia, F. Lemaire, G. Reid, R. Scott, and L. Zhi. Determination of Approximate Symmetries of Differential Equations. *Centre de Recherches Mathématiques, CRM Proceedings and Lecture Notes.* Vol 39, pages 233–250, 2004.

[2] L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and Real Computation.* Springer–Verlag, 1998.

[3] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Representation for the radical of a finitely generated differential ideal. Proc. ISSAC 1995. ACM Press. 158–166, 1995.

[4] G. Chèze and A. Galligo. Four Lectures on Polynomial Absolute Factorization. In A. Dickenstein and I.Z. Emiris (Eds.), *Solving Polynomial Equations: Foundations, Algorithms, and Applications.* Volume 14 of *Algorithms and Computation in Mathematics 14*, Springer-Verlag, pages 339–392, 2005.

[5] Y. Chen and X.-S. Gao. Involutive Bases of Algebraic Partial Differential Equation Systems. Science in China (A), 33(2), page 97–113, 2003.

[6] M. Giusti and J. Heintz. La détermination de la dimension et des points isolées d'une variété algébrique peuvent s'effectuer en temps polynomial. In D. Eisenbud and L. Robbiano, eds., *Computational Algebraic Geometry and Commutative Algebra, Cortona 1991*, vol. XXXIV of *Symposia Mathematica*, pages 216–256. Camb. Univ. Press, 1993.

[7] K. Hazaveh, D.J. Jeffrey, G.J. Reid, S.M. Watt, and A.D. Wittkopf. An exploration of homotopy solving in Maple. Proc. of the Sixth Asian Symp. on Computer Math. (ASCM 2003). Lect. Notes Series on Computing by World Sci. Publ. 10 edited by Z. Li and W. Sit (Singapore/River Edge, USA) 145–162, 2003.

[8] P.E. Hydon. Discrete point symmetries of ordinary differential equations. *R. Soc. Lond. Proc. Ser. A Math. Phys. Eng. Sci.* **454** 1961–1972, 1998.

[9] E. Hubert. Detecting degenerate cases in non-linear differential equations of first order. *Theoretical Computer Science* 187(1-2): 7–25, 1997.

[10] G. Lecerf. Computing the equidimensional decomposition of an algebraic closed set by means of lifting fibers. *J. Complexity* 19(4):564–596, 2003.

[11] A. Leykin and J. Verschelde. PHCmaple: A Maple Interface to the Numerical Homotopy Algorithms in PHCpack. In Quoc-Nam Tran, ed., *Proceedings of the Tenth International Conference on Applications of Computer Algebra (ACA'04)*, pages 139–147, 2004.

[12] E. Mansfield. *Differential Gröbner Bases.* Ph.D. thesis, Univ. of Sydney, 1991.

[13] J.F. Pommaret. *Systems of Partial Differential Equations and Lie Pseudogroups.* Gordon and Breach Science Publishers, Inc. 1978.

[14] G. Reid, C. Smith, and J. Verschelde. Geometric completion of differential systems using numeric-symbolic continuation. *SIGSAM Bulletin* 36(2):1–17, 2002.

[15] G.J. Reid, D.T. Weih and A.D. Wittkopf. A Point symmetry group of a differential equation which cannot be found using infinitesimal methods. In Modern Group Analysis: Advanced Analytical and Computational Methods in Mathematical Physics. Edited by N.H. Ibragimov, M. Torrisi and A. Valenti. Kluwer, Dordrecht, 93–99, 1993.

[16] G.J. Reid, A.D. Wittkopf and A. Boulton. Reduction of systems of nonlinear partial differential equations to simplified involutive forms. *Eur. J. of Appl. Math.* 7: 604–635.

[17] G.J. Reid, P. Lin, and A.D. Wittkopf. Differential elimination-completion algorithms for DAE and PDAE. *Studies in Applied Math.* 106(1): 1–45, 2001.

[18] C.J. Rust, *Rankings of derivatives for elimination algorithms and formal solvability of analytic partial differential equations*, Ph.D. Thesis, University of Chicago, 1998.

[19] A. Sedoglavic. A probabilistic algorithm to test local algebraic observability in polynomial time. *J. Symbolic Computation* 33(5): 735–755, 2002.

[20] W.M. Seiler. *Involution - The formal theory of differential equations and its applications in computer algebra and numerical analysis.* Habilitation Thesis, Univ. of Mannheim, 2002.

[21] A.J. Sommese and J. Verschelde. Numerical homotopies to compute generic points on positive dimensional algebraic sets. *Journal of Complexity* 16(3):572-602, 2000.

[22] A.J. Sommese, J. Verschelde, and C.W. Wampler. Numerical irreducible decomposition using PHCpack. In M. Joswig and N. Takayama, editors, *Algebra, Geometry, and Software Systems*, pages 109–130. Springer–Verlag, 2003.

[23] A.J. Sommese, J. Verschelde, and C.W. Wampler. Homotopies for intersecting solution components of polynomial systems. *SIAM J. Numer. Anal.* 42(4):1552–1571, 2004.

[24] A.J. Sommese and C.W. Wampler. Numerical algebraic geometry. In *The Mathematics of Numerical Analysis*, Volume 32 of *Lectures in Applied Mathematics*, edited by J. Renegar, M. Shub, and S. Smale, 749–763, 1996. Proceedings of the AMS-SIAM Summer Seminar in Applied Mathematics, Park City, Utah, July 17-August 11, 1995, Park City, Utah.

[25] A.J. Sommese and C.W. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science*. World Scientific Press, Singapore, 2005.

[26] J. Tuomela and T. Arponen. On the numerical solution of involutive ordinary differential systems. *IMA J. Numer. Anal.* 20: 561–599, 2000.

[27] J. Verschelde. Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. *ACM Transactions on Mathematical Software* 25(2): 251–276, 1999. Software available at `http://www.math.uic.edu/~jan`.

[28] A. Wittkopf and G.J. Reid. Fast differential elimination in C: The CDiffElim environment. *Computer Physics Communications*, 139: 192–217, 2001.

[29] A. Wittkopf. *Algorithms and Implementations for Differential Elimination*. Ph.D. Thesis, Simon Fraser University, 2004.

[30] W.-T. Wu. On the foundations of algebraic differential geometry. *Mathematics-Mechanization Research Preprint* No. 3, pages 1–26, 1989.

Chapter 3

Application of Numerical Algebraic Geometry and Numerical Linear Algebra to PDE

The computational difficulty of completing nonlinear PDE to involutive form by differential elimination algorithms is a significant obstacle in applications. We apply numerical methods to this problem which, unlike existing symbolic methods for exact systems, can be applied to approximate systems arising in applications.

We use Numerical Algebraic Geometry to process the lower order leading nonlinear parts of such PDE systems. The irreducible components of such systems are represented by certain generic points lying on each component and are computed by numerically following paths from exactly given points on components of a related system. To check the conditions for involutivity Numerical Linear Algebra techniques are applied to constant matrices which are the leading linear parts of such systems evaluated at the generic points. Representations for the constraints result from applying a method based on Polynomial Matrix Theory.

Examples to illustrate the new approach are given. The scope of the method, which applies to complexified problems, is discussed. Approximate ideal and differential ideal membership testing are also discussed.

3.1 Introduction

Over and under-determined (*non-square*) systems of ODE and PDE arise in applications such as constrained multibody mechanics and control systems. For example, differential-algebraic equations (DAE) arise from constrained Lagrangian mechanics (see [1] and the references therein).

Much progress has been made in exact differential elimination methods, theory and algorithms for polynomially nonlinear systems of PDE [3, 8, 14, 20, 19]. Such methods enable the identification of all hidden constraints of PDE systems and the

computation of initial data and associated formal power series solutions in the neighborhood of a given point. Algorithmic membership tests (specifically in the radical of a differential ideal) can be given [3, 8]. They can ease the difficulty of numerical solution of DAE systems [1].

This paper is a sequel to [17] and [18] in which theory and methods are developed for using numerical homotopy continuation techniques in the differential elimination process. In [17] such methods were first introduced by combining the Cartan-Kuranishni approach with homotopy methods to identify missing constraints for PDE. Our tool to numerically solve polynomial systems is homotopy continuation. When applied to PDE we stress that the solutions obtained by Homtopy continuation are not graphs of solutions of the PDE but instead zeros of the functions defining the PDE. Homotopy methods define families of systems, embedding a system to be solved in a homotopy, connecting it to a start system whose solutions are known. Such methods track the paths defined by the homotopy, leading to the solutions.

In [23], a new field "Numerical Algebraic Geometry" was described which led to the development of homotopies to describe all irreducible components (all meaning: for all dimensions) of the solution set of a polynomial system. Witness Sets are the key data in a numerical irreducible decomposition. A *witness set* for a k-dimensional solution component consists of k random hyperplanes and all isolated solutions in the intersection of the component with those hyperplanes. The degree of the solution component equals the number of witness points. Witness sets are equivalent to lifting fibers in a geometric resolution [10].

During the application of the Cartan-Kuranishi approach all equations are differentiated up to the current highest derivative order, resulting in potentially large numbers of PDE. These PDE are treated as polynomial equations in jet space, and their large number implies that the number of continuation paths that must be tracked can be impractically large in a direct application of Homotopy methods.

A hybrid method is introduced in [18] to exploit the structure of such systems to make progress in dealing with the difficulty above. However the hybrid method uses exact linear algebra (Gaussian Elimination) to process the leading linear part of such systems, and so is not applicable to approximate systems since it is unstable. In this paper we instead use stable methods from Numerical Linear Algebra.

In particular we use a numerical version of the geometric Cartan-Kuranishi method. This yields a coordinate independent split between leading linear and nonlinear systems, which grades only by total order of derivative, and not within derivatives of the same order. This independence aids numerical stability. Since the derivatives of leading nonlinear equations are leading linear with respect to highest order jet variables, the new PDE are viewed as linear equations corresponding to a coefficient matrix with polynomial entries. We apply the Singular Value Decomposition (a fundamental technique of Numerical Linear Algebra) to the null spaces of these polynomial matrices. This construction is based on a modification due to [2] of the classical criterion of involution for PDE (see [9, 15, 20] for the classical criterion).

3.2 PDE in Jet Space

There are several theoretical approaches to systems of PDE such as differential algebra, exterior differential systems and the so-called formal theory built on the jet bundle formalism. Jet space methods associate a given PDE system with a locus of points in a Jet space. Such methods concern the geometrical study of this locus and its relationship with the solutions of the differential equations [9, 15, 20].

3.2.1 Jet Space and Jet variety of a PDE

Our tools are applicable to systems of polynomially nonlinear PDE with complex-valued variables and solutions. Consider a polynomially nonlinear system of PDE $R = (R^1, \ldots, R^l) = 0$ with independent variables $x = (x_1, \ldots, x_r) \in \mathbb{C}^r$ and complex-valued dependent variables $u = (u^1, \ldots, u^s)$. We define a multi-index q as an r-tuple $[q_1, q_2, ..., q_r]$ with $q_i \in \mathbb{N}$. The order of the multi-index q, denoted $|q|$, is given by the sum of the q_i. As in [3, 20] solutions and derivatives are replaced by formal (jet) variables. In particular, denoting the p-th order jet variables corresponding to derivatives as $\underset{p}{u}$, the jet variety (locus) of a q-th order system in the jet space $J^q(\mathbb{C}^r, \mathbb{C}^s) \approx \mathbb{C}^{r_q}$ is

$$V(R) := \{(x, u, \underset{1}{u}, \ldots, \underset{q}{u}) \in J^q : R(x, u, \underset{1}{u}, ..., \underset{q}{u}) = 0\} \, . \tag{3.2.1}$$

Here $r_q = r + s\binom{r+q}{q}$ is the number of independent variables, dependent variables and derivatives of order less than or equal to q. We will use the shorthand $J^q(\mathbb{C}^r, \mathbb{C}^s) \equiv J^q$.

EXAMPLE 3.2.1. *We use the following running example [16, 7]:*

$$\begin{aligned} \frac{\partial^2 u(x,y)}{\partial y^2} - \frac{\partial^2 u(x,y)}{\partial x \partial y} &= 0, \\ \left(\frac{\partial u(x,y)}{\partial x}\right)^2 + \frac{\partial u(x,y)}{\partial x} - u(x,y) &= 0 \, . \end{aligned} \tag{3.2.2}$$

This is a differential polynomial system $R = (u_{yy} - u_{xy}, u_x^2 + u_x - u) = 0$ in the jet space of second order $J^2 \approx \mathbb{C}^8$ and has jet variety $V(R) = \{(x,\ y,\ u,\ u_x,\ u_y,\ u_{xx},\ u_{xy},\ u_{yy}) \in J^2$: $u_{yy} - u_{xy} = 0$, $u_x^2 + u_x - u = 0\}$.

3.2.2 Prolongation and Projection

There are two fundamental operations, prolongation and projection, to manipulate the locus in Jet space. We give a brief description of them here. For details see [15]. Before we define prolongation of a PDE system, we introduce the operator of Formal

Total Derivation

$$D_{x_j} = \frac{\partial}{\partial x_j} + \sum_{\ell=1}^{s} u^{\ell}_{x_j} \frac{\partial}{\partial u^{\ell}} + \cdots .$$

Given a list of equations $R = 0$, $\mathbf{D}(R)$ is the list of first order total derivatives of all equations of R with respect to all independent variables:

$$\mathbf{D}(R) := \{(x, u, \ldots, \underset{q+1}{u}) \in J^{q+1} : R = 0, D_{x_i} R_k = 0\} . \tag{3.2.3}$$

It forms a single prolongation of R.

For example, let $R = u_x^2 + u_x - u = 0$, then:
$\mathbf{D}(R) = \{(x, y, u, u_x, u_y, u_{xx}, u_{xy}, u_{yy}) \in J^2 :$
$u_x^2 + u_x - u = 0,\ 2u_x u_{xx} + u_{xx} - u_x,\ 2u_x u_{xy} + u_{xy} - u_y\}$.
Prolongation extends the locus of a PDE system from lower order jet space to higher order space. An inverse operation, the so-called projection, maps the locus from higher to lower order jet space.

Definition 3.2.1 (Projection)**.** *Given a jet variety R in J^q, a single projection is:*

$$\boldsymbol{\pi}(R) := \{(x, u, \underset{1}{u}, \ldots, \underset{q-1}{u}) \in J^q : \exists \underset{q}{u}, R(x, u, \underset{1}{u}, ..., \underset{q}{u}) = 0\}.$$

Let $T_p V(R)$ denote the tangent space to $V(R)$ at a given point $p \in V(R)$ and $\mathcal{N}(p)$ be a neighborhood of p. We restrict to the case where $\dim \boldsymbol{\pi}^q (\mathcal{N}(p) \cap T_p V(R)) = r$, that is the r variables x are independent and dim is the dimension as a complex manifold. Here $\boldsymbol{\pi}^q : J^q \to \mathbb{C}^r$ is the projection onto the space of variables $x \in \mathbb{C}^r$.

3.2.3 Formally Integrable and Involutive Systems

The symbol of a system of PDE R of order q is the Jacobian of its equations with respect to the highest derivatives:

$$\mathcal{S}R := \frac{\partial R}{\partial \underset{q}{u}} . \tag{3.2.4}$$

The computational characterization for the symbol being involutive is that in a δ-regular coordinate system

$$\operatorname{rank} \mathcal{S}DR = \sum_{k=1}^{r} k \beta_k^{(q)} . \tag{3.2.5}$$

Alternatively Spencer's involutivity test based on homology groups (and implementable using numerical linear algebra) can be used and this avoids the difficulty of δ-irregular coordinate systems. See [15, 20] for details and the definition of the

characters $\beta_k^{(q)}$. The most important properties of an involutive system of PDE are that $\overline{\boldsymbol{\pi}\mathbf{D}R} = V(R)$ and the symbol of R is involutive. An involutive system is also a formally integrable system. That is for any $k \geqslant 0$:

$$\overline{(\boldsymbol{\pi}\mathbf{D})(\mathbf{D}^k R)} = V(\mathbf{D}^k R) \ . \tag{3.2.6}$$

Remark 3.2.1. *In this paper $\overline{S}$ means the Zariski Closure of the set S which is the intersection of all varieties containing S. Since the projection of a variety may not be a variety, it is necessary to consider the Zariski closure. It is easy to show that $\boldsymbol{\pi}\boldsymbol{D}R = V(R)$ implies $\overline{\boldsymbol{\pi}\boldsymbol{D}R} = V(R)$.*

3.2.4 Cartan-Kuranishi Completion

The full geometric method to complete systems of partial differential equations is the Cartan-Kuranishni algorithm [19, 20]. This method prolongs the system to order $q+1$, then projects to order q to test for the existence of new constraints. This is continued until no new constraints are found. If the symbol of the resulting q-th order system is involutive, then the method has terminated and the system is involutive. If the symbol is not involutive, the system is prolonged until its symbol becomes involutive. The system is again tested for the existence of constraints by prolongation and projection. See [19, 20] for the relevant definitions. In particular the main iteration involves comparing R and $\boldsymbol{\pi}\mathbf{D}(R)$. Note in general the locus of R contains that of $\overline{\boldsymbol{\pi}\mathbf{D}R}$. A probabilistic method to check the involutivity of the symbol using Numerical Linear Algebra, and in particular the Singular Value Decomposition, is given in [26, Section 6]. Numerical difficulties can occur, if there are multiplicities, and that case is under investigation.

3.3 Polynomial Matrix

In this section we will exploit the linearity of the PDE which always appears after prolongation. Suppose $R = (R^1, \ldots, R^l) = 0$ is a polynomially nonlinear system of PDE with independent variables $x = (x_1, \ldots, x_r)$ and dependent variables $u = (u^1, \ldots, u^s)$. If the order of R is q, then we can represent the prolongation of R as:

$$\mathbf{D}R = \{\mathcal{S} \cdot \underset{q+1}{u} + \mathbf{r}, R\} \tag{3.3.1}$$

where $\mathcal{S}$ is called the *Symbol Matrix* of $\mathbf{D}R$. The corresponding augmented matrix is denoted by $[\mathcal{S}, \mathbf{r}]$. Obviously they are matrices with polynomial coefficients.

We briefly review some polynomial matrix theory and the associated results on rank and null-space computation. We let $\mathcal{R}$ denote the polynomial ring $K[z]$ in this paper, where $z = (z_1, ..., z_s)$ and the field K can be $\mathbb{R}$ or $\mathbb{C}$. The ring $\mathcal{R}$ is an integral

domain and also is a unique factorization domain. $Q(\mathcal{R})$ is the quotient field of $\mathcal{R}$ or say rational functions in the variables $z_1, ..., z_s$.

Definition 3.3.1. *The set of all $m \times n$ matrices with entries from $\mathcal{R}$ is denoted by $M^{m\times n}(\mathcal{R})$. Each member in $M^{m\times n}(\mathcal{R})$ is called a polynomial matrix over $\mathcal{R}$.*

3.3.1 Rank of Polynomial Matrix

Consider the column vectors of a polynomial matrix $A = (\alpha_1|\alpha_2|...|\alpha_n) \in M^{m\times n}(\mathcal{R})$ and assume $y_k \in \mathcal{R}$ for $k = 1, ..., n$. If $\sum_{k=1}^{m} y_k\alpha_k = 0^{m\times 1}$ implies $y_k = 0$ for $k = 1, ..., n$, then these vectors are said to be *linearly independent*. Otherwise these vectors are said to be *linearly dependent*.

Definition 3.3.2 (Rank)**.** *The (column) rank of polynomial matrix $A \in M^{m\times n}(\mathcal{R})$ is the maximum number of linearly independent column vectors of A.*

Several other frequently used definitions of rank are equivalent to our definition over a polynomial ring $\mathcal{R}$ since it is an integral domain. For example in the book [4], (algebraic) rank is generalized to arbitrary commutative rings using ideals generated by the minors.

Theorem 3.3.1. *Let $A \in M^{m\times n}(\mathcal{R})$. Then* $\operatorname{rank}(A) = k$ *if and only if any $t \times t$ minor of A is zero when $t > k$ and there exist some $k \times k$ nonzero minors.*

By Theorem 6.2.1, the rank of a polynomial matrix with coefficient field $K = \mathbb{R}$ will not change when the K is extended to $\mathbb{C}$. Moreover the rank evaluation of a polynomial matrix can be reduced to a constant matrix by choosing a random point in $\mathbb{C}^s$. In Sommese and Wampler's book [23], the concept of a *generic point* over $\mathbb{C}$ is introduced, which plays an essential role in "Numerical Algebraic Geometry". Suppose some property P is satisfied everywhere except on a proper algebraic subset U of an irreducible variety V. We call the points in $V\backslash U$ generic points. Then $\dim V > \dim U$, so $V\backslash U$ is dense in V (with the standard Lebesgue measure 1). So we say P holds with *algebraic probability one* for a random point of V. The following proposition easily follows:

Proposition 3.3.1. *For any generic point $z_0 \in \mathbb{C}^s$ we have* $\operatorname{rank}(A) = \operatorname{rank}(A_{z_0})$.

Remark 3.3.1. *In Numerical Algebraic Geometry generic points in $\mathbb{C}^s$ can be produced by choosing points in $\mathbb{C}^s$ randomly. With probability 1, the rank of a polynomial matrix is equal to the rank of the matrix evaluated at some random point (actually this result is also valid in $\mathbb{R}$ by Schwartz-Zippel theorem). That is, this will fail only on some algebraic variety with standard Lebesgue measure 0 in the whole space. This reduces the cost of rank computation dramatically.*

The witness points of a variety V yield a finite number of generic points on each irreducible component of V. This set is denoted by $W(V)$. Note that the witness

points of a polynomial system R is $W(V(R))$ and shortly we denote it by $W(R)$. A useful result in [18] is that each point in $W(V)$ is contained in another variety V' implies $V \subseteq V'$ with probability 1.

3.3.2 Computing the Null-space

Given a polynomial matrix $A \in M^{m\times n}(\mathcal{R})$, there exist $r = n - \text{rank}(A)$ linearly independent polynomial vectors $\{f_i\}$ such that $Af_i = 0^{m\times 1}$. Let $F := [f_1, ..., f_r]$, then $AF = 0^{m\times r}$. In particular F generates a linear space of A over quotient field $Q(\mathcal{R})$, which is called the *null-space* of A over $Q(\mathcal{R})$ and is denoted by $NullSpace(A)$. F is called a *basis* of $NullSpace(A)$. Note that F may not be a module basis of the Syzygy module of A. In this section, we propose a method to compute F in $\mathcal{R}$ by using Sylvester Matrices (see [27] for more details).

There is a natural bijection: $M^{m\times n}(K[z]) \leftrightarrow M^{m\times n}(K)[z]$, where $K[z]$ is the polynomial ring $\mathcal{R}$ and $M^{m\times n}(K)$ is the matrix with entries in the field K. Hence, equivalently we can consider a polynomial matrix as a polynomial with matrix coefficients, a so-called *matrix polynomial*.

Let $T(d) = \binom{s+d}{d}$ (for notational simplification the parameter s, which is the number of variables in the polynomial ring, is omitted). The polynomial matrix A can be written in terms of increasing total degree order of monomials of z: $A(z) = \sum_{i=1}^{T(d_1)} A_i z^{\alpha_i}$. Here d_1 is the maximum total degree of the entries of A and $T(d_1)$ is maximum number of terms of $A(z)$. Assume $f \in N$ has degree d_2. Similarly we have $f(z) = \sum_{j=1}^{T(d_2)} f_j z^{\beta_j}$. Hence

$$A(z)f(z) = \sum_{k=1}^{T(d_1+d_2)} C_k z^{\gamma_k} = 0^{m\times 1} \tag{3.3.2}$$

where $C_k := \sum_{\alpha_i+\beta_j=\gamma_k} A_i f_j$. This equation is equivalent to each coefficient $C_k = 0$.

Naturally, we write the coefficients of $f(z)$ as a vector: $v_f := [f_1, ..., f_{T(d_2)}]^t$. It is not hard to find a matrix M_A whose entries are the coefficients of $A(z)$, such that

$$M_A^{mT(d_1+d_2)\times nT(d_2)} \cdot v_f^{nT(d_2)\times 1} = 0^{mT(d_1+d_2)\times 1}\ . \tag{3.3.3}$$

We call M_A the *Sylvester Matrix*. We make the relations above clear by a diagram:

$$\begin{aligned} f \xrightarrow{\phi} f(z) \xrightarrow{\psi} v_f, \qquad & f \xrightarrow{\omega} v_f \\ A \xrightarrow{\phi} A(z) \xrightarrow{\psi} M_A, \qquad & A \xrightarrow{\omega} M_A \end{aligned} \tag{3.3.4}$$

where ϕ, ψ are bijections and $\omega = \psi \circ \phi$.

We can use the SVD to compute the null-space of the Sylvester matrix M_A, denoted by N_A, then construct v_f and f from N_A. If f_i is in the null-space of A, then

v_{f_i} must be in N_A. Note that $\dim N_A$ can be larger than r. First we choose lowest degree columns from N_A which are linearly independent vectors over the polynomial ring, denoted by F. Second we ascend from lower degree to higher degree columns to check the linear independency (using rank estimation). If a column is linearly independent it is included in F. Finally we obtain an updated F with rank r, which is a basis.

The remaining issue is the estimation of a degree bound for a null-space basis to guarantee the termination of the algorithm. Henrion [6] gave a bound for such bases. Using the Laplace Theorem in [4] we also give a similar result which easily follows the standard linear algebra argument about the degree of the determinant of a polynomial matrix (see [27] for details).

Proposition 3.3.2. *Suppose $A \in M^{m\times n}(\mathcal{R})$ is a polynomial matrix. Suppose* $\mathrm{rank}(A) = k < n$, *$r = n - k$, and $deg(Col_i(A))$ is the maximum degree of all the elements in the i-th column of A. We can always change the order of columns to satisfy $deg(Col_1(A)) \geqslant deg(Col_2(A)) \geqslant \cdots \geqslant deg(Col_n(A))$. Then there exists G which is a basis of the null-space of A, such that*

$$degree(G) \leq d_A = \sum_{i=1}^{k} deg(Col_i(A)) \ . \tag{3.3.5}$$

If each $deg_c(A_i) = d$, then $d_1 = d$ and $d_2 = (n-1)d$. So the maximum size of M_A is $m\binom{s+nd}{s} \times n\binom{s+nd-d}{s}$.

3.4 Numerical Completion Methods

In this section we will present a numerical completion method based on polynomial matrix computation. In order to use generic points to ease our computation, we extend the coefficient field to $\mathbb{C}$. Note that the key step in completion of a PDE system is to determine whether R is equal to $\boldsymbol{\pi}\mathbf{D}R$ or not. The projection of a variety is not necessarily a variety. So we compute the *Zariski Closure* of the projection. But our method will fail to detect the singular cases of a PDE system when the Zariski closure has more points than the projection. Here we only consider the generic case and show that this problem can be reduced to rank computation.

To avoid any order dependence on the independent variables we propose a modified definition of leading linear part of PDE. An equation is *modified leading linear* (respectively, *modified leading nonlinear*) if it is linear (respectively, nonlinear) in the jet variables $\underset{q}{u}$, where q is the order of this equation (this (partial) ranking is: $\underset{0}{u} \prec \underset{1}{u} \prec \ldots \prec \underset{q}{u} \prec \ldots$).

The definition of modified leading linear and nonlinear PDE partitions R into two subsystems, the leading linear subsystem and the leading nonlinear subsystem

respectively. Then we compute the witness sets of the leading nonlinear subsystem by (diagonal) homotopy continuation methods [22, 18]. The leading linear subsystem will be processed by numerical differential elimination methods using witness sets.

3.4.1 Using Witness Points

Here we first use witness points to detect whether there are some new constraints in lower order jet space. If they exist, then we find them by numerical differential elimination methods introduced in the next section. The advantage of this strategy is that it can avoid useless elimination of the strategy in [18] whose cost is much higher than checking the existence of new constraints.

Theorem 3.4.1. *For any* $p \in W(R)$, $V(R) = \overline{\boldsymbol{\pi D} R}$ *if and only if* $\mathrm{rank}(\mathcal{S}_p) = \mathrm{rank}([\mathcal{S}_p, \boldsymbol{r}_p])$.

Proof: Suppose for any $p \in W(R)$, we have $\mathrm{rank}(\mathcal{S}_p) = \mathrm{rank}([\mathcal{S}_p, \mathbf{r}_p])$. At point p, there exists at least one solution u_p of $\mathcal{S}\cdot \underset{q+1}{u} + \mathbf{r} = 0$, so (p, u_p) must be in $V(\mathbf{D}R)$. Hence $p \in \overline{\boldsymbol{\pi}\mathbf{D}R}$. This is true for any generic point of R, so $V(R) \subseteq \overline{\boldsymbol{\pi}\mathbf{D}R}$. Consequently $V(R) = \overline{\boldsymbol{\pi}\mathbf{D}R}$.

Suppose $V(R) = \overline{\boldsymbol{\pi}\mathbf{D}R}$, then each $p \in W(R)$ must be in $\boldsymbol{\pi}\mathbf{D}R$ and $\boldsymbol{\pi}^{-1}p \in V(\mathbf{D}R)$. This means $\mathcal{S}\cdot \underset{q+1}{u} + \mathbf{r} = 0$ has at least one solution at point p, so $\mathrm{rank}(\mathcal{S}_p) = \mathrm{rank}([\mathcal{S}_p, \mathbf{r}_p])$.

3.4.2 Numerical Differential Elimination

Suppose there are some new constraints resulting from the leading linear equations of $\mathbf{D}R$ (3.3.1). Consider a polynomial vector f of order q, such that $f \cdot \mathcal{S} = 0$, then

$$f \cdot (\mathcal{S}\cdot \underset{q+1}{u} + \mathbf{r}) = f \cdot \mathbf{r} \tag{3.4.1}$$

which is a polynomial of order q. Obviously, this polynomial is also in the ideal generated by the leading linear part. To find all such polynomials in order to construct $\boldsymbol{\pi}\mathbf{D}R$, naturally leads us to consider the null-space of $\mathcal{S}^t$.

Theorem 3.4.2. *Let* $F := NullSpace(\mathcal{S}^t)$, $P := \boldsymbol{r}^t \cdot F$ *then*

1. *The inclusion* $\boldsymbol{\pi D} R \subseteq V(R) \cap V(P)$ *holds, and*
2. *For all* $p \in W(V(R) \cap V(P))$, $\mathrm{rank}(\mathcal{S}_p) = \mathrm{rank}([\mathcal{S}_p, \boldsymbol{r}_p])$ *implies* $\overline{\boldsymbol{\pi D} R} = V(R) \cap V(P)$.

Proof: (1) Because $F := NullSpace(\mathcal{S}^t)$ and $\mathcal{S}\cdot \underset{q+1}{u} + \mathbf{r} = 0$, $F^t \cdot (\mathcal{S}\cdot \underset{q+1}{u} + \mathbf{r}) = F^t \cdot \mathbf{r} = P^t = 0$. Hence $V(\mathbf{D}R) \subseteq V(P)$ and P only involves order q jet variables, so

$\boldsymbol{\pi}\mathbf{D}R \subseteq V(P)$. And $\boldsymbol{\pi}\mathbf{D}R \subseteq V(R)$, hence (1) is proved.
(2) We only need to prove $V(R) \cap V(P) \subseteq \overline{\boldsymbol{\pi}\mathbf{D}R}$. Because for any $p \in W(V(R) \cap V(P))$, $\text{rank}(\mathcal{S}_p) = \text{rank}([\mathcal{S}_p, \mathbf{r}_p])$. At point p, there exists at least one solution u_p of $\mathcal{S}\cdot \underset{q+1}{u} + \mathbf{r} = 0$, so (p, u_p) must be in $V(\mathbf{D}R)$. Hence $p \in \overline{\boldsymbol{\pi}\mathbf{D}R}$. This is true for any generic point of $V(R) \cap V(P)$, so (2) is true.

3.5 Simple Examples

Recall the simple illustrative system (3.2.2). At first differentiating R up to order 2 yields:

$$\begin{aligned} R^{(0)} = \{&u_x^2 + u_x - u = 0, && u_{yy} - u_{xy} = 0, \\ &2u_x u_{xx} + u_{xx} - u_x = 0, && 2u_x u_{xy} + u_{xy} - uy = 0\} \ . \end{aligned}$$

We can partition $R^{(0)}$ into a single leading nonlinear PDE $N^{(0)} = \{u_x^2 + u_x - u = 0\}$ and 3 leading linear PDE $L^{(0)}$:

$$\begin{pmatrix} 0 & (1+2u_x) & 0 \\ (1+2u_x) & 0 & 0 \\ 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} u_{xx} \\ u_{xy} \\ u_{yy} \end{pmatrix} = \begin{pmatrix} u_y \\ u_x \\ 0 \end{pmatrix} \ . \tag{3.5.1}$$

Applying ***WitnessSet*** [18] to $N^{(0)}$ yields a witness set $W^{(0)}$ with two approximate generic points in $V(N^{(0)})$. Applying rank test at the witness points of $W^{(0)}$ shows that there are no new constraints arising from projection. Since the symbol matrix has full rank, the algorithm has terminated.

Actually, for this example the second order jet variables, if desired, can be expressed in terms of lower order jet variables yielding the same answer as ***HybridRif*** [18] and the fully symbolic algorithm ***rifsimp*** [16]. However our goal is to obtain an involutive form rather than put the system into triangular solved form. The advantage is that we can avoid computing the inverse of a symbolic matrix which in some cases yields an unmanageably large polynomial matrix.

EXAMPLE 3.5.1 (Use of All Witness Points)**.** *The input system is* $\langle u_t, v_t - u(u-1), u(v-1)\rangle$. *First we prolong* $u(v-1)$ *once and obtain* $D_t(u(v-1)) = (v-1)u_t + uv_t$. *We write the system in matrix form as:*

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ (v-1) & u \end{pmatrix} \begin{pmatrix} u_t \\ v_t \end{pmatrix} = \begin{pmatrix} 0 \\ u(u-1) \\ 0 \end{pmatrix} \tag{3.5.2}$$

with the constraint $u(v-1) = 0$. *The witness set contains two points:* $(0, \tilde{v})$ *and* $(\tilde{u}, 1)$, *where* $\tilde{u}, \tilde{v}$ *are some random complex floating point numbers. At* $(0, \tilde{v})$, *the rank of symbol matrix is equal to the rank of the augmented matrix which indicates*

that there are no new constraints in this case. At $(\tilde{u}, 1)$*, there exists a new constraint, since the ranks are not equal. We construct the projected polynomial by computing the null-space of the symbol matrix, which is* $(1-v, -u, 1)$*. So the new constraint is* $(1-v, -u, 1) \cdot (0, u(u-1), 0)^t = -u^2(u-1)$*. Appending the prolongation of the new equation* $((3u^2 - 2u)u_t)$ *to the system, we obtain a new system in matrix form:*

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ (v-1) & u \\ (3u^2-2u) & 0 \end{pmatrix} \begin{pmatrix} u_t \\ v_t \end{pmatrix} = \begin{pmatrix} 0 \\ u(u-1) \\ 0 \\ 0 \end{pmatrix} \tag{3.5.3}$$

with constraints $\{u(v-1) = 0, u^2(u-1) = 0\}$*. This implies two cases:* $u = 0$ *which was found before and* $(u, v) = (1, 1)$*. In this case the rank test shows that there are no new constraints. Hence our algorithm terminates.*

3.6 Physical Example

Systems such as the DAE below often arise in applications. Such systems of higher index can become very challenging for symbolic differential elimination algorithms such as **rifsimp**. Such algorithms attempt to triangularize the systems, and expression swell, from the inversion of densely filled symbolic matrices, can follow. We briefly mention that the size of these matrices below can be sharply reduced when a strategy is applied to detect constant full rank sub-matrices and reduce the number of variables by elimination.

EXAMPLE 3.6.1 (Distillation Stages [25])**.** *Consider the square* DAE *system:*

$$\begin{aligned} z_t^1 - f_1(z^1, u, t) &= 0, \quad z_t^2 - f_2(z^1, z^2, t) = 0, \\ z_t^3 - f_3(z^2, z^3, t) &= 0, \quad z_t^4 - f_4(z^3, z^4, t) = 0, \\ z^4 - out(t) &= 0 \end{aligned} \tag{3.6.1}$$

The unknown functions $\{f_1, f_2, f_3, f_4, out\}$ *are replaced with random polynomials with degree* 2*. The system is prolonged to order* 1 *to obtain* 5 *equations in* J^1 *and one equation in* J^0*. These* 5 *equations are written in matrix form and the rank test shows there are new constraints. We construct them by null-space computation. In the next iteration, the new equations are prolonged to order* 1 *and the matrix updated and so on. After* 5 *iterations, our algorithm stops and finds* 5 *constraints in* J^0*. There are* 11 *equations in* J^1*. The singular values of the symbol matrix are* $[158.7, 65.1, 54.1, 25.9, .316]$*. So it has full rank. The largest matrix processed in this example is* 1120×210*. We also applied* ***rifsimp*** *to this problem using Maple 10, on a* 1.5 *GHZ Pentium M, with* 512 *MB of RAM, running under Windows XP. After* 2 *hours the computation exhausted RAM and failed. Since the symbol matrix has*

an identity sub-matrix, a more efficient alternative way is to reduce the size of the Sylvester matrix by solving the corresponding sub-system first.

3.7 Random PDE Examples

In this section we use random systems of PDE to illustrate the methods developed in this paper. By their generic form, one would expect integrability conditions to impose new algebraic conditions in Jet space, cascading until such systems became algebraically inconsistent. However, we have:

Theorem 3.7.1. *Consider a system of s random* PDE*: $\{R^1, ..., R^s\}$ in $\mathbb{C}[x, u, ..., \underset{q}{u}]$ with s dependent variables u^1, u^2, ... , u^s and r independent variables $x_1, ..., x_r$ where each* PDE *has order q. Then with probability 1 the system is involutive.*

Outline of Proof: The proof follows directly from the definitions in the Cartan-Kuranishi approach.

Consider the s so-called highest class order q jet variables w corresponding to $\left(\frac{\partial}{\partial x_r}\right)^q u^k$ and denote the remaining order q jet variables by z (see [15, 20] for the definition of the class of a jet variable). Then $\mathcal{S}R = \left(\frac{\partial R}{\partial w} \mid \frac{\partial R}{\partial z} \right)$ and randomness implies that $\det\left(\frac{\partial R}{\partial w}\right) \neq 0$ and $\operatorname{rank}\left(\frac{\partial R}{\partial w}\right) = s$ on $V(R)$ with probability 1.

Then by the definition of class of a jet variable $\beta_r^{(q)} = s$, $\beta_{r-1}^{(q)} = \cdots = \beta_1^{(q)} = 0$. In addition it easily follows from $\det\left(\frac{\partial R}{\partial w}\right) \neq 0$ that $\operatorname{rank}(\mathcal{S}\mathbf{D}R) = rs$. As a consequence (3.2.5) is satisfied and $\operatorname{rank}(\mathcal{S}\mathbf{D}R) = rs = \sum_1^r k\beta_k^{(q)}$. Thus the symbol of the system is involutive. Then $\mathbf{D}R$ is easily seen to be of maximal rank, and hence there are no projected conditions and the system is involutive.

EXAMPLE 3.7.1 (Random Square PDE)**.** *We generate a* PDE *system R' randomly as follows. First generate two random polynomial* PDE *with degree 2:*

$$R = \{R^1(u_x, u_y, v_x, v_y, u, v), R^2(u_x, u_y, v_x, v_y, u, v)\}$$

Note that R is involutive by Theorem 3.7.1. This implies the prolongation $\boldsymbol{D}R$ is also involutive. Then we obtain our test system R' (6 equations with order 2) using random linear combination of $\boldsymbol{D}R$. Since R' has the same variety as $\boldsymbol{D}R$ it is also an involutive system (in disguise). We show that our method can determine the involutivity of R'.

First we verify $\boldsymbol{\pi}\boldsymbol{D}R' = R'$, which requires tracing 2^6 homotopy paths to compute the witness set of $V(R')$ (if the degree is 5, this number will be 15625). Applying the rank test at generic points in J^2 space shows there are no new constraints. The test (3.2.5) shows that the symbol is involutive since $\sum_{k=1}^2 k\beta_k = 2 \times 2 + 1 \times 2 = 6$ and the rank of the symbol matrix of $\boldsymbol{D}R'$ is 6. This means R' is involutive.

Actually R' is leading linear, which motivates us to compute $\boldsymbol{\pi}R'$. Applying the rank test at generic points in J^1 space shows there are new constraints. We use our algorithm to construct the projected equations S^1, S^2 in J^1. They have degree 2, which means only 4 (when the degree is 5, it is 25) homotopy paths need to be traced and this is much more efficient. Let $H = \{R', \boldsymbol{D}(S^1), \boldsymbol{D}(S^2), S^1, S^2\}$. Similarly we can check that H is involutive. Using PHCpack [24] we verify $V(S^1, S^2) = V(R^1, R^2)$, which shows our algorithm finds the projected equations correctly.

*When symbolic methods such as **rifsimp** are applied to R', they can explode in memory as a result of trying to triangularize (or invert) complicated high degree polynomial matrices. Here **rifsimp** failed to terminate on the above systems with degree ≥ 2, while the method of this paper easily handled systems up to degree 5 in a few minutes of CPU time.*

3.8 Experiments with Approximate Ideal Membership Testing

It is natural to wonder how some sort of approximate ideal membership testing might be done with the output of symbolic-numeric methods. Simply following the same strategy of exact membership testing, reducing first to a Gröbner Basis, then finding a normal form of an expression h to test its ideal membership, will usually be unstable.

To test membership of an expression h in a differential ideal generated by R, instead of finding a normal form for R we use the tables of dimensions $\dim \boldsymbol{\pi}^\ell \mathbf{D}^k R$. If done exactly, when $\boldsymbol{\pi}^\ell \mathbf{D}^k R$ is involutive, this information encodes the differential Hilbert function of the differential ideal. See [20] for a discussion of the Hilbert function of involutive systems. If an expression is not in the differential ideal, then it must change the Hilbert function (a measure of the indeterminancy in the formal power series solutions of the system). Thus, in our approach, if applied exactly, we would first determine ℓ and k such that $\boldsymbol{\pi}^\ell \mathbf{D}^k R$ satisfies the involutive dimension criteria. Then, exact involution would be applied to the system R, h. If any of the dimensions determining the Hilbert function at involution change, then h is not in the differential ideal generated by R. We follow a similar strategy in the approximate case.

EXAMPLE 3.8.1 (Differential Ideal Membership)**.** *Consider the* ODE

$$y_{xx} + 5\,y_x - 6\,y^2 + 6\,y = 0\,. \tag{3.8.1}$$

The symmetry vector fields $\xi(x,y)\frac{\partial}{\partial x} + \eta(x,y)\frac{\partial}{\partial y}$ generating Lie symmetries leaving its solution set invariant have coefficients satisfying a linear homogeneous system of PDE *[13]. Most computer algebra systems have programs for automatically generating*

	$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$
$\ell=0$	8	8	6	4	3	2	2
$\ell=1$	6	8	6	4	3	2	2
$\ell=2$			6	4	3	2	2
$\ell=3$				4	3	2	2
$\ell=4$					3	2	2
$\ell=5$						$\boxed{2}$	2
$\ell=6$						2	2

Figure 3.8.1: Table of $\dim \boldsymbol{\pi}^\ell \mathbf{D}^k R$ for (3.8.2) with SVD tolerance 10^{-7}. The location of the passing of the involution test, is indicated by the box.

such systems. The symmetry defining system R associated with ODE *(3.8.1) is:*

$$\begin{aligned} \xi_{yy} = 0\,, \quad 10\xi_y - 2\xi_{xy} + \eta_{yy} = 0 & \qquad (3.8.2)\\ (6-12y)\eta + (6y^2-6y)(\eta_y - 2\xi_x) + 5\eta_x + \eta_{xx} &= 0\\ 5\xi_x + 18(y-y^2)\xi_y - \xi_{xx} + 2\eta_{xy} &= 0 \end{aligned}$$

Consider the problem of testing whether h lies in the differential ideal generated by (3.8.2) (i.e. check if h is a consequence of the differential ideal) where:

$$h := x(\eta_{xx} - \eta_x) + y(2y\xi_{xx} + \eta_x) + (x+2)(y^2\eta_{yy} - y\eta_y + \eta_{xy})$$

Reduction of R to a (linear) differential Gröbner Basis easily yields $\eta_x - \eta, \xi_x + \frac{1}{2y}\eta, \eta_y - \frac{1}{y}\eta, \xi_y$ in a ranking dominated by total order of derivative. Reducing h with respect to this basis yields 0, and so h lies in the differential ideal.

Instead of following this standard procedure, we first applied our symbolic-numeric projective involutive form method [2]. We observed that the system $\boldsymbol{\pi}^5\boldsymbol{D}^5 R$ approximately satisfies the dimension criteria for projective involution (see Figure 3.8.1). Next, a perturbation of order 10^{-9} was added to h to form $\widetilde{h}$. An SVD tolerance 10^{-7} was used to test approximate involution, but this time for the system $R, \widetilde{h}$. We found that the relevant dimensions at involution did not change. If these results were obtained exactly then $\widetilde{h}$ would be in the ideal generated by R. However since the computations are approximate they only offer some evidence that some nearby exact $\hat{R}$, $\hat{h}$ has $\hat{h}$ in the ideal generated by $\hat{R}$.

Suppose we have approximate $\widetilde{R}, \widetilde{h}$ where the Hilbert dimensions for $\widetilde{R}, \widetilde{h}$ at involution are the same as those for $\widetilde{R}$, using some reasonable tolerance. We then use refinement processes to attempt to construct nearby systems $\hat{R}$, $\hat{h}$ which exactly satisfy all of the dimension criteria for (exact) ideal membership.

EXAMPLE 3.8.2 (Polynomial Ideal Membership)**.** *Consider the system of polyno-*

mials in $\mathbb{Q}[x, y]$

$$\begin{aligned} p &= x^3 - y^3, \quad q = (x^2 + y + 1)(x - y), \\ f &= -5\,y^3x + 7\,x^2y^3 + xy^4 + 12\,y^4 - 8\,y^5 - 3\,y^2x - 7\,y^2x^2 \\ &\quad -12\,y^3 + 3\,x^2 + 7\,x^3 + 8\,x^2y - 4\,y^2 - 4\,x + yx + 4\,y \end{aligned} \tag{3.8.3}$$

It is easily exactly verified by Gröbner Basis computation that $\langle p, q\rangle$ *is positive dimensional and that* $f \in \langle p, q\rangle$.

To apply our approximate differential elimination methods, we exploit the well-known bijection between PDE *and polynomials where monomials in* x, y *are mapped to monomials in the differential operators* $\frac{\partial}{\partial x}$, $\frac{\partial}{\partial y}$.

We form $\tilde{p} = p + \delta p$, $\tilde{q} = q + \delta q$ *and* $\tilde{f} = f + \delta f$ *where the perturbations* δp, δq, δf *are randomly generated degree* 3 *dense polynomials with random coefficients of order* 10^{-9}.

We apply the approximate projective involution method to $\tilde{p}$, $\tilde{q}$, *with an SVD tolerance of* 10^{-7} *and obtain the results given in Figure 3.8.2. This gives some evidence of the possibility of a nearby projectively involutive system. To give stronger evidence, we actually now search for an exact such nearby system. We set our search space as the following symbolic class of polynomials in which* $\tilde{p}$, $\tilde{q}$ *is embedded (this is a step where there are often many choices):*

$$P(a) = \sum_{j+k=0}^{3} a_{j,k}\, x^j y^k, \quad Q(b) = \sum_{j+k=0}^{3} b_{j,k}\, x^j y^k\ . \tag{3.8.4}$$

So $\tilde{p} = P(a^{(0)})$, $\tilde{q} = Q(b^{(0)})$ *where* $a^{(0)}$, $b^{(0)}$ *is the list of* $10 + 10 = 20$ *coefficients defining* $\tilde{p}$, $\tilde{q}$.

Scott's STLS (Structured Total Least Squares) implementation in Maple of the method [11] is applied to $\tilde{p}$, $\tilde{q}$. *In* 2 *iterations, it converges to a nearby system,* $\{\hat{p} = P(a^{(0)} + \delta a), \hat{q} = Q(b^{(0)} + \delta b)\}$ *(ie.* δa *and* δb *were computed numerically). Now, with the obtained* $\hat{p}$ *and* $\hat{q}$, *the dimensions in the table in Figure 3.8.2 can be recovered with tolerances roughly equal to working precision.*

We apply the approximate projective involution method to $\hat{p}$, $\hat{q}$, $\tilde{f}$ *with an SVD tolerance of* 10^{-5} *and obtain the results given in Figure 3.8.2. This gives some evidence of the possible existence of a nearby projectively involutive system. The nearby system was chosen to consist of* $\hat{p}$, $\hat{q}$ *and* $F(c)$. *Here the forms of* $\hat{p}$, $\hat{q}$ *are fixed as* $\hat{p} = P(a^{(0)} + \delta a)$, $\hat{q} = Q(b^{(0)} + \delta b)$ *and* $F(c)$ *is a member of the class of polynomials:*

$$F(c) = \sum_{j+k=0}^{5} c_{j,k}\, x^j y^k\ . \tag{3.8.5}$$

So, $\tilde{f} = F(c^{(0)})$ *where* $c^{(0)}$ *is the initial list of its* 21 *defining coefficients, while the*

	$d=3$	$d=4$	$d=5$	$d=6$
	$k=0$	$k=1$	$k=2$	$k=3$
$\ell=0$	8	9	10	11
$\ell=1$	6	8	9	10
$\ell=2$	3	6	8	9
$\ell=3$	1	3	6	8

Figure 3.8.2: Table of $\dim \boldsymbol{\pi}^\ell \mathbf{D}^k R$ for R, which is of degree $d=3$, given by $\tilde{p}$, $\tilde{q}$ in (3.8.3) SVD tolerance 10^{-7} (& also for $\hat{p},\hat{q}$ with tolerance 10^{-13}). The box gives the location of the passing of the involution test.

	$d=5$	$d=6$	$d=7$	$d=8$
	$k=0$	$k=1$	$k=2$	$k=3$
$\ell=0$	10	11	12	13
$\ell=1$	9	10	11	12
$\ell=2$	8	9	10	11
$\ell=3$	6	8	9	10
$\ell=4$	3	6	8	9
$\ell=5$	1	3	6	8

Figure 3.8.3: Table of $\dim \boldsymbol{\pi}^\ell \mathbf{D}^k R$ for R, which is of degree $d=5$, given by $\hat{p}$, $\hat{q}$, $\tilde{f}$ with tolerance 10^{-5} (& also for $\hat{p}$, $\hat{q}$, $\hat{f}$ with tolerance 10^{-13}).The box gives the location of the passing of the involution test.

20 *coefficients of $\hat{p}$, $\hat{q}$ will not be altered in the following refinement step.*

This time, instead of STLS, Scott's structured Newton's method in Maple is applied to $\hat{p}$, $\hat{q}$, $\tilde{f}$ and converges to a nearby system $\{\hat{p},\hat{q},\hat{f}=F(c^{(0)}+\delta c)\}$ in 1 iteration. This new system is exactly projectively involutive (to within working precision). Now, with tolerances about working precision, the dimensions of Figure 3.8.2 can be recovered.

With the exact systems $\{\hat{p},\hat{q}\}$ and $\{\hat{p},\hat{q},\hat{f}\}$ in mind, Figure 3.8.2 and 3.8.2 can be compared. Note that the pattern of dimensions is the same in both tables and implies that these two systems have the same Hilbert Function. Thus $\hat{f}\in\langle\hat{p},\hat{q}\rangle$.

3.9 Discussion

Our method applies to inexact systems of polynomially nonlinear PDE and relies on splitting the system into a leading linear subsystem and its complement. A new numerical differential elimination method based on polynomial matrix solving is applied to the leading linear part of the system. The success of this strategy enables the shrinking of the number of genuinely nonlinear equations that are dealt with by the numerical continuation methods.

A shortcoming of the new differential elimination method is that the size of

matrices we need to process can be very large (see Example 3.6). Let us consider a polynomial matrix $A \in M^{m\times n}(\mathcal{R})$, if each $deg(Col_i(A)) = d$ and rank of A is k, then $d_1 = d$ and $d_A = kd$. So the maximum size of M_A is $m\binom{s+d+kd}{s} \times n\binom{s+kd}{s}$. Assume $m \approx n$ and $kd \gg s$, the size of this matrix is bounded by $n(k+1)^s d^s$. We know $k < n$, so the bound is $n^{s+1}d^s$. When $s = 1$, a symbolic complexity result in [21] reports that the cost to compute the rank and null-space is the same as the cost of multiplication of matrices $\tilde{O}(n^{2.7}d)$, where $\tilde{O}$ indicates missing logarithmic factors $\alpha(\log n)^\beta(\log d)^\gamma$ for three positive real constants α, β, γ. Since the Sylvester matrix M_A is always sparse with block Toeplitz structure [28] gives a numerical algorithm with complexity $O(n^3 d)$ using block LQ factorization. However when $s > 1$, the block Toeplitz structure of M_A is much more complicated and further study is required.

In general, when the size, degree and number of unknowns of the symbol matrix are large, it is unrealistic to solve the corresponding matrix M_A. However, in many applications (e.g. multi-link pendula and Example 3.6) the symbol matrix has a very special structure, enabling the easy solution of subsystems. If we solve such sub-systems first, then the projected relations can be obtained directly without polynomial matrix solving. Hence our strategy is to find well-conditioned constant sub-matrices and substitute the corresponding solutions into the original system.

Geometric approaches have the advantage that they apply to both real ($\mathbb{F} = \mathbb{R}$) and and complex ($\mathbb{F} = \mathbb{C}$) smooth manifolds. One of our main tools, numerical algebraic geometry, depends on $\mathbb{F}$ being algebraically closed (so that a polynomial over $\mathbb{F}$ always has a root in $\mathbb{F}$). Indeed many of the main tools of (exact) algebraic geometry, although algorithmically powerful, suffer from the same restriction. To apply our approach to a real system, the PDE, the problem is first complexified, and the results for the real case checked heuristically on a case by case basis. However, progress in making numerical algebraic geometric techniques algorithmic for the real case is reported in [12].

Our experimental approach for testing approximate ideal membership differs radically from Gröbner type approaches, which utilize normal forms and reductions which are not numerically stable. In some sense, we are going back in history to Macaulay and Hilbert's initial studies. In particular we are framing ideal membership in terms of the dimensions that determine the Hilbert function of an ideal. Analogously, the new methods of Numerical Algebraic Geometry in some sense go back to a more primitive notion of geometry — that of a point on a variety.

This paper belongs to a series initiated in [26], continued in [17], [7] and [18] aimed at developing "Numerical Jet Geometry", based on "Numerical Algebraic Geometry".

Acknowledgements

We thank Robin Scott especially who contributed to the material in Section 3.8, and all of Example 3.8.2. We thank Lihong Zhi for discussions including valuable

comments about the STLS algorithm. We are indebted to the Referees for their comments, which helped to dramatically improve the paper.

Bibliography

[1] T. Arponen. *Numerical Solution and Structural Analysis of Differential-Algebraic Equations.* Ph.D. Thesis. Helsinki University of Technology, 2002.

[2] J. Bonasia, F. Lemaire, G. Reid, R. Scott, and L. Zhi. Determination of Approximate Symmetries of Differential Equations. *Centre de Recherches Mathématiques, CRM Proceedings and Lecture Notes.* Vol 39, pages 233–250, 2004.

[3] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Representation for the radical of a finitely generated differential ideal. Proc. ISSAC 1995. ACM Press. 158–166, 1995.

[4] W. C. Brown. Matrices Over Commutative Rings. Marcel Dekker, New York, 1992.

[5] M. Giusti and J. Heintz. La détermination de la dimension et des points isolées d'une variété algébrique peuvent s'effectuer en temps polynomial. In D. Eisenbud and L. Robbiano, eds., *Computational Algebraic Geometry and Commutative Algebra, Cortona 1991*, vol. XXXIV of *Symposia Mathematica*, pages 216–256. Camb. Univ. Press, 1993.

[6] D. Henrion. Reliable Algorithms for Polynomial Matrices, Ph. D. Thesis, Institute of Information Theory and Automation, Czech Academy of Sciences, Prague, Czech Republic, 1998.

[7] K. Hazaveh, D.J. Jeffrey, G.J. Reid, S.M. Watt, and A.D. Wittkopf. An exploration of homotopy solving in Maple. Proc. of the Sixth Asian Symp. on Comp. Math. (ASCM 2003). Lect. Note Series on Comput. by World Sci. Publ. 10 ed. by Z. Li and W. Sit (Singapore/River Edge, USA) 145–162, 2003.

[8] E. Hubert. Notes on triangular sets and triangulation-decomposition algorithms II: Differential Systems. *Symbolic and Numerical Scientific Computations*, Edited by U. Langer and F. Winkler. LNCS, volume 2630, Springer-Verlag Heidelberg, 2003.

[9] Kuranishi, M. On E. Cartan's prolongation theorem of exterior differential systems, Amer. J. Math., 79 1-47, 1957.

[10] G. Lecerf. Computing the equidimensional decomposition of an algebraic closed set by means of lifting fibers. *J. Complexity* 19(4):564–596, 2003.

[11] P. Lemmerling, N. Mastronardi, S. Van Huffel, Fast algorithm for solving the Hankel/Toeplitz Structured Total Least Squares Problem, Numerical Algorithms, vol. 23, 2000, pp. 371-392.

[12] Y. Lu, A.J. Sommese and C.W. Wampler. Finding all real solutions of polynomial systems: I The curve case, in preparation.

[13] P.J. Olver. Applications of Lie Groups to Differential Equations, Second Edition, Graduate Texts in Mathematics **107**, Springer-Verlag, New York, 1993.

[14] E. Mansfield. *Differential Gröbner Bases.* Ph.D. thesis, Univ. of Sydney, 1991.

[15] J.F. Pommaret. *Systems of Partial Differential Equations and Lie Pseudogroups.* Gordon and Breach Science Publishers, Inc. 1978.

[16] G.J. Reid, A.D. Wittkopf and A. Boulton. Reduction of systems of nonlinear partial differential equations to simplified involutive forms. *Eur. J. of Appl. Math.* 7: 604–635, 1996.

[17] G. Reid, C. Smith, and J. Verschelde. Geometric completion of differential systems using numeric-symbolic continuation. *SIGSAM Bulletin* 36(2):1–17, 2002.

[18] G. Reid, J. Verschelde, A.D. Wittkopf and Wenyuan Wu. Symbolic-Numeric Completion of Differential Systems by Homotopy Continuation. Proc. ISSAC 2005. ACM Press. 269–276, 2005.

[19] G.J. Reid, P. Lin, and A.D. Wittkopf. Differential elimination-completion algorithms for DAE and PDAE. *Studies in Applied Math.* 106(1): 1–45, 2001.

[20] W.M. Seiler. *Involution - The formal theory of differential equations and its applications in computer algebra and numerical analysis.* Habilitation Thesis, Univ. of Mannheim, 2002.

[21] A. Storjohann and G. Villard. Computing the rank and a small nullspace basis of a polynomial matrix. Research Report, volume 3, 2005.

[22] A.J. Sommese, J. Verschelde, and C.W. Wampler. Homotopies for intersecting solution components of polynomial systems. *SIAM J. Numer. Anal.* 42(4):1552–1571, 2004.

[23] A.J. Sommese and C.W. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science.* World Scientific Press, Singapore, 2005.

[24] J. Verschelde. Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. *ACM Transactions on Mathematical Software* 25(2): 251–276, 1999.

[25] J. Visconti. *Numerical Solution of Differential Algebraic Equations, Global Error Estimation and Symbolic Index Reduction.* Ph.D. Thesis. Laboratoire de Modélisation et Calcul. Grenoble. 1999.

[26] A. Wittkopf and G.J. Reid. Fast differential elimination in C: The CDiffElim environment. *Computer Physics Commun.*, 139: 192–217, 2001.

[27] Wenyuan Wu. Computing the Rank and Null-space of Polynomial Matrices, preprint.

[28] Juan C. Zuniga and Didier Henrion. Block Toeplitz Methods in Polynomial Matrix Computations. International Symposium on Mathematical Theory of Networks and Systems, 2004.

Chapter 4

Symbolic-numeric Computation of Implicit Riquier Bases for PDE

Riquier Bases for systems of analytic PDE are, loosely speaking, a differential analogue of Gröbner Bases for polynomial equations. They are determined in the exact case by applying a sequence of prolongations (differentiations) and eliminations to an input system of PDE.

We present a symbolic-numeric method to determine Riquier Bases in implicit form for systems which are dominated by pure derivatives in one of the independent variables and have the same number of PDE and unknowns.

The method is successful provided the prolongations with respect to the dominant independent variable have a block structure which is uncovered by Linear Programming and certain Jacobians are non-singular when evaluated at points on the zero sets defined by the functions of the PDE. For polynomially nonlinear PDE, homotopy continuation methods from Numerical Algebraic Geometry can be used to compute approximations of the points.

We give a differential algebraic interpretation of Pryce's method for ODE, which generalizes to the PDE case. A major aspect of the method's efficiency is that only prolongations with respect to a single (dominant) independent variable are made, possibly after a random change of coordinates. Potentially expensive and numerically unstable eliminations are not made. Examples are given to illustrate theoretical features of the method, including a curtain of Pendula and the control of a crane.

4.1 Introduction

Differential elimination algorithms apply a finite number of differentiations (prolongations) and eliminations to uncover obstructions to formal integrability. Exact differentiation elimination algorithms that apply to exact polynomially nonlinear systems of PDE are given in [2, 7, 14, 22, 17, 16]. Such methods enable the identification of all hidden constraints of PDE systems and the computation of initial data

and associated formal power series solutions in the neighborhood of a given point. Algorithmic membership tests (specifically in the radical of a differential ideal) can be given [2, 7]. They can ease the difficulty of numerical solution of ODE systems.

A major problem in these approaches is the exploding size of prolongations for more than one independent variable. In symbolic approaches much effort has been devoted to control the growth of this size by developing redundancy criteria (for integrability conditions), and making strong use of elimination with respect to rankings to decrease the size of the prolongations [1, 27]. However symbolic elimination can also cause expression swell, and even in the case of one independent variable, for constrained ODE problems arising in multi-body mechanics, it is a significant problem [26].

Very little work has been done on the corresponding problems for symbolic-numeric methods. Techniques which are helpful for the symbolic case are often unstable for the approximate case, since rankings (the differential analogue of term orders) can cause pivoting on small quantities and result in instability.

In this paper we make some progress on this problem for a certain class of PDE. For this class, only prolongations with respect to *one independent variable* are needed. Paradoxically rankings are important in our approach but don't cause instability since no eliminations are made. Hence we also avoid the expression swell due to the eliminations mentioned above. A suitable ranking is determined by solving an integer linear programming problem to uncover a block structure in the PDE system.

Another main idea in our paper is that such prolongations are essentially ODE like enabling us to generalize ODE techniques to the PDE case. In our case we generalize a method of Pryce for ODE in the framework of Riquier Theory. However we might imagine this being also used as a bridge for other ODE techniques (e.g. that of Sedoglavic [21]).

In particular, we give methods for computing approximate implicit Riquier Bases for square systems of analytic PDE.

There already exist exact methods for computing Riquier Bases for non-square polynomially nonlinear PDE together with an input ranking of derivatives [18]. However these exact methods may not succeed if the intermediate systems can not be solved explicitly for their leading derivatives.

For polynomially nonlinear PDE, our approximate Riquier Basis method uses an approximate method, homotopy continuation, to by-pass this difficulty. From a given set of solutions of a system of similar structure, homotopy paths converge to points on the zero set of the functions in the prolongations of the PDE system. It is these points that are used to verify the conditions of the Implicit Function Theorem, allowing the implicit solution of the given functions for their leading derivatives. For background on the homotopy methods, constituting the new area of Numerical Algebraic Geometry, please see the book [24].

In addition our method yields the method of Pryce [13] for systems of differential algebraic equations as a special case. Prolongation will usually introduce more equa-

tions as well as more (jet) variables, but this is not always true. If some equations after differentiation do not introduce new variables for whole system, then there is the possibility that the dimension of the system is lowered, since generically the system's dimension is the number of its variables minus the number of its equations. Pryce [13] proposed a method to detect such "chances" that minimize the dimension by taking advantage of the special structure of some systems. Pryce's method was the generalization of a method developed by Pantiledes. Ilie et al [6] show Pryce's method can be extended to give a polynomial cost method for numerical solution of differential algebraic equations.

4.2 Zero Set of PDE

Let $\mathbb{F}$ be a field ($\mathbb{R}$ or $\mathbb{C}$ in this paper), $x = (x_1, \cdots, x_n)$ be the independent variables and $u = (u^1, \cdots, u^m)$ be the dependent variables for a system of PDE. The usual commutative approaches to differential algebra and differential elimination theory [18, 2] consider a set of indeterminates $\Omega = \{u^i_\alpha \mid \alpha = (\alpha_1, \cdots, \alpha_n) \in \mathbb{N}^n, i = 1, \cdots, m\}$ where each member of Ω corresponds to a partial derivative by:

$$u^i_\alpha \leftrightarrow (\mathbf{D}_{x_n})^{\alpha_n} \cdots (\mathbf{D}_{x_1})^{\alpha_1} u^i(x_1, \cdots, x_n) := \mathbf{D}^\alpha u^i(x_1, \cdots, x_n) \; .$$

Formal commutative total derivative operators are introduced to act on members of Ω by a unit increment of the i-th index of their vector subscript: $\mathbf{D}_{x_i} u^k_\alpha := u^k_{\alpha+1_i}$ where $\alpha + 1_i = (\alpha_1, ..., \alpha_i + 1, ..., \alpha_n)$. The usual total derivatives $\mathbf{D}_{x_i}$ act on functions of $\{x\} \cup \Omega$ by:

$$\mathbf{D}_{x_i} = \frac{\partial}{\partial x_i} + \sum_{u \in \Omega} (\mathbf{D}_{x_i} u) \frac{\partial}{\partial u} \tag{4.2.1}$$

where $\frac{\partial}{\partial v}$ are the usual partial derivatives.

A q-th order differential system with ℓ equations is associated with a locus (or zero set) of points

$$Z(f) := \{(x, v^i_\alpha) \in J^q(\mathbb{F}^n, \mathbb{F}^m) : f^k(x, v^i_\alpha) = 0, k = 1, ..., \ell\} \tag{4.2.2}$$

where $J^q(\mathbb{F}^n, \mathbb{F}^m) \simeq \mathbb{F}^n \times \mathbb{F}^m \times \mathbb{F}^{m_1} \times ... \times \mathbb{F}^{m_q}$ is the jet space of order q and $f^k : J^q(\mathbb{F}^n, \mathbb{F}^m) \to \mathbb{F}$, $k = 1, ..., \ell$ are the maps defining the differential equations. Here $m_r := m \cdot \binom{r+n-1}{r}$ is the number of jet variables corresponding to r-th order derivatives.

One class of systems considered in this paper will be differential polynomials in $\mathbb{F}[x_1, ..., x_n; v^i_\alpha : |\alpha| \geq 0]$, the ring of all polynomials over $\mathbb{F}$ in the infinite set of indeterminates $\{x\} \cup \Omega$, where $|\alpha| = \alpha_1 + \cdots \alpha_n$. The other case is that where the f^k are $\mathbb{F}$-analytic functions in a neighborhood of a point $(x^0, (v^i_\alpha)^0)$, which by our finiteness restriction can be taken in J^q. We restrict to f^k being functions of finitely

many indeterminates. We alert the reader that although we occasionally use Jet notation, we always work locally over some $\mathbb{F}$-Euclidian space. So we don't use the more global geometric features of Jet Geometry, such as bundles, contact structures, etc (see [22]).

The simple pendulum gives an example of a constrained set of differential equation (commonly called differential algebraic equations or DAE) that arise frequently in applications. As a matter of terminology, throughout this paper we will use the term ODE to include DAE. Such systems are ubiquitous in multi-body dynamics. From CAD-like graphical descriptions of links, joints, motors, etc, there are several software packages (e.g. Adams, Dads and WorkingModel [23]), that automatically produce the equations of motion, using Lagrangian mechanics formulations.

EXAMPLE 4.2.1 (The Pendulum)**.** *For the pendulum of unit mass, under constant gravity, we have*

$$\begin{aligned} X_{tt} + \lambda X &= 0, \\ Y_{tt} + \lambda Y &= -g, \\ X^2 + Y^2 &= 1. \end{aligned} \tag{4.2.3}$$

Here

$$\begin{aligned} Z(f) = \{(t, X, Y, \lambda, X_t, Y_t, \lambda_t, X_{tt}, Y_{tt}, \lambda_{tt}) \in J^2 : \\ X_{tt} + \lambda X = 0, Y_{tt} + \lambda Y + g = 0, X^2 + Y^2 - 1 = 0\} \end{aligned}$$

is a 7 dimensional submanifold of $\mathbb{F}^{10} \simeq J^2$.

4.3 Rankings of Derivatives

A detailed formal treatment of this subject, and the classification of all such rankings are given in Rust et al. [18]. Rankings are fundamental in *Differential Algebra* [8].

Definition 4.3.1 (Ranking [18])**.** *A positive ranking* $\prec$ *of* Ω *is a total ordering on* Ω *which satisfies:*

$$v_\alpha^i \prec v_\beta^j \Rightarrow v_{\alpha+\gamma}^i \prec v_{\beta+\gamma}^j, \tag{4.3.1}$$

$$v_\alpha^i \prec v_{\alpha+\gamma}^i, \tag{4.3.2}$$

for all $\alpha, \beta, \gamma \in \mathbb{N}^n$.

Let HDf denote the greatest member in Ω in f with respect the ranking $\prec$.

EXAMPLE 4.3.1. *An example of a ranking for the Pendulum system given in Example 4.2.1 is:*

$$X \prec Y \prec \lambda \prec X_t \prec Y_t \prec \lambda_t \prec X_{tt} \prec Y_{tt} \prec \lambda_{tt} \prec \cdots \tag{4.3.3}$$

It is easily seen that (4.3.3) is invariant under differentiation, so (4.3.1) is satisfied. In addition any derivative of a member is greater than itself, so (4.3.2) is satisfied. In this ranking $\text{HD}(X_{tt}+\lambda X)=X_{tt}$, $\text{HD}(Y_{tt}+\lambda Y-g)=Y_{tt}$, *and* $\text{HD}(X^2+Y^2-1)=Y$.

There are many ways to specify a ranking. In this paper we use a matrix representation following Riquier and Rust [18, 19]. First we introduce a map ψ from Ω to $\mathbb{Z}^{m+n}$:

$$\psi : \frac{\partial^{\alpha_1+\cdots+\alpha_n} u^j}{\partial x_1^{\alpha_1}\cdots\partial x_n^{\alpha_n}} \mapsto (0,\cdots,0,1,0,\cdots,0,\alpha_1,\cdots,\alpha_n)^t \tag{4.3.4}$$

where the "1" appears in the jth coordinate.

An ordering of the elements in $\mathbb{Z}^{m+n}$ denoted by $<$ is defined by lexical order (comparing the values at the first coordinate, then the second coordinate, and so on).

Definition 4.3.2 (Ranking by Matrix)**.** *Suppose M is an $\ell\times(m+n)$ matrix with nonnegative integer entries and satisfies: $\theta\neq\tau\Rightarrow M\cdot\psi(\theta)\neq M\cdot\psi(\tau)$. We define $\prec_M$ to be a ranking with respect to M, if for $\theta,\tau\in\Omega$, we have $\theta\prec_M\tau\Leftrightarrow M\cdot\psi(\theta)<M\cdot\psi(\tau)$. Here M called a matrix representation of this ranking. And $\theta\preceq_M\tau$ means $\theta\prec_M\tau$ or $\theta=\tau$.*

4.4 Signature Matrix of t-Dominated Systems using Rankings

The methods developed in this paper are applicable to a class of PDE that are *dominated by pure derivatives* in one of their independent variables.

Examples of such PDE include those of Cauchy-Kovaleskya type such as hyperbolic equations (e.g. the wave equation $u_{tt}=c^2u_{xx}$). Equations of parabolic type, such as the classical Heat equation $u_t=u_{xx}$ are also included. In these cases the dominating variable is the time t. PDE of elliptic type are included in this class, such as the Cauchy-Riemann equations: $\{u_x=v_y, v_x=-u_y\}$.

Our main illustrative example is:

EXAMPLE 4.4.1 (Pendulum Curtain)**.** *Consider a curtain made of many pendula hanging under gravity g as shown in Figure 4.4.1. The Pendula are restricted to move on the surface of the cylinder and in planes perpendicular to the s-axis displayed in Figure 4.4.1. The pendula form a continuous curtain in the limit. For small deviations from the vertical equilibrium position the equations for $X(t,s)$, $Y(t,s)$ and Lagrange multiplier $\lambda(t,s)$ for the continuous curtain are:*

$$\begin{aligned} X_{tt}+\lambda X &= \kappa X_{ss} \\ Y_{tt}+\lambda Y+g &= \kappa Y_{ss} \\ \frac{1}{2}(X^2+Y^2-1) &= 0 \end{aligned} \tag{4.4.1}$$

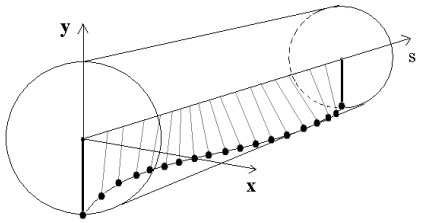

Figure 4.4.1: Pendulum Curtain

Note that when $\kappa = 0$ this reduces to the simple pendulum equations given in Example 4.2.1.

By a *pure derivative* with respect to an independent variable x_i, we mean a derivative of form $\left(\frac{\partial}{\partial x_i}\right)^k u^j$ where $k \in \mathbb{N}$. By Definition 4.4.1 given later, a PDE system which is *dominated by pure derivatives* with respect to an independent variable x_i, must at least contain such a derivative in each of its equations. The Pendulum Curtain system (4.4.1) satisfies this requirement with respect to t. A physically important class of PDE which are dominated by pure derivatives in time consists of *evolution* PDE. In that case the time derivatives can be expressed as functions of spatial derivatives.

For two independent variables t, x and for each u^j, a ranking consistent with such systems should satisfy:

$$u^j \prec u^j_x \prec u^j_{xx} \prec \cdots \prec u^j_t \prec u^j_{tx} \prec \cdots \tag{4.4.2}$$

It is easy to extend this (partial) ranking to the case when x is a vector (e.g. using lexical order on x).

For the pendulum curtain example, the differential order of t is more important than spatial derivatives. More generally, we can focus on a special independent variable x_k and for notational convenience denote x_k by t. However we warn the reader that t may not represent time for some physical t-dominated systems. For example the elliptic Cauchy-Riemann equations ($u_x = v_y, v_x = -u_y$) are x-dominated.

We hide the details about the differential order of the other independent variables by defining a weight map $\varphi : \Omega \to \mathbb{R}$ as follows:

$$\varphi(v^i_\alpha) := \begin{cases} \alpha_k, & \text{if } \alpha_p = 0, \text{ for any p} \neq \text{k} ; \\ \alpha_k + \epsilon, & \text{otherwise.} \end{cases} \tag{4.4.3}$$

The leading derivative of each equation R_i with respect to each u^j using the ranking (4.4.2), is denoted by $\text{LD}(R_i, u^j)$. Applying (4.4.3) to the leading derivatives of R, we obtain an $\ell \times m$ matrix $(\sigma_{i,j})$ which is called the *signature matrix* of R (see

Pryce [13] for the ODE case):

$$(\sigma_{i,j})(R) := \begin{cases} \varphi(\text{LD}(R_i, u^j)), & \text{if } R_i \text{ depends on } u^j \ ; \\ -\infty, & \text{otherwise .} \end{cases} \tag{4.4.4}$$

It is easy to show that $(\sigma_{i,j})(\mathbf{D}_t R) = (\sigma_{i,j})(R) + 1^{\ell\times m}$, where $1^{\ell\times m}$ is a matrix with all entries equal to 1.

We define the *leading class* of derivatives by

$$\text{LCD}(R) := \{\text{LD}(R, u^j) : 1 \le j \le m\} \ .$$

These are the highest derivatives of u^j appearing in R.

If for each equation of R, the leading class of derivatives are pure t-derivatives, then regarding the other independent variables as parameters the PDE has an ODE-like structure. Then we can consider the other independent variables as parameters to regard the PDE as an ODE. To study the PDE with this structure, we introduce a new concept:

Definition 4.4.1. *We say R is dominated by pure derivatives in the independent variable t if there is no ϵ appearing in $(\sigma_{i,j})(R)$. For notational simplicity, we also call R a t-dominated system.*

Such t-dominated systems are not as special as they appear.

Proposition 4.4.1. *[Genericity of t-dominated Systems]*
A generic $\mathbb{F}$-analytic or polynomially nonlinear PDE *system R with order k is t-dominated. Any $\mathbb{F}$-analytic or polynomially nonlinear* PDE *system R with order k is t-dominated after a random linear coordinates transformation in the independent variables with coefficients in $\mathbb{F}$.*

PROOF. Let R be a generic PDE. So each R_i contains all pure t derivatives with order k, which are the leading class of derivatives with respect to Ranking (4.4.2). For any nonlinear PDE R, after a random linear coordinate change, any derivative with order k becomes a linear combination of all the kth order derivatives. So R contains all pure kth order t derivatives which are the leading class of derivatives. □

Remark 4.4.1. *A symbolic random linear coordinates transformation often destroys the sparsity of the original system, which causes a dramatic increase in size of the system if subsequent eliminations are applied. However our numeric transformation in fixed precision lessens expression growth. Also, as we will show, no eliminations are involved in our method.*

4.5 Generalizing Pryce's Prolongation Method to PDE

Let R be a square (i.e. #equations = #unknowns) and t-dominated system. From Section 4.4, the signature matrix $(\sigma_{i,j})(R)$ contains information on differential order and ignores details on the degrees and coefficients of a system R. We introduce a fast method based on $(\sigma_{i,j})(R)$ to differentiate (prolong) R with respect to t. Pryce's prolongation method for square ODE is a special case. It yields a local existence and uniqueness result (equivalently all local constraints in initial conditions for R are determined). We obtain a local existence and uniqueness result for square PDE which is given in Section 4.6.

If we consider R as ODE (the only independent variable is t) then Pryce's method [13] finds all the local constraints for a large class of square ODE using only prolongation. We generalize this construction to PDE. To be precise, the description of this construction is given in this section, but the detailed justification of its properties under certain conditions is given in Section 4.6. Suppose R_i is differentiated c_i times ($c_i \geq 0$). The new system after differentiation is denoted by $\mathbf{D}_t^c R$. Suppose the highest order of u^j appear in $\mathbf{D}_t^c R$ is d_j. From the definition of $(\sigma_{i,j})$, clearly d_j is the largest of $c_i + \sigma_{ij}$, which implies that

$$d_j - c_i \geq \sigma_{ij}, \quad \text{for all } i, j \tag{4.5.1}$$

Obviously there are at most $\sum d_j + m$ pure t-derivative jet variables and $\sum c_i + m$ equations in $\mathbf{D}_t^c R$ (considering independent variables and all non-t-derivatives as parameters). We can embed $\mathbf{D}_t^c R$ into a $\sum d_j + m$ dimensional space. If each equation drops the dimension of the zero set of $\mathbf{D}_t^c R$ by one, then the dimension of $\mathbf{D}_t^c R$ is $\sum d_j - \sum c_i$. Roughly speaking, to find all the constraints is equivalent to minimizing the dimension of $\mathbf{D}_t^c R$. This can be formulated as an integer linear programming problem in the variables $c = (c_1, ..., c_m)$ and $d = (d_1, ..., d_m)$:

$$\begin{cases} \text{Minimize} & z = \sum d_j - \sum c_i, \\ \text{where} & d_j - c_i \geq \sigma_{ij}, \\ & c_i \geq 0 \end{cases} \tag{4.5.2}$$

The computation of c and d which only involves the information on differential order and is consequently very fast.

Remark 4.5.1. *However, this linear programming problem may not have a solution. If we consider its dual problem in the sense of linear programming, which is an Assignment Problem, the task is to choose just one element in each row and column of the signature matrix, then maximize the sum of these m elements. The maximum is called the Maximal Transversal Value. If this value exists, then (4.5.2) has a finite solution. Like Pryce's method [13], we always assume that the maximal transversal*

B_0	B_1	$\cdots$	B_{k_c-1}	B_{k_c}
$R_1^{(0)}$	$R_1^{(1)}$	$\cdots$	$R_1^{(c_1-1)}$	$R_1^{(c_1)}$
	$R_2^{(0)}$	$\cdots$	$R_2^{(c_2-1)}$	$R_2^{(c_2)}$
		$\vdots$	$\vdots$	$\vdots$
		$R_m^{(0)}$	$\cdots$	$R_m^{(c_m)}$

Table 4.1: The triangular block structure of $\mathbf{D}_t^c R$. For $0 \le i < k_c$, B_i has fewer jet variables than B_{i+1}.

value exists in this paper.

EXAMPLE 4.5.1. *Note that Example (4.4.1) is t-dominated (and also s-dominated). Thus we can apply the method above for* PDE. *The signature matrix for the above system with columns corresponding to X, Y and λ from left to right is:*

$$(\sigma_{i,j}) = \begin{pmatrix} 2 & -\infty & 0 \\ -\infty & 2 & 0 \\ 0 & 0 & -\infty \end{pmatrix}$$

Recall that c_i means the i-th equation needs to be differentiated c_i times ($c_i \ge 0$) and d_j is the highest order of u^j after the prolongation. Then (4.5.2) is

$$\begin{cases} \textit{Minimize} & z = d_1 + d_2 + d_3 - c_1 - c_2 - c_3, \\ \textit{where} & d_1 - c_1 \ge 2, \quad d_1 - c_2 \ge -\infty, \quad d_1 - c_3 \ge 0, \\ & d_2 - c_1 \ge -\infty, \quad d_2 - c_2 \ge 2, \quad d_2 - c_3 \ge 0, \\ & d_3 - c_1 \ge 0, \quad d_3 - c_2 \ge 0, \quad d_3 - c_3 \ge -\infty, \\ & c_1 \ge 0, \quad c_2 \ge 0, \quad c_3 \ge 0 \end{cases}$$

Solving this integer linear programming problem by LPSolve *in the Optimization package of Maple 10, we obtain*

$$c_1 = 0, \quad c_2 = 0, \quad c_3 = 2; \tag{4.5.3}$$

$$d_1 = 2, \quad d_2 = 2, \quad d_3 = 0. \tag{4.5.4}$$

After we obtain the number of prolongation steps c_i for each equation, we can construct the partial prolonged system $\mathbf{D}_t^c R$ using c.

We also point out that $\mathbf{D}_t^c R$ has a favorable block triangular structure which enables us to compute points on $Z(\mathbf{D}_t^c R)$ more efficiently. Without loss of generality, we assume $c_1 \ge c_2 \ge \cdots \ge c_m$, and let $k_c = c_1$, which is closely related to the *index* of system R (see [13] for more details about the index). Then we can partition $\mathbf{D}_t^c R$ into $k_c + 1$ parts (see Table 4.1).

For each $B_i, 0 \le i \le k_c$, we denote the leading class of variables by $U_i := \text{LCD}(B_i)$

and define the Jacobian Matrix

$$\mathcal{J}_i := \left(\frac{\partial B_i}{\partial U_i}\right). \tag{4.5.5}$$

Proposition 4.5.1. *Let $\mathcal{J}(\boldsymbol{D}_t^c R) := \{\mathcal{J}_i\}$ be the set of Jacobian matrices of $\{B_i\}$. For any $0 \leq i < j \leq k_c$, $\mathcal{J}_i$ is a sub-matrix of $\mathcal{J}_j$. Moreover, if $\mathcal{J}_{k_c}$ has full rank, then any $\mathcal{J}_i$ also has full rank.*

PROOF. The first result is by the chain rule and the fact that if θ is the leading variable of a PDE F then θ_t is the leading variable of $\mathbf{D}_t F$.

Because $\mathcal{J}_{k_c}$ is an $m \times m$ full rank, each row is linearly independent to the others. Since $\mathcal{J}_i$ is a sub-matrix of $\mathcal{J}_{k_c}$, we can assume it consists of the first p rows and first q columns of $\mathcal{J}_{k_c}$, where q is the number of elements in U_i. If $q = m$, then $\operatorname{rank}(\mathcal{J}_i) = p$. If $q < m$, then the entries in first p rows and last $m - q$ columns must be 0. So $\operatorname{rank}(\mathcal{J}_i) = p$. □

In the following section we will show that the output of the t-prolongation implicitly yields a Riquier Basis for which an associated existence theorem is available.

4.6 The Formal Riquier Existence Theorem

In this section, we state Theorem 4.6.1 for the existence and uniqueness of formal power series solutions of a Riquier Basis. This theorem is the result of a Gröbner style development and extension of Riquier's classical existence results for PDE. The details can be found in the works of Rust et al. [18, 19]. The corresponding exact symbolic differential elimination algorithms were implemented [27] in distributed Maple. Reference [27] also discusses applications of the algorithms.

Given a ranking of partial derivatives, such bases are in solved form with respect to leading derivatives. They are symbolically determined by successively including integrability conditions and performing eliminations on the resulting systems. The solved form requirement means that in the exact case they are essentially restricted to PDE which are linear in their highest derivatives. Closely related to Riquier Bases are Schwarz's Janet Bases [20].

We say that f is *$\prec$-monic with respect to a ranking $\prec$* if f has the form $f = \mathrm{HD} f + g$, with $\mathrm{HD} g \prec \mathrm{HD} f$. For example the equation $X^2 + Y^2 - 1 = 0$ of the Pendulum system of (4.2.3) is not $\prec$-monic with respect to the ranking given in (4.3.3) since it is nonlinear in Y, its highest derivative.

Definition 4.6.1. *[$\mathcal{M}$, $\mathcal{V}$] In the remainder of the paper, fix a finite set $\mathcal{M}$ of $\prec$-monic functions of which are $\mathbb{F}$-analytic functions on some subset $\mathcal{V}$ of $J^r(\mathbb{F}^n, \mathbb{F}^m)$ for some finite r. The subset is connected and open in the usual $\mathbb{F}$-Euclidean topology.*

Definition 4.6.2. *[Principal and Parametric Derivatives] The principal derivatives of $\mathcal{M}$ are defined as*

$$Prin\mathcal{M} := \{u \in \Omega | \exists f \in \mathcal{M} \text{ and } \alpha \in \mathbb{N}^n \text{ with } u = \mathrm{HD}\boldsymbol{D}^\alpha f\}$$

The parametric derivatives of $\mathcal{M}$, which we denote $Par\mathcal{M}$, are those derivatives that are not principal.

The parametric and principal derivatives enable us to specify initial data, that will be important in the existence and uniqueness theorem.

Definition 4.6.3. *A specification of initial data for $\mathcal{M}$ is a map*

$$\phi : \{x\} \cup \ Par\ \mathcal{M} \to \mathbb{F}$$

For $x^0 \in \mathbb{F}^m$, we say that ϕ is a specification at x^0 if

$$\phi(x) := (\phi(x_1), \phi(x_2), \cdots, \phi(x_m)) = x^0.$$

For an analytic function g on jet space, let $\phi(g)$ be the function of the principal derivatives obtained from g by evaluating x and the parametric derivatives using ϕ:

$$\phi(g) := g(\phi(x), (\phi(u))_{u \in Par\mathcal{M}}).$$

Definition 4.6.4. *$\mathcal{M}$ is called a Riquier Basis if for all $\alpha, \alpha' \in \mathbb{N}^m$ and $f, f' \in \mathcal{M}$ with $\mathrm{HD}\boldsymbol{D}^\alpha f = \mathrm{HD}\boldsymbol{D}^{\alpha'} f'$, the integrability condition $\boldsymbol{D}^\alpha f - \boldsymbol{D}^{\alpha'} f'$ is reduced to zero by a sequence of one-step reductions by members of $\mathcal{M}$.*

See [19] for the definition of one-step reduction used above. Recall that $\mathcal{M}$ and $\mathcal{V}$ are as given in Definition 4.6.1.

Theorem 4.6.1 (Formal Riquier Existence Theorem)**.** *Let $\mathcal{M}$ be a Riquier Basis such that each $f \in \mathcal{M}$ is polynomial in the principal derivatives. For $x^0 \in \mathbb{F}^n$, let ϕ be a specification of initial data for $\mathcal{M}$ at x^0 such that $\phi(f)$ is well-defined for all $f \in \mathcal{M}$. Then there is formal power series solution $u(x) \in \mathbb{F}[[x - x^0]]^n$ to $\mathcal{M}$ at x^0 such that $\boldsymbol{D}^\alpha u^i(x^0) = \phi(u^i_\alpha)$ for all $u^i_\alpha \in Par\mathcal{M}$. Furthermore, every formal power series solution to $\mathcal{M}$ at x^0 may be obtained in this way for some ϕ.*

Note that the set of integrability conditions given by Definition 4.6.4 is generally infinite. This infinite number of conditions is shown in [18] to be a consequence of a finite set of integrability conditions given below; thus enabling finite implementation [27]. Further more refined redundancy criteria for integrability conditions are given in [27].

Definition 4.6.5. *Let $f, f' \in \mathcal{M}$ with $\mathrm{HD}f = \boldsymbol{D}^\alpha u^i$ and $\mathrm{HD}f' = \boldsymbol{D}^{\alpha'} u^{i'}$, and β be the least common multiple of α and α'. Then if $i = i'$, define the minimal integrability*

condition of f and f' to be $\text{IC}(f, f') = \boldsymbol{D}^{\beta-\alpha} f - \boldsymbol{D}^{\beta-\alpha'} f'$. *If $i \neq i'$, then* $\text{IC}(f, f')$ *is said to be undefined.*

See [19] for the definition of reduction used below.

Theorem 4.6.2. *Suppose that for each pair $f, f' \in \mathcal{M}$ with* $\text{IC}(f, f')$ *well-defined we have* $\text{IC}(f, f')$ *is reduced to* 0 *by a sequence of one-step reductions. Then $\mathcal{M}$ is a Riquier Basis.*

4.6.1 Implicit Riquier Existence Theorem

We know that for ODE if the Jacobian matrix is non-singular, Pryce's method can successfully construct the unique local solution at a given consistent initial point. Now let us consider the PDE case. We show that if $\mathcal{J}$ is non-singular at some point p, which satisfies system $\mathbf{D}_t^c R$, then any order derivative of each u^j is determined by p. So the Taylor series coefficients of the solution passing through p can be computed to arbitrary order.

For each dependent variable we have a ranking of type (4.4.2). To apply the Riquier Existence Theorem, we need to merge these partial rankings (4.4.2) to a total ranking which is consistent with all the partial rankings.

Proposition 4.6.1. *Let the leading class derivatives of R be $\{\theta_1, ..., \theta_m\}$ and let B be the set of all the other derivatives of R. Then there exists a positive ranking $\prec$ which satisfies the partial ranking (4.4.2) and $\theta_1 \succ \theta_2 \succ \cdots \succ \theta_m$ and each θ_i is greater than any $b \in B$.*

PROOF. Case 1: $m \geq n$. Suppose the dependent variable index of θ_i is i and $t = x_1$. If the dependent and independent variable indices do not satisfy this condition, then it can be satisfied after a permutation of the variables. Let $\begin{pmatrix} I^{m\times m} \\ X^{n\times m} \end{pmatrix} = (\psi(\theta_1), ..., \psi(\theta_m))$. Suppose c is the maximum entry of X. Then let $M' = c \cdot 1^{m\times m} - \begin{pmatrix} X \\ 0 \end{pmatrix}^{m\times m}$. Finally we construct an $(m+1)\times(m+n)$ matrix $M = \begin{pmatrix} M' & I^{n\times n} \\ \mathbf{v} & 0 \end{pmatrix}$, where $\mathbf{v} = (m, m-1, \cdots, 1)$. All the entries of M are non-negative. Suppose $\theta, \tau \in \Omega$ and $\theta \neq \tau$. One case is they have different dependent variables, then at least the last coordinates of $M \cdot \psi(\theta)$ and $M \cdot \psi(\tau)$ are different. The other case is that they have the same dependent variable. Then their ranks are determined by the last n columns of M, which is the lexical order over independent variables. In this case, $M \cdot \psi(\theta) \neq M \cdot \psi(\tau)$. So M is a matrix representation of a ranking which satisfies Ranking (4.4.2).

Suppose $i < j$, we can check $\theta_i \succ \theta_j$. Since $\begin{pmatrix} \gamma_i \\ m-i+1 \end{pmatrix} = M \cdot \psi(\theta_i) > M \cdot \psi(\theta_j) = \begin{pmatrix} \gamma_j \\ m-j+1 \end{pmatrix}$, where $\gamma_j = M'_j + \begin{pmatrix} X_j \\ 0 \end{pmatrix} = c \cdot 1^{m\times 1} = \gamma_i$.

Suppose $\tau \in B$ with dependent variable u^i, we can show $\theta_j \succ \tau$, for any j. Since $\prec_M$ satisfies Ranking (4.4.2),
$\begin{pmatrix} \gamma_\tau \\ m-i+1 \end{pmatrix} = M \cdot \psi(\tau) < M \cdot \psi(\theta_i) = \begin{pmatrix} \gamma_i \\ m-i+1 \end{pmatrix}$, which implies $\gamma_\tau < \gamma_i$. So $\gamma_\tau < \gamma_j = \gamma_i$, for any $1 \leq j \leq m$.

Therefore, $M \cdot \psi(\tau) < M \cdot \psi(\theta_j)$, which implies for any θ_j and any $\tau \in B$ we have $\tau \prec_M \theta_j$.

Case 2: $m < n$. In the proof, we only need to change the construction slightly by setting $M' = c \cdot 1^{n \times m} - X$. Similarly we construct an $(n+1) \times (m+n)$ matrix $M = \begin{pmatrix} M' & I^{n \times n} \\ \mathbf{v} & 0 \end{pmatrix}$. □

Lemma 4.6.3. *Let* $C = \begin{pmatrix} A^{n \times m} \\ B^{\ell \times m} \end{pmatrix}$ *and* $n + \ell \leq m$. *If* C *is a full rank matrix, then any rank* n *square sub-matrix of* A *can be extended to a rank* $n + \ell$ *square sub-matrix of* C.

Proof. Because C is a full rank matrix and $n + \ell \leq m$, $\operatorname{rank}(C) = n + \ell$. Suppose the first n columns of A form a full rank matrix, so the first n columns of C are linearly independent. A set of linearly independent columns can be extended to a basis of the column space of C. Hence we can find ℓ columns which generate a basis for the column space of C together with the first n columns. □

Lemma 4.6.4. *Let* R *be a square* $\mathbb{F}$*-analytic system of* PDE. *Suppose the maximal transversal value of* $(\sigma_{ij})(R)$ *exists. Let* $\boldsymbol{D}_t^c R$ *be the system obtained by the* t*-prolongation method of Section 4.5. If* $\mathcal{J}_{k_c}$ *is nonsingular at some point* p *in* $Z(\boldsymbol{D}_t^c R)$, *then there exists a positive ranking* $\prec$ *that determines a local solved form* $w^{(i)} = f^{(i)}(z)$ *for each block* B_i, *such that* $\boldsymbol{D}_t w^{(i-1)} \subseteq w^{(i)}$.

Proof. Because $\mathcal{J}_{k_c}$ is nonsingular at p, each $\mathcal{J}_i$ is full rank by Proposition 4.5.1. So B_0 is full rank and we can find an invertible sub-matrix M_0 of $\mathcal{J}_0$, and solve for the corresponding leading variables $w^{(0)}$ locally, which are t-derivatives of the dependent variables, by using the Implicit Function Theorem. Let the solved form be $w^{(0)} = f^{(0)}(z)$. Let S_0 be the set of the dependent variables of $w^{(0)}$. For the next block B_1 we can choose an invertible sub-matrix M_1 of $\mathcal{J}_1$ which contains M_0 by Lemma 4.6.3. Let S_i is the set of dependent variables of $w^{(i)} \setminus (S_0 \cup \cdots \cup S_{i-1})$.

Continue the process until the last block B_{k_c}. We can check that the union of all S_i is the set of all dependent variables.

Suppose that $U_{k_c} = \{\theta_1, \cdots, \theta_m\}$ and (after appropriate re-indexing) satisfies the condition: for any $1 \leq i < j \leq m$, if the dependent variables of θ_i and θ_j belong to S_p and S_q respectively then $p \leq q$. We can define a positive ranking $\prec$ by the Proposition 4.6.1 such that the solved term is leading variable for each solved form

in $\{w^{(i)} = f^{(i)}(z)\}$. □

For background on the Implicit Function Theorem and related results needed in what follows please see [5, 9]. Let $w_0 \in \mathbb{F}^k$, $z_0 \in \mathbb{F}^\ell$ and $\mathcal{U} \subset \mathbb{F}^k \times \mathbb{F}^\ell$ be a neighborhood of (w_0, z_0).

Let $F : \mathcal{U} \to \mathbb{F}^k$ be an analytic function with $F(w_0, z_0) = 0$ and $\text{rank}\frac{\partial F}{\partial w} = k$ at $(w_0, z_0) \in \mathcal{U}$. That is, the Jacobian of F has maximal rank with respect to w at (w_0, z_0). Then by the Implicit Function Theorem there exists an analytic function $f : \mathbb{F}^\ell \to \mathbb{F}^k$, such that the zero set of $\{(w, z) : F(w, z) = 0\}$ is equivalent to $\{(w, z) : w = f(z)\}$ in a neighborhood of $\mathcal{N}$ of (w_0, z_0).

We have the following simple consequence.

Remark 4.6.1. *There exists a neighborhood of $\mathcal{N}$ of (w_0, z_0) and an analytic function $H : \mathcal{N} \to \mathbb{F}^{k\times k}$ such that*

$$F(w, z) = H(w, z)(w - f(z)) \tag{4.6.1}$$

and $H(w, z)$ is invertible in $\mathcal{N}$.

Theorem 4.6.5. *Let R be a square $\mathbb{F}$-analytic system of* PDE. *Suppose the maximal transversal value of $(\sigma_{ij})(R)$ exists. Let $\boldsymbol{D}_t^c R$ be the system computed by t-prolongation method. If $\mathcal{J}_{k_c}$ is nonsingular at some point p in $Z(\boldsymbol{D}_t^c R)$, then $\boldsymbol{D}_t^c R$ is an Implicit Riquier Basis.*

Proof. By Proposition 4.6.1, there is a ranking in which all leading derivatives are pure t-derivatives. And by Lemma 4.6.4, there exists a solved form $w = f(z)$ of $\mathbf{D}_t^c R$ in a sufficiently small neighborhood $\mathcal{N}_p$, where w is the union of all $w^{(i)}$ defined in Lemma 4.6.4. We will show that $w = f(z)$ is a Riquier Basis in $\mathcal{N}_p$. First note that the principal derivatives of $w = f(z)$ are given by w. Thus $w = f(z)$ is certainly polynomial in w as required by Theorem 4.6.1. Secondly, it remains to prove that the integrability conditions of $w = f(z)$ are satisfied. So without loss of generality, we consider two particular equations $\hat{w} - \hat{f}(z) = 0$ and $\tilde{w} - \tilde{f}(z) = 0$ with $(\mathbf{D}_t)^\gamma \hat{w} = \tilde{w}$. By Theorem 4.6.2, the corresponding integrability condition is $(\mathbf{D}_t)^\gamma(\hat{w} - \hat{f}(z)) - (\tilde{w} - \tilde{f}(z))$. By the more refined redundancy criterion given in Corollary 5.3.2 of [18], this can be reduced to case $\gamma = 1$:

$$\mathbf{D}_t(\hat{w} - \hat{f}(z)) - (\tilde{w} - \tilde{f}(z)) \tag{4.6.2}$$

where $\hat{w} - \hat{f}(z) = 0$ and $\tilde{w} - \tilde{f}(z) = 0$ are two particular equations out of the solved forms $w^{(i-1)} = f^{(i-1)}(z)$ and $w^{(i)} = f^{(i)}(z)$ respectively, with $\mathbf{D}_t \hat{w} = \tilde{w}$.

Remark 4.6.1 implies that $w^{(i)} - f^{(i)}(z) = H_i^{-1} \cdot B_i$ in $\mathcal{N}_p$. Thus $\tilde{w} - \tilde{f}(z) = \tilde{h} \cdot B_i$ in $\mathcal{N}_p$, for some analytic function vector $\tilde{h}$. Similarly $\hat{w} - \hat{f}(z) = \hat{h} \cdot B_{i-1}$ in $\mathcal{N}_p$, for some analytic function vector $\hat{h}$. Then (4.6.2) is

$$\mathbf{D}_t(\hat{h} \cdot B_{i-1}) - \tilde{h} \cdot B_i \tag{4.6.3}$$

which has the general form

$$\mathbf{D}_t\hat{h} \cdot B_{i-1} + \hat{h} \cdot \mathbf{D}_t B_{i-1} - \tilde{h} \cdot B_i \tag{4.6.4}$$

Because $\mathbf{D}_t B_{i-1} \subseteq B_i$, (4.6.2) is zero on $\mathcal{N}_p \cap Z(\mathbf{D}_t^c R)$, which is equivalent to $\{(w,z) : w = f(z)\} \cap \mathcal{N}_p$. So (4.6.2) is zero when $w = f(z)$ in $\mathcal{N}_p$, which means (4.6.2) can be reduced to zero by $w = f(z)$ locally. Therefore $\mathbf{D}_t^c R$ is an implicit Riquier Basis in $\mathcal{N}_p$. □

Remark 4.6.2. *Suppose the maximal transversal value of a signature matrix exists. Then the prolongation step vector c is determined only by the signature matrix rather than the algebraic degree and coefficients. So a signature matrix corresponds to a class of t-dominated* PDE. *For a square polynomially nonlinear* PDE *system R in such a class, if the coefficient of each term is generic, then at a generic point in the variety defined by* $\boldsymbol{D}_t^c R$ *in Jet space, the Jacobian matrix* $\mathcal{J}_{k_c}$ *is non-singular. This means the t-prolongation method can be applied to a large class of* PDE *together with Proposition 4.4.1.*

4.7 Approximating Points on Zero Sets of PDE

The method we have developed depends on finding a point p on the zero set $Z(R)$ of the PDE system R to test that the relevant Jacobians are non-singular. Their non-singularity at a point (and thus in a neighbourhood) ensures that the conditions for local existence and uniqueness are satisfied for Theorem 4.6.5.

We consider polynomially nonlinear PDE as polynomial systems in Jet space. Our tool to numerically solve polynomial systems is homotopy continuation. In [24], a new field "Numerical Algebraic Geometry" was described which led to the development of homotopies to describe all irreducible components (all meaning: for all dimensions) of the solution set of a polynomial system by witness sets. These methods have been implemented in PHCpack [25].

Following Pryce's idea in [13], we can compute $p \in Z(R)$ by exploiting the triangular block structure of the PDE system after the partial prolongation (see Table 4.1).

Remark 4.7.1. *In the case of* ODE, *we can compute the witness points of* B_0, *which is the projection of the variety to the subspace, then substitute the solutions into* B_1 *to extend the solutions to higher dimensional space. Continuing this process, we can find the the witness points of non-singular components. This way is more efficient than solving the whole polynomial system directly. Let R be a polynomially* ODE $\{R_1, ..., R_m\}$ *with total degree d. Then the Bezout bound of* $\boldsymbol{D}_t^c(R)$ *in Jet space is* $d^C d^m$, *where* $C = \sum c_i$. *However if we solve it by bottom up substitution it only has at most* d^m *homotopy continuation paths to track, since any nonlinear equation will be linear with respect to highest Jet variables after prolongation.*

Usually applications involve finding real solutions. For real differential polynomial systems using our approach, we need to find points on a real variety. Real algebraic geometry is a rapidly developing area with many recent developments detailed in the book [3]. There are several techniques for compact varieties while approaches are less well-developed in the non-compact case. Lu [12] uses homotopy continuation in $\mathbb{C}$ to decompose varieties first over $\mathbb{C}$, then obtains points on the real curves embedded in the 1-dimensional complex components. In our experiments, we heuristically selected some proper real linear equations to slice the variety to obtain real points on the zero set of the PDE.

4.8 Examples

The t-prolongation procedure for ODE and PDE was implemented in Maple 10. The integer linear programming involved using Maple10's LPSolve command. As a feasibility test we applied the code to a Test Set of Visconti [26] containing 27 DAE representing diverse applications, with index ranging from 1 to 6. The procedure successfully identified index consistent with Visconti's results for 21 of the DAE. The LP problems were solved in less, and often much less, than one second. Our 6 failures were due to: 3 non-square system; 3 systems with singular Jacobians. Like other standard DAE approaches, Visconti required the user to supply an initial guess for a consistent initial point, and then Gauss-Newton iteration was applied. An example is given below.

EXAMPLE 4.8.1 (ODE for a Crane)**.** *This model which is illustrated in Figure 4.8.1, is discussed in [4]. The problem is to determine the horizontal velocity $u_1(t)$ of a winch of mass M_1, and the angular velocity $u_2(t)$ of the winch so that the attached load M_2 moves along a prescribed path.*

The equations of motion are given by [4] and also by Visconti [26] with unknowns $\{x, x', z, z', d, d', r, r', \theta, \tau, u_1, u_2\}$:

$$\begin{aligned} x_t - x' = 0, \quad z_t - z' = 0, \quad d_t - d' = 0, \quad r_t - r' &= 0 \\ M_2\, x'_t + \tau \sin(\theta) = 0, \quad M_1\, d'_t + C_1\, d_t - u_1 - \tau \sin(\theta) &= 0 \\ M_2\, z'_t + \tau \cos(\theta) - mg = 0, \quad J\, r'_t + C_2\, r_t + C_3\, u_2 - {C_3}^2 \tau &= 0 \\ r \sin(\theta) + d - x = 0, \quad r \cos(\theta) - z &= 0 \\ H_1(x, z, t) = 0, \quad H_2(x, z, t) &= 0. \end{aligned}$$

The prescribed path of the mass M_2 is described by an algebraic equations $\{H_1 = 0, H_2 = 0\}$. The winch has moment of inertia J and is attached with a cable of length $r(t)$, making an angle $\theta(t)$ to the vertical.

Substitute $sin(\theta)$ and $cos(\theta)$ by $s(t)$ and $c(t)$ respectively to convert the ODE to an algebraic differential system, and introduce an extra equation $s(t)^2 + c(t)^2 = 1$.

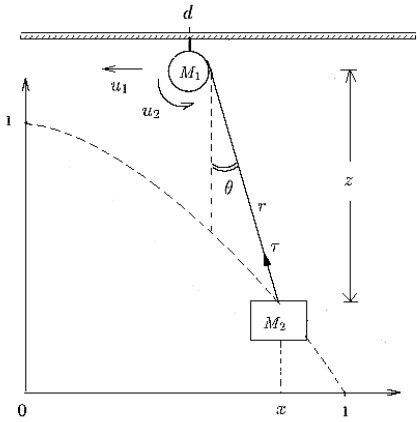

Figure 4.8.2: Control of a Crane

Applying the t-prolongation method and our Maple program, we obtain

$$d_1 = 4, d_2 = 3, d_3 = 4, d_4 = 3, d_5 = 2, d_6 = 1, d_7 = 2,$$
$$d_8 = 1, d_9 = 2, d_{10} = 0, d_{11} = 0, d_{12} = 2, d_{13} = 2;$$
$$c_1 = 3, c_2 = 3, c_3 = 1, c_4 = 1, c_5 = 2, c_6 = 2, c_7 = 0,$$
$$c_8 = 0, c_9 = 2, c_{10} = 2, c_{11} = 4, c_{12} = 4, c_{13} = 2.$$

Since d_{10} and d_{11} are equal to zero, we need to prolong one more time to reduce the system to ODE. *For this example we have index* 5 *in agreement with Visconti. Note that the result does not depend on the coefficients and degrees of H_1, H_2 since the signature matrix only requires the differential orders of H_1, H_2 which are both* 0.

To simply illustrate how to use the output, we choose a path $\{H_1 = 0, H_2 = 0\}$ of the mass M_2 which is described by a parameterized system $x(t) = 1 - t^2, z(t) = 1 - t$. After the partial prolongation we obtain 13 ODE *and* 39 *algebraic constraints. The total Bezout degree of the constraints is* 65536, *however it has block triangular structure which enable us to solve it by bottom up substitution.*

Let the initial time t be .396, *we obtain* 4 *witness points using PHCpack. We choose one as the initial point which is $x(.396) = .843$, $z(.396) = .604$, $d(.396) = .601$, $s(.396) = .371$, $c(.396) = .928$, $r(.396) = 0.650$. Note that if the degree of H_1, H_2 is d, there are at most $4d^2$ witness points during the computation by Remark 4.7.1. The computational difficulty of this problem for the symbolic differential elimination algorithm* Rifsimp *explosively increases with the degree d of H_1, H_2 in comparison with the t-prolongation method.*

Finally we numerically solve the ODE *together with this initial condition using* dsolve *in Maple10 with its implicit option. The integral curve of $x(t), z(t)$ is very close to the curve $(1 - t^2, 1 - t)$.*

EXAMPLE 4.8.2 (Pendulum Curtain PDE)**.** *Applying the t-prolongation method to Example 4.5.1 gives:*

$$c_1 = 0, \quad c_2 = 0, \quad c_3 = 2; \tag{4.8.1}$$

$$d_1 = 2, \quad d_2 = 2, \quad d_3 = 0. \tag{4.8.2}$$

The main point is that the analysis for this PDE *example is virtually identical to that for the classical pendulum (see Pryce [13] for those details). Essentially the analysis indicates that the constraint should be differentiated twice to yield an implicit Riquier Basis:*

$$\begin{aligned} X_{tt} + \lambda X &= \kappa X_{ss} \\ Y_{tt} + \lambda Y + g &= \kappa Y_{ss} \\ XX_{tt} + YY_{tt} + X_t^2 + Y_t^2 &= 0 \\ XX_t + YY_t &= 0 \\ X^2 + Y^2 - 1 &= 0. \end{aligned} \tag{4.8.3}$$

The top block B_2 of the system is the first three equations of (4.8.3). The blocks B_1 and B_0 are the 4th and 5th equations of (4.8.3) respectively.

Then the system has Jacobian matrix with respect to X_{tt}, Y_{tt}, λ which is full rank. This is also obvious by inspection, without using the generalization of Pryce's method. We include it here, so that the reader can see it working on an example, which is closely related a one of the fundamental examples of DAE *theory. We note that a change of coordinates to cylindrical coordinates $X = \sin(\theta(s,t))$, $Y = -\cos(\theta(s,t))$, considerably simplifies the problem. However, in general, such coordinate changes cannot be algorithmically made to eliminate all constraints for* PDE*.*

In summary we obtain an explicit hyperbolic system on a system of constraints. Just as an explicit ODE *is uncovered in the analysis of the classical pendulum, an explicit Hyperbolic System of* PDE *is obtained in the Pendulum Curtain example. We solved this system using Wittkopf's finite difference code in Maple10. We performed experiments with various initial and boundary conditions and values of κ. One of these was for an exponential bump located in the middle of the s-range, where the curtain is released from rest. As expected this forms two waves, moving in opposite directions. If the coefficient κ of the X_{ss} and Y_{ss} terms are set close to zero (i.e. $\kappa \approx 0$) then as expected the pendulum motion rather than the wave motion dominates.*

EXAMPLE 4.8.3 (Changing the Coordinates)**.** *The equation below is both x and y dominated. However for small ϵ_1, ϵ_2, the resulting Jacobians in our method are poorly conditioned.*

$$\epsilon_1 u_{xx} + u_{xy} + \epsilon_2 u_{yy} = 0 \tag{4.8.4}$$

The problem is well conditioned after a coordinate change (see Proposition 4.4.1).

4.9 Discussion

A significant problem in the development of symbolic-numeric differential elimination methods is to create methods to control the growth of prolongations. Although much progress has been made on the symbolic case [1], little has been done for the symbolic-numeric case.

In the current work we define a class of systems for which only prolongations with respect to a single independent variable t are needed.

We generalized Pryce's technique in the framework of Riquier Bases. Riquier's classical approach has fallen out of favor in recent times, since for a purely symbolic implementation it is limited to systems linear in their highest derivatives, and modern symbolic alternatives now exist [2, 27]. However in our article, Riquier's approach makes a comeback, by using the Implicit Function Theorem, which requires points on the zero set of the system. For systems of differential polynomials over $\mathbb{C}$, we can use homotopy methods from Numerical Algebraic Geometry to compute approximations to such points [24]. For systems of differential polynomials over $\mathbb{R}$, there are also rapidly evolving methods [12, 3]. For analytic systems, methods are less systematic but progress can be made using Gaussian-Newton iteration from initial guesses close enough to a solution.

It may seem strange that such implicit representations could be useful, especially since the representations given by such symbolic elimination methods as [2] provide output systems in much closer to explicit solved or triangular form. However such eliminations can often cause severe expression swell. The Pryce method appears to find a balance between working implicitly, and at the same time uncovering and exploiting the block structure of a system. Finally we note that such implicit representations are usually the choice in the numerics community. Solving a constant matrix system, at the intermediate steps of a numerical integration, is often preferred over first symbolically inverting, then evaluating the explicit solution at those intermediate steps.

The disadvantages of our method include its limitation to square and t-dominated systems. It also has the disadvantage that it is a local method, and not a universal method, and does not pursue all singular cases as is possible using [2, 7]. For example when the method is applied to $(u_t)^2 + tu_t - u = 0$ it locates a generic initial point, and does not identify the fact that this equation has a singular solution. In addition, a linear combination of the input system will destroy the sparse structure of the signature matrix. However this can be detected by a rank test and the hidden equations can be constructed by the methods we give in [28].

The implicit Riquier Bases obtained by our method are a type of formally integrable system. Such bases only give local, and sometimes unnatural, boundary and initial conditions. We direct the reader to Krupchyk et al. [10, 11] for very interesting work on linking formal properties (such as formal integrability and involutivity) to elliptic BVP.

Our method provides a bridge between ODE techniques and PDE techniques. In this paper we generalized a method of Pryce and Pantiledes, to PDE. An obvious future work, is to attempt the same with other ODE methods. We are investigating PDE models arising as more realistic cases of DAE system, for which our t-prolongation method promises to be practically useful.

Acknowledgement

The authors gratefully acknowledge support for this work from the NSF funded IMA Thematic Year on Applications of Algebraic Geometry, and also acknowledge support from Reid's NSERC grant. We especially thank Jan Verschelde and John McPhee for many discussions and valuable assistance. We also thank Silvana Ilie, Eric Schost and Allan Wittkopf for helpful discussions. We thank the Referees for many helpful comments.

Bibliography

[1] F. Boulier. *Réécriture algébrique dans les systèmes d'équations différentielles en vue d'applications dans les Sciences du Vivant*. Habilitation Thesis, 2006.

[2] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Representation for the radical of a finitely generated differential ideal. Proc. ISSAC 1995. ACM Press. 158–166, 1995.

[3] S. Basu, R. Pollack and M. Roy. Algorithms in Real Algebraic Geometry, Springer, 2003.

[4] S.L. Campbell. High index differential algebraic equations. J. Mech. Struct. and Machines, 23: pp 199-222, 1995.

[5] R.C. Gunning and H. Rossi. Analytic functions of several complex variables. Prentice-Hall, 1965.

[6] S. Ilie, R.M. Corless and G. Reid. Numerical solutions of index-1 differential algebraic equations can be computed in polynomial time. Numerical Algorithms, Vol. 41 (2), pp. 161–171, 2006.

[7] E. Hubert. Notes on triangular sets and triangulation-decomposition algorithms II: Differential Systems. *Symbolic and Numerical Scientific Computations*, Edited by U. Langer and F. Winkler. LNCS, volume 2630, Springer-Verlag Heidelberg, 2003.

[8] E. Kolchin. Differential Algebra and Algebraic Groups. Academic Press, New York, 1973.

[9] S. Krantz and H. Parks. A Primer of Real Analytic Functions. Basler Lehrbücher, 2002.

[10] K. Krupchyk and J. Tuomela. Shapiro-Lopatinskij Condition for Elliptic Boundary Value Problems. LMS J. Comput. Math. 9 (2006) pp. 287–329.

[11] K. Krupchyk, W. Seiler and J. Tuomela. Overdetermined Elliptic PDEs. Found. Comp. Math. 6 (2006), No. 3, pp 309–351.

[12] Y. Lu. Finding all real solutions of polynomial systems. Ph.D. Thesis, University of Notre Dame, 2006 (Submitted).

[13] J.D. Pryce. A Simple Structure Analysis Method for DAEs. BIT, vol 41, No. 2, pp. 364-394, 2001.

[14] E. Mansfield. *Differential Gröbner Bases.* Ph.D. thesis, Univ. of Sydney, 1991.

[15] G. Reid, C. Smith, and J. Verschelde. Geometric completion of differential systems using numeric-symbolic continuation. *SIGSAM Bulletin* 36(2):1–17, 2002.

[16] G. Reid, J. Verschelde, A.D. Wittkopf and W. Wu. Symbolic-Numeric Completion of Differential Systems by Homotopy Continuation. Proc. ISSAC 2005. ACM Press. 269–276, 2005.

[17] G.J. Reid, P. Lin, and A.D. Wittkopf. Differential elimination-completion algorithms for DAE and PDAE. *Studies in Applied Math.* 106(1): 1–45, 2001.

[18] C.J. Rust, *Rankings of derivatives for elimination algorithms and formal solvability of analytic partial differential equations*, Ph.D. Thesis, University of Chicago, 1998.

[19] C.J. Rust, G.J. Reid, and A.D. Wittkopf. Existence and uniqueness theorems for formal power series solutions of analytic differential systems. Proc. ISSAC 99. ACM Press. 105-112, 1999.

[20] F. Schwarz. Janet bases for symmetry groups, in: Groebner bases and applications. London Math. Soc., LNS 251, Cambridge Univ. Press, 221–234, 1998.

[21] A. Sedoglavic. A probabilistic algorithm to test local algebraic observability in polynomial time. *J. Symbolic Computation* 33(5): 735–755, 2002.

[22] W.M. Seiler. *Involution - The formal theory of differential equations and its applications in computer algebra and numerical analysis.* Habilitation Thesis, Univ. of Mannheim, 2002.

[23] Pengfei Shi, John McPhee. Symbolic Programming of a Graph-Theoretic Approach to Flexible Multibody Dynamics. Mechanics of Structures and Machines, 30(1), 123–154, 2002.

[24] A.J. Sommese and C.W. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science.* World Scientific Press, Singapore, 2005.

[25] J. Verschelde. Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. *ACM Transactions on Mathematical Software* 25(2): 251–276, 1999.

[26] J. Visconti. *Numerical Solution of Differential Algebraic Equations, Global Error Estimation and Symbolic Index Reduction.* Ph.D. Thesis. Laboratoire de Modélisation et Calcul. Grenoble. 1999.

[27] A. Wittkopf. *Algorithms and Implementations for Differential Elimination.* Ph.D. Thesis, Simon Fraser University, 2004.

[28] Wenyuan Wu and Greg Reid. Application of Numerical Algebraic Geometry and Numerical Linear Algebra to PDE. Proc. of ISSAC'06, pages 345-352, ACM 2006.

Chapter 5

On Approximate Triangular Decompositions in Dimension Zero

Triangular decompositions for systems of polynomial equations with n variables, with exact coefficients are well-developed theoretically and in terms of implemented algorithms in computer algebra systems. However there is much less research about triangular decompositions for systems with approximate coefficients.

In this paper we discuss the zero-dimensional case of systems having finitely many roots. Our methods depend on having approximations for all the roots, and these are provided by the homotopy continuation methods of Sommese, Verschelde and Wampler. We introduce approximate equiprojectable decompositions for such systems, which represent a generalization of the recently developed analogous concept for exact systems. We demonstrate experimentally the favourable computational features of this new approach, and give a statistical analysis of its error.

5.1 Introduction

Ritt initiated the algebraic study of differential polynomial systems through characteristic sets [28]. Their modern study was revitalized by the work of Wu. In [41], he adapted the work of Ritt for solving algebraic systems: he showed that the zero set of such a system could be decomposed as finitely many characteristic sets, leading to the notion of a triangular decomposition of an algebraic variety. Considerable developments have followed by many authors; among them: Aubry et al. [1], Chou [7], Dahan et al. [11], Gao et al. [15], Kalkbrener [19], Lazard [20], Moreno Maza [25], Schost [29], Wang [40], and others. These works have led to efficient algorithms for triangular decomposition of an algebraic variety given by an exact input polynomial system.

Often, in applications we are interested in producing a useful triangular form where some of the variables are functions of others. Such systems frequently have approximate coefficients that are inferred from experimental data. This means that

the stability, or sensitivity to coefficient changes, of such triangular decompositions is a concern. While considerable progress in both theoretical and algorithmic aspects has been made for exact input polynomial systems, much less is known about generalizations of these methods to input systems which are approximate.

However, in recent years, motivated by many realistic problems, some related work has been made, for example: numerical Gröbner Bases by Stetter [35] and the study about approximate radical of zero-dimension ideals by Szántó et. al. [18].

In this paper, we present some initial results in this direction, for the case of an algebraic variety V over $\mathbb{C}$. We rely on the methods of Sommese, Verschelde, and Wampler [31, 38, 24, 32] which use Homotopy continuation to determine so-called generic points on the components of the numerical decomposition of V. We are interested in the set V_0 of the isolated points of V (the 0 dimensional case). Each point of V_0, and more generally every irreducible component of V, is trivially a triangular set, although not generally rationally constructible from rational input. This is in contrast to the usual forms of exact triangular decomposition, which are modeled on equi-dimensional decomposition over $\mathbb{Q}$ rather than irreducible decomposition over $\mathbb{C}$.

Following [10, 11], we consider the equiprojectable decomposition of V_0. Then, we use the interpolation formulas of Dahan and Schost [12] for computing an approximate triangular set for each equiprojectable component of V_0, leading to an approximate triangular decomposition of V_0 in Section 5.3.

We provide a stability analysis of the interpolation formulas of Dahan and Schost in Section 5.4. One of our main tools is Lindeberg's theorem [30] that is described in the Appendix. In Sections 5.5 and 5.6, we report on experiments that illustrate the efficiency of our approach and support the accuracy of our stability analysis.

In [27], we study the simplest class of positive dimensional systems: linear homogeneous systems. Our aim in that article is to explore local structure of non-linear problems with linearized approximate triangular decompositions. The combination of the two approaches allows us to form an accessible bridge to the study of the fully non-linear case which we will describe in a forthcoming paper.

5.2 Triangular decompositions

A triangular decomposition of a zero-dimensional algebraic variety V is a family of polynomial sets, called triangular sets, that describe symbolically the points of V [20]. Triangular decompositions extend to algebraic varieties of arbitrary dimension, see for instance [19, 25]. In [12] it is shown that the height of a coefficient in a triangular set T can be bounded by the height of the variety represented by T. Combined with the notion of *equiprojectable decomposition* introduced in [10], this motivated the work of [11], in which the authors obtained a very efficient method for computing triangular decompositions of zero-dimensional varieties over $\mathbb{Q}$ given by an input polynomial system with exact coefficients.

On top of these good computational properties, triangular sets and triangular decompositions have natural geometrical interpretations. In Section 5.3, we will rely on these properties to introduce a notion of an *approximate triangular decomposition* of a zero-dimensional variety given by approximate coordinates of its points. In the present section, we recall some results for triangular decompositions in the exact case and refer to [12, 10, 11] for more details. For the reader's convenience, we sketch the proof of Propositions 5.2.1 and 5.2.2, which play a central role in this paper. See [12] for their complete proofs.

Let $\mathbb{K}$ be a perfect field, let $\mathbb{L}$ be an algebraic closure of $\mathbb{K}$ and let $X_1 \prec \cdots \prec X_n$ be $n \geq 1$ ordered variables.

Definition 5.2.1. *A set $T = \{T_1, \ldots, T_n\}$ of n polynomials in $\mathbb{K}[X_1, \ldots, X_n]$ is a* triangular set *if the ideal $\langle T \rangle$ generated by T is radical and if for all $1 \leq i \leq n$ the polynomial T_i is not constant, the greatest variable occurring in T_i is X_i, and its leading coefficient w.r.t. X_i is invertible modulo the ideal $\langle T_1, \ldots, T_{i-1} \rangle$. The triangular set T is* normalized *if for all $1 \leq i \leq n$ the leading coefficient of T_i w.r.t. X_i is one.*

Clearly, a triangular set generates a zero-dimensional ideal and a normalized triangular set is a reduced lexicographical Gröbner basis. In [20], it is shown that every maximal ideal of $\mathbb{K}[X_1, \ldots, X_n]$ can be generated by a triangular set. Hence, a natural question is to characterize the zero-dimensional varieties over $\mathbb{K}$ that can be generated by a triangular set. The answer is given by [3]. We report on it here by means of Definition 5.2.2 and Theorem 5.2.1, after introducing some notation.

Let i and j be integers such that $1 \leq i \leq j \leq n$. We denote by $A^i(\mathbb{L})$ the affine space of dimension i over $\mathbb{L}$. For $V \subseteq A^n(\mathbb{L})$ we denote by $\mathcal{I}(V)$ the ideal of $\mathbb{K}[X_1, \ldots, X_n]$ composed by the polynomials which vanish on V. For $F \subseteq \mathbb{K}[X_1, \ldots, X_n]$ we denote by $V(F)$ the set of the points of $A^n(\mathbb{L})$ where every element of F vanishes. Finally, we denote by π_i^j the natural projection map from $A^j(\mathbb{L})$ to $A^i(\mathbb{L})$, which sends $(X_1, \ldots, X_j)$ to $(X_1, \ldots, X_i)$.

Definition 5.2.2. *A zero-dimensional variety $V \subseteq A^j(\mathbb{L})$ over $\mathbb{K}$ is said to be*

(1) equiprojectable on $V_i = \pi_i^j(V)$, *its projection onto $A^i(\mathbb{L})$, if there exists an integer c such that for every $M \in V_i$ the cardinality of $(\pi_i^j)^{-1}(M) \cap V_i$ is c.*

(2) equiprojectable *if V is equiprojectable on $V_1, \ldots, V_{j-1}$.*

Theorem 5.2.1. *A zero-dimensional variety $V \subseteq A^j(\mathbb{L})$ over $\mathbb{K}$ is equiprojectable if and only if there exists a triangular set T of $\mathbb{K}[X_1, \ldots, X_j]$ such that T generates $\mathcal{I}(V)$.*

Given an equiprojectable variety $V \subseteq A^n(\mathbb{L})$ the normalized triangular set T generating $\mathcal{I}(V)$ can be constructed as follows from the coordinates of the points of V (see [12] for details). Let $\mathbf{K}$ be a field such that $\mathbb{K} \subseteq \mathbf{K} \subseteq \mathbb{L}$ and such that

every point of V has its coordinates in $\mathbf{K}$. We define $V_i = \pi_i^n(V)$. Let $1 \leq \ell < n$. Following [12], we describe how to interpolate $T_{\ell+1}$ from the coordinates (in $\mathbf{K}$) of the points of $V_{\ell+1}$. Let $\alpha = (\alpha_1, \ldots, \alpha_\ell) \in V_\ell$. Define:

$$\begin{array}{rcl} V_\alpha^1 & = & \{\beta = (\beta_1, \ldots, \beta_\ell, \beta_{\ell+1}) \in V_{\ell+1} \ \mid \ \beta_1 \neq \alpha_1\}, \\ V_\alpha^2 & = & \{\beta = (\alpha_1, \beta_2, \ldots, \beta_\ell, \beta_{\ell+1}) \in V_{\ell+1} \ \mid \ \beta_2 \neq \alpha_2\}, \\ V_\alpha^3 & = & \{\beta = (\alpha_1, \alpha_2, \beta_3, \ldots, \beta_\ell, \beta_{\ell+1}) \in V_{\ell+1} \ \mid \ \beta_3 \neq \alpha_3\}, \\ \cdots & \cdots & \cdots \\ V_\alpha^\ell & = & \{\beta = (\alpha_1, \ldots, \alpha_{\ell-1}, \beta_\ell, \beta_{\ell+1}) \in V_{\ell+1} \ \mid \ \beta_\ell \neq \alpha_\ell\}, \\ V_\alpha^{\ell+1} & = & \{\beta = (\alpha_1, \ldots, \alpha_\ell, \beta_{\ell+1}) \in V_{\ell+1}.\} \end{array} \tag{5.2.1}$$

The sets $V_\alpha^1, V_\alpha^2, V_\alpha^3, \ldots, V_\alpha^\ell, V_\alpha^{\ell+1}$ partition $V_{\ell+1}$. We consider also the projections:

$$\begin{array}{rcccl} v_\alpha^1 & = & \pi_1^{\ell+1}(V_\alpha^1) & = & \{(\beta_1) \in V_1 \ \mid \ \beta_1 \neq \alpha_1\}, \\ v_\alpha^2 & = & \pi_2^{\ell+1}(V_\alpha^2) & = & \{(\alpha_1, \beta_2) \in V_2 \ \mid \ \beta_2 \neq \alpha_2\}, \\ v_\alpha^3 & = & \pi_3^{\ell+1}(V_\alpha^3) & = & \{(\alpha_1, \alpha_2, \beta_3) \in V_3 \ \mid \ \beta_3 \neq \alpha_3\}, \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ v_\alpha^\ell & = & \pi_\ell^{\ell+1}(V_\alpha^\ell) & = & \{(\alpha_1, \ldots, \alpha_{\ell-1}, \beta_\ell) \in V_\ell \ \mid \ \beta_\ell \neq \alpha_\ell\} \end{array} \tag{5.2.2}$$

For $1 \leq i \leq \ell + 1$, we define

$$T_{\alpha,i} = T_i(\alpha_1, \ldots, \alpha_{i-1}, X_i) \ \ \textit{and} \ \ e_{\alpha,i} = \prod_{\beta \in v_\alpha^i} (X_i - \beta_i). \tag{5.2.3}$$

Observe that for $1 \leq i \leq \ell + 1$ we have $T_{\alpha,i} \in \mathbf{K}[X_i]$ and $e_{\alpha,i} \in \mathbf{K}[X_i]$. Finally, we define

$$E_\alpha = \prod_{1 \leq i \leq \ell} e_{\alpha,i} \tag{5.2.4}$$

and note that $E_\alpha \in \mathbf{K}[X_1, \ldots, X_\ell]$ holds.

Proposition 5.2.1. *For $1 \leq i \leq \ell$ we have*

$$T_{\alpha,i} = \prod_{(\alpha_1,\ldots,\alpha_{i-1},\beta_i) \in V_i} (X_i - \beta_i) = e_{\alpha,i} \ (X_i - \alpha_i), \tag{5.2.5}$$

$$T_{\alpha,\ell+1} = \prod_{\beta \in V_\alpha^{\ell+1}} (X_{\ell+1} - \beta_{\ell+1}), \tag{5.2.6}$$

$$T_{\ell+1} = \sum_{\alpha \in V_\ell} \frac{E_\alpha T_{\alpha,\ell+1}}{E_\alpha(\alpha)}. \tag{5.2.7}$$

Proof. Relations (5.2.5) and (5.2.6) follow easily from (5.2.1), (5.2.2) and (5.2.3). In order to prove (5.2.7) we observe that:

$$(\forall \beta \in V_\ell) \ E_\alpha(\beta) = 0 \iff \beta \neq \alpha. \tag{5.2.8}$$

Indeed, for $1 \leq i \leq \ell$, we have $e_{\alpha,i}(\alpha) \neq 0$ leading to $E_\alpha(\alpha) \neq 0$. Now let $\beta \in V_\ell$ with $\beta \neq \alpha$. Then, there exists $i \leq \ell$ such that

$$(\pi_i^\ell)^{-1}(\beta) \in v_\alpha^i.$$

Hence, for this index i we have $e_{\alpha,i}(\beta) = 0$, which proves (5.2.8). From there, establishing (5.2.7) is routine. □

In [12], another triangular set N is obtained from the coordinates of the points of V, see Proposition 5.2.2. The authors show that it has much smaller coefficients than the normalized triangular set given by the formulas of Proposition 5.2.1. We will be generalizing this second triangular set to the approximate case.

Proposition 5.2.2 (Interpolation formulas)**.** *Let $D_1 = 1$ and $\tau_1 = N_1 = T_1$. For $2 \leq \ell \leq n$, define*

$$D_\ell = \prod_{1 \leq i \leq \ell-1} \frac{\partial T_i}{\partial X_i} \mod \langle T_1, \ldots, T_{\ell-1} \rangle \tag{5.2.9}$$

and

$$N_\ell = D_\ell T_\ell \mod \langle T_1, \ldots, T_{\ell-1} \rangle. \tag{5.2.10}$$

Then, for $1 \leq i \leq \ell$ we have

$$N_{\ell+1} = \sum\nolimits_{\alpha \in V_\ell} E_\alpha T_{\alpha,\ell+1}. \tag{5.2.11}$$

Proof. Indeed, for $1 \leq i \leq \ell$, we have

$$T_{\alpha,i} = e_{\alpha,i}\ (X_i - \alpha_i) \in \mathbf{K}[X_i]$$

leading to

$$\begin{aligned} \frac{\partial T}{\partial X_i}(\alpha) &= T'_{\alpha,i}(\alpha) \\ &= e'_{\alpha,i}(\alpha)\ (\alpha_i - \alpha_i) + e_{\alpha,i}(\alpha) \\ &= e_{\alpha,i}(\alpha). \end{aligned}$$

By definition, we have

$$N_{\ell+1} = \left(\prod_{1 \leq i \leq \ell} \frac{\partial T}{\partial X_i} \right) T_{\ell+1} \mod \langle T_1, \ldots, T_\ell \rangle.$$

Hence, we have

$$\begin{aligned} N_{\ell+1}(\alpha) &= \left(\prod_{1 \leq i \leq \ell} \frac{\partial T}{\partial X_i}(\alpha) \right) T_{\ell+1}(\alpha) \\ &= \left(\prod_{1 \leq i \leq \ell} e_{\alpha,i}(\alpha) \right) T_{\ell+1}(\alpha) \\ &= E_\alpha(\alpha)\ T_{\ell+1}(\alpha) \end{aligned}$$

where $T_{\ell+1}(\alpha) = T_{\alpha,\ell+1}$ holds. Finally we obtain

$$\begin{aligned} N_{\ell+1} &= \sum_{\alpha \in V_\ell} \frac{E_\alpha N_{\ell+1}(\alpha)}{E_\alpha(\alpha)} \\ &= \sum_{\alpha \in V_\ell} E_\alpha T_{\ell+1}(\alpha). \end{aligned}$$

□

Clearly, not all zero-dimensional varieties over $\mathbb{Q}$ are equiprojectable. Consider, for example, with $n = 2$ the variety consisting of the three points A, B, C with respective coordinates $(1, 0)$, $(0, 0)$ and $(0, 1)$. However, we do have the following result, see for instance [20].

Proposition 5.2.3. *For every zero-dimensional radical ideal $\mathcal{I}$ of $\mathbb{K}[X_1, \ldots, X_n]$ there exists finitely many triangular sets $T^1, \ldots, T^e$ such that $\mathcal{I}$ is the intersection of the ideals $\langle T^1 \rangle$, $\ldots$, $\langle T^e \rangle$. If, in addition, the ideals $\langle T^1 \rangle$, $\ldots$, $\langle T^e \rangle$ are pairwise relatively prime, then the set $\{T^1, \ldots, T^e\}$ is called a* triangular decomposition *of the ideal $\mathcal{I}$.*

Triangular decompositions of algebraic varieties (with arbitrary dimension) are discussed in depth in [25] together with an algorithm for computing them, which is implemented in [22]. Observe that a radical ideal may admit several triangular decompositions. For instance, there are four different triangular decompositions for the ideal $\mathcal{I}(\{A, B, C\})$. Choosing a canonical triangular decomposition for the radical $\mathcal{I}$ with the variable ordering $X_1 \prec \cdots \prec X_n$ is achieved by the following combinatorial construction. We refer to [11] for a more formal definition.

Definition 5.2.3. *Consider a zero-dimensional variety V and denote by $\pi = \pi_{n-1}^n$ the projection which removes the last coordinate. To a point x in V, we associate $N(x) = \#\pi^{-1}(\pi(x))$, that is, the number of points lying in the same π-fiber as x. Then, we split V into the disjoint union $V_1 \cup \cdots \cup V_d$, where for all $i = 1, \ldots, d$, V_i equals $N^{-1}(i)$, that is, the set of points $x \in V$ which have $N(x) = i$. This splitting process is applied recursively to all varieties $V_1, \ldots, V_d$, taking into account the fibers of the successive projections π_i^n, for $i = n-1, \ldots, 1$. In the end, we obtain a family*

of pairwise disjoint, equiprojectable varieties, whose reunion equals V; they form the equiprojectable decomposition *of V.*

5.3 Approximate Equiprojectable Decomposition in Dimension Zero

In this section, we consider a zero-dimensional variety $V \subseteq A^n(\mathbb{C})$ over $\mathbb{Q}$. Each point of V is given by approximate coordinates in a sense that we make precise in Definition 5.3.1. We aim at defining and computing an *approximate triangular decomposition* of V. To do so, we extend the construction given by Definition 5.2.3 and introduce a notion of an *approximate equiprojectable decomposition* of V in Definition 5.3.4. Then, to each approximate equiprojectable component, we associate an approximate triangular set, leading to Definition 5.3.5 of an *approximate triangular decomposition of V*.

Therefore, an approximate triangular decomposition of V is obtained by interpolating the points of V given by approximate coordinates. We provide a stability analysis for this interpolation in Section 5.4. Moreover, we report on experiments that illustrate the accuracy of our stability analysis in Sections 5.5 and 5.6.

Definition 5.3.1. *Let $\epsilon > 0$ and $r \geq 0$ be real numbers. Let $\bar{x} = (\bar{x}_1, \ldots, \bar{x}_n)$ be a point of V and let $x = (x_1, \ldots, x_n) \in A^n(\mathbb{C})$ with $x \neq 0$. We say that (x, r) is an* approximate point *for $\bar{x}$ with tolerance ϵ, denoted by $\bar{x} \simeq_\epsilon (x, r)$, if the following conditions hold for all $1 \leq i \leq n$:*

(i) $\mid \bar{x}_i - x_i \mid \leq \ r$,

(ii) $r \ \leq \ \epsilon \mid x \mid$.

where $\mid x \mid = \max(\mid x_1 \mid, \ldots, \mid x_n \mid)$.

With the notations of Definition 5.3.1 let (x, r) be an approximate point for $\bar{x}$ with tolerance ϵ. Let $1 \leq i \leq n$ be fixed. If $\bar{x}_i$ and x_i are complex numbers and $\bar{x}_i \neq 0$ then a frequently-used measure of the number of correct significant decimal digits in the approximate coordinate x_i is the *logarithm of the relative error* $\mathrm{lre}(x_i, \bar{x}_i)$ given by

$$\mathrm{lre}(x_i, \bar{x}_i) = -\log_{10} \frac{\mid \bar{x}_i - x_i \mid}{|\bar{x}_i|}. \tag{5.3.1}$$

Properties (i) and (ii) of Definition 5.3.1 lead to

$$\mathrm{lre}(x_i, \bar{x}_i) \ \geq \ -\log_{10} \epsilon - \log_{10} \frac{|x|}{|\bar{x}_i|}. \tag{5.3.2}$$

In practice, one requires $\epsilon < 1$ and thus Formula (5.3.2) gives a good measure of the approximation of coordinate $\bar{x}_i$ by means of coordinate x_i. Similarly, Formula (5.3.3)

below gives a good measure of the approximation of point $\bar{x}$ by means of point x, for $x \neq 0$:

$$lre(x, \bar{x}) = -\log_{10} \frac{| \bar{x} - x |}{|\bar{x}|}. \tag{5.3.3}$$

As we shall see now, another good measure of this approximation is

$$lb(\bar{x}, x) = -\log_{10} \frac{| \bar{x} - x |}{|x|}. \tag{5.3.4}$$

Indeed, one can easily check that the following holds:

$$\left| \log_{10} \frac{| \bar{x} - x |}{|x|} - \log_{10} \frac{| \bar{x} - x |}{|\bar{x}|} \right| \; = \; \left| \log_{10} \frac{|\bar{x}|}{|x|} \right|. \tag{5.3.5}$$

Moreover, we claim that when ϵ is close to zero:

$$\left| \log_{10} \frac{|\bar{x}|}{|x|} \right| \; \approx \; \epsilon \; . \tag{5.3.6}$$

Thus, $lre(x, \bar{x})$ and $lb(\bar{x}, x)$ are very close when ϵ is very small. To prove our claim, we start from

$$||\bar{x}| - |x|| \; \leq | \bar{x} - x | \leq \; \epsilon \, |x|, \tag{5.3.7}$$

which holds by assumption (points (i) and (ii) of Definition 5.3.1). We deduce

$$\left| \frac{|\bar{x}|}{|x|} - 1 \right| \; \leq \; \epsilon. \tag{5.3.8}$$

Since ϵ is meant to be very small, using $\log_{10}(1 - \epsilon) \approx -\epsilon$ and $\log_{10}(1 + \epsilon) \approx \epsilon$, we finally obtain Formula (5.3.6).

A representation (using approximate points in the sense of Definition 5.3.1) of the isolated roots of the variety $V \subseteq A^n(\mathbb{C})$ of an input polynomial system $F = \{F_1, \ldots, F_n\} \subset \mathbb{Q}[X_1, \ldots, X_n]$ can be obtained by numerical homotopy construction. In particular, we used the `PHC` software [38]. Indeed, for each point $\bar{x}$ of V, the corresponding solution x returned by `PHC` is given with the condition number of the Jacobian matrix of F at x, denoted by *cond*. The value *cond* can be used to estimate the distance between $\bar{x}$ and x (see [24] for details). More precisely, because we use double precision floating-point numbers in the computation, a reasonable formula is: $| \bar{x}_i - x_i | / |x_i| \approx cond \cdot 10^{-16}$ for all $1 \leq i \leq n$ (see Table 5.4). Given $\epsilon > 0$, with this estimate, one can check whether each isolated point $\bar{x}$ of V admits approximate points within tolerance ϵ. Theoretically, the homotopy continuation method can obtain approximate points arbitrarily close to the exact roots for any tolerance ϵ. So, if the multiplicity of each point is 1, a one-to-one map between approximate roots and exact ones can be computed. Note that none of the systems

used in Section 5.6 have multiple roots (see Table 5.2).

Remark 5.3.1. The definition of approximate points of a polynomial systems is related to alpha-theoretic concepts of approximate zero [5]. Although alpha theory can determine a basin in which Newton's method is guaranteed to converge, we note that our approximate zero is not necessarily in the basin of attraction of the given root. Another related concept is that of "pseudozero domains", as introduced by Stetter to make a general study of the data to result maps in the context of the Numerical Polynomial Algebra [35]. In particular, we consider only local properties (especially in the stability analysis) specifically aimed at the tasks for our paper.

Let $\epsilon > 0$. From now on, we assume that for each point $\bar{x} \in V$ we are given $x \in A^n(\mathbb{C})$ and $r > 0$, such that $\bar{x} \simeq_\epsilon (x, r)$ holds. Then, we denote by $\widetilde{V}$ the set of all (x, r), and we write $V \simeq_\epsilon \widetilde{V}$.

We now return to the construction given by Definition 5.2.3. Again let $\pi = \pi_{n-1}^n$ be the natural projection from $A^n(\mathbb{C})$ to $A^{n-1}(\mathbb{C})$ which removes the last coordinate. Given two points $\bar{x}$ and $\bar{x}'$ of V we have to decide if they lie in the same π-fiber. Since $\bar{x}$ and $\bar{x}'$ are given by approximate points we need the following.

Definition 5.3.2. *Let i and j be integers such that $1 \leq i \leq j \leq n$. Let $\bar{x}, \bar{y} \in \pi_j^n(V)$. Let $x = (x_1, \ldots, x_j)$ (resp. $y = (y_1, \ldots, y_j)$) and (x, r) (resp. (y, r')) be approximate coordinates of $\bar{x}$ (resp. $\bar{y}$) with tolerance ϵ. We say that $\bar{x}$ and $\bar{y}$ lie approximately in the same π_i^j-fiber with tolerance ϵ if for all $1 \leq k \leq i$ we have*

$$| x_k - y_k | \leq r + r'. \tag{5.3.9}$$

Proposition 5.3.1. *With the notations of Definition 5.3.2, if the points $\bar{x}, \bar{y} \in \pi_j^n(V)$ are in the same π_i^j-fiber, that is, if $\pi_i^j(\bar{x}) = \pi_i^j(\bar{y})$ then, the points $\bar{x}$ and $\bar{y}$ lie approximately in the same π_i^j-fiber with tolerance ϵ.*

Proof. Since $\bar{x}$ and $\bar{y}$ are in the same π_i^j-fiber and suppose (x, r) (resp. (y, r')) are the approximate coordinates of $\bar{x}$ (resp. $\bar{y}$) with tolerance ϵ. Then, for any $1 \leq k \leq i$, this leads to:

$$|x_k - y_k| = |x_k - y_k - \bar{x}_k + \bar{y}_k| \leq |\bar{x}_k - x_k| + |\bar{y}_k - y_k| \leq r + r'. \tag{5.3.10}$$

□

Remark 5.3.2. Suppose $1 \leq i \leq j \leq n$. For the points of $\pi_j^n(V)$, the relation "lying approximately in the same π_i^j-fiber with tolerance ϵ" may not be an equivalence relation, since the transitivity axiom does not hold here. We need to exclude this situation in order to adapt the construction of Definition 5.2.3 for the points of V to approximate points of V. In theory, for exact systems, this situation may be avoided by reducing the tolerance ϵ, and thus the radius r at each point of V. However,

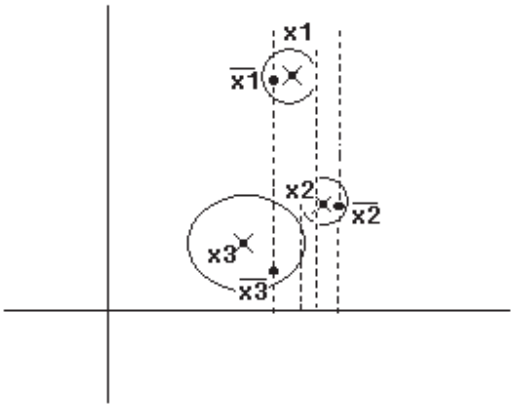

Figure 5.3.1: $\overline{\texttt{x1}}, \overline{\texttt{x2}}, \overline{\texttt{x3}}$ are exact points, `x1,x2,x3` are the approximate points respectively. Here, $\overline{\texttt{x1}}, \overline{\texttt{x2}}$ lie in different fibers, but are approximately in the same fiber, and $\widetilde{V}$ satisfies the weak equivalence condition.

in practice, for some systems it is hard to obtain approximate roots when ϵ is very small. For example, for systems possessing a cluster of points, it can be difficult to compute these roots with high precision [23]. Additionally, for input systems with limited accuracy, a tolerance beyond this limit could not be achieved. So for such systems, we would not be able to meet the requirements of Definition 5.3.4. These precautionary remarks being made, we will propose in Definition 5.3.4 a notion of an *approximate equiprojectable decomposition* of V, where the points of V are given by approximate points in the sense of Definition 5.3.1.

For any zero-dimensional system, using some random linear coordinates change, each fiber has only one point. However, changes of coordinates will generally destroy the sparsity of the original systems. An alternative approach to avoid unfavorable projections is to view a cluster as a perturbed multiple solution (e.g. see the recent work of Szántó et. al. [18]).

Definition 5.3.3. *We say that $\widetilde{V}$ satisfies* the weak equivalence condition with tolerance ϵ *if for all* $1 \leq i \leq j \leq n$*, the relation "lying approximately in the same π_i^j-fiber with tolerance ϵ" is an equivalence relation in $\pi_j^n(V)$. Furthermore, we say that $\widetilde{V}$ satisfies* the strong equivalence condition with tolerance ϵ *if for every $\bar{x}, \bar{y} \in V$ with approximate points $(x, r), (y, r') \in \widetilde{V}$, with tolerance ϵ, for all $1 \leq j \leq n$ the following conditions are equivalent:*

- *we have $\pi_j^n(\bar{x}) = \pi_j^n(\bar{y})$,*
- *the points $\bar{x}$ and $\bar{y}$ lie approximately in the same π_j^n-fiber.*

Here we illustrate Definition 5.3.3 through Figures 1, 2 and 3 where we consider different $\widetilde{V}$'s for the same V. In Figure 5.3.1, the set $\widetilde{V}$ satisfies the weak equivalence

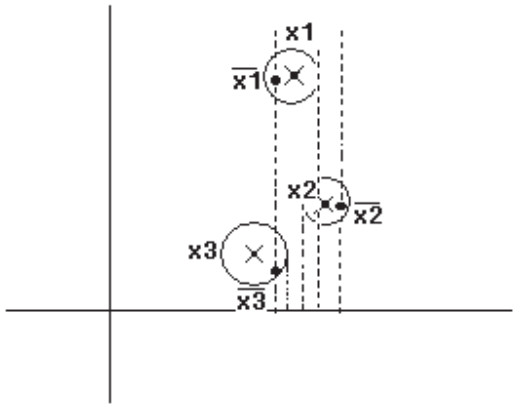

Figure 5.3.2: Refining x3 we get a smaller radius. Here, both pairs $\overline{\texttt{x1}}, \overline{\texttt{x2}}$ and $\overline{\texttt{x1}}, \overline{\texttt{x3}}$ lie approximately in the same fiber, but $\overline{\texttt{x2}}, \overline{\texttt{x3}}$ do not lie approximately in the same fiber. The set $\widetilde{V}$ does not satisfy the weak equivalence condition.

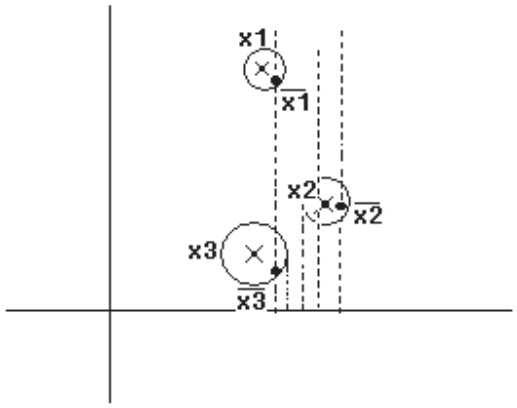

Figure 5.3.3: Refining x1 we get the correct result. Here, $\overline{\texttt{x1}}, \overline{\texttt{x2}}$ lie in different fibers and both weak and strong equivalence conditions for this ϵ are satisified.

condition; observe that $\overline{\texttt{x1}}, \overline{\texttt{x2}}$ lie approximately in the same fiber, but $\overline{\texttt{x1}}$ and $\overline{\texttt{x2}}$ lie in different fibers. In Figure 5.3.2, the points $\overline{\texttt{x1}}, \overline{\texttt{x2}}$ and $\overline{\texttt{x1}}, \overline{\texttt{x3}}$ are pairs of points lying approximately in the same fiber, but $\overline{\texttt{x2}}, \overline{\texttt{x3}}$ do not lie approximately in the same fiber. Hence, in this case, the set $\widetilde{V}$ does not satisfy the weak equivalence condition. In Figure 5.3.3, we refine the three approximate roots until the weak equivalence condition is satisfied again (the strong equivalence condition is also satisfied); we see that $\overline{\texttt{x1}}, \overline{\texttt{x2}}$ lie in the different fibers.

In practice, the "exact" points of V are unknown, so we cannot determine whether the strong equivalence condition is satisfied or not. However, we can detect whether the weak equivalence condition holds or not. In our experiments reported in Section 5.6, however, the exact points are known for each variety V, and we could decide whether or not $\widetilde{V}$ satisfies the strong equivalence condition.

If the weak equivalence condition is satisfied but the strong equivalence condition is not (e.g. see Figure 5.3.1), then there exists two distinct points $\bar{x}, \bar{y} \in V$, with respective approximate points $(x, r), (y, r')$, and an index $1 \leq i \leq n$ such that $\bar{x}_i$ and $\bar{y}_i$ are different but very close to each other; more precisely $|\bar{x}_i - \bar{y}_i| < 2r + 2r'$ holds (generally the distance $|\bar{x}_i - \bar{y}_i|$ will be less than 10^{-13}, see Table 5.4). Due to (for example) roundoff errors in numerical computation, we cannot always avoid these rare cases.

Finally, we note that introducing the notion of "weak equivalence condition" is needed by Definition 5.3.4.

Definition 5.3.4. *Assume that $\widetilde{V}$ satisfies the weak equivalence condition with tolerance ϵ. Define $\pi = \pi^n_{n-1}$. To every point $\bar{x}$ in V, we associate $N(\bar{x})$ the number of points in V which lie approximately in the same π-fiber as x with tolerance ϵ. For all $i \geq 1$, we denote by V_i the set of points $x \in V$ satisfying $N(x) = i$. Then, we split V into a disjoint union $V_1 \cup \cdots \cup V_d$, for some $d \in \mathbb{N}$ large enough. This splitting process is applied recursively to all $V_1, \ldots, V_d$, taking into account the fibers of the successive projections π^n_i, for $i = n-1, \ldots, 1$. In the end, we obtain a family of pairwise disjoint subsets of V, whose union equals V; they form an* approximate equiprojectable decomposition *of V with tolerance ϵ. If this approximate equiprojectable decomposition of V (with tolerance ϵ) consists of only one subset, that is, V itself, we say that V is* equiprojectable *with tolerance ϵ, otherwise the parts of the approximate equiprojectable decomposition of V (with tolerance ϵ) are called* approximate equiprojectable components *of V with tolerance ϵ.*

Note that each approximate equiprojectable component of V is equiprojectable with tolerance ϵ. To each approximate equiprojectable component of V with tolerance ϵ we can associate an *approximate triangular set* by means of Definition 5.3.5. This leads to a notion of an *approximate triangular decomposition* for the variety V.

Definition 5.3.5. *Assume that the zero-dimensional variety V is equiprojectable with tolerance ϵ. Then, by means of the interpolation formulas of Proposition 5.2.2*

one can compute a triangular set $\{N_1, \ldots, N_n\}$ *called an* approximate triangular set *of* V *with tolerance* ϵ.

Now, assume that V *is not approximately equiprojectable with tolerance* ϵ. *A family of approximate triangular sets of approximate equiprojectable components of* V *(with tolerance* ϵ*) forms an* approximate triangular decomposition *of* V, *with tolerance* ϵ.

5.4 Stability Analysis

In this section, we explore the relation between the *relative error* on the coordinates of the approximate points of V and the *relative error* on the interpolated polynomials of the approximate triangular decomposition given by Definition 5.3.5. The coefficients of a polynomial continuously depend on its roots. However, a small error in a root may result in a large error in the coefficients, motivating some of stability analysis.

For the relation between the errors mentioned above to be useful in practice, we must face the following fact: the relative error of a root cannot be computed when the exact root is unknown. In order to overcome this difficulty, for a point $\bar{x}$ of V given by an approximate point (x, r), we view the exact coordinates $\bar{x} = (\bar{x}_1, \ldots, \bar{x}_n)$ as a random variable which takes values in the region defined by the following: for all $1 \leq i \leq n$

$$| x_i - \bar{x}_i | \leq r. \tag{5.4.1}$$

In this paper, we used the word *bias* instead of *relative error* in order to avoid conflicting terminology.

Definition 5.4.1. *For* $\bar{x}, x \in \mathbb{C}$, *we call the the* bias *of* x *w.r.t.* $\bar{x}$ *the fraction*

$$\delta_x = \frac{\bar{x} - x}{x} \tag{5.4.2}$$

simply denoted by δ, *when no confusion may occur.*

Remark 5.4.1. We would like to observe at this point that none of the results of this section require knowledge of the exact coordinates of the points of V. Hence, our results apply also in practice to the situation where V is initially given by a polynomial system with inexact coefficients rather than a polynomial system with exact coefficients. Note that the `PHC` software [38, 24] can process both types of polynomial systems.

We define now the bias for the coefficients of a polynomial. Our definition applies to univariate polynomials as well as to multivariate ones. Let $e = (e_1, \ldots, e_n) \in \mathbb{N}^n$ be an exponent vector. We denote by X^e the monomial $X_1^{e_1} \cdots X_n^{e_n}$ of $\mathbb{C}[X_1, \ldots, X_e]$. We write $p = \sum_{e \in S} f_e X^e$ a polynomial of $\mathbb{C}[X_1, \ldots, X_e]$ with (finite) support S. For every $e \in \mathbb{N}^n$ with $e \notin S$ we set to zero the coefficient f_e, i.e. we define $f_e = 0$. Hence we can simply write $p = \sum_e f_e X^e$.

Typically, in our stability analysis, the polynomial f of Definition 5.4.2 will be a polynomial interpolating the approximate coordinates of the points of V, whereas $\bar{f}$ will be the corresponding polynomial obtained from the exact coordinates of the points of V.

Definition 5.4.2. *Let $\bar{p} = \sum_e \bar{f}_e X^e$ and $p = \sum_e f_e X^e$ be polynomials in $\mathbb{C}[x_1, \ldots, x_e]$. For every $e \in \mathbb{N}^n$, the* bias of coefficient *f_e w.r.t. $\bar{p}$ is defined by*

$$\delta_e = \frac{\bar{f}_e - f_e}{f_e}. \tag{5.4.3}$$

The bias of the polynomial *p w.r.t. $\bar{p}$ is the bias of the coefficient of p w.r.t. $\bar{p}$ which has the largest norm.*

The interpolated polynomials given by Proposition 5.2.2 are multivariate polynomials that are constructed as univariate ones over a suitable coefficient ring. Because of these formulas, we can focus on the univariate case. Let $\bar{p} \in \mathbb{C}[X]$ be a univariate monic polynomial of degree b given by approximate values $x_1, \ldots, x_b$ of its roots with respective radii $r_1, \ldots, r_b$.

$$p = \prod_{i=1}^{i=b} (x - x_i). \tag{5.4.4}$$

Let $\delta_1, \ldots, \delta_b$ be the respective biases of $x_1, \ldots, x_b$ such that the exact roots of $\bar{p}$ are $x_1 + x_1\,\delta_1, \cdots, x_b + x_b\,\delta_b$. Hence we have

$$\bar{p} = \prod_{i=1}^{i=b} (x - x_i - x_i\,\delta_i). \tag{5.4.5}$$

Notation 1. *In the remainder of this section, we assume that $\delta_1, \ldots, \delta_b$ are independent random (complex) variables, each of them with uniform distribution in a disk centered at 0 and with respective radii $r_1/|x_1|, \ldots, r_b/|x_b|$. We define the bias bound and we denote it by ρ the maximum of $r_1/|x_1|, \ldots, r_b/|x_b|$.*

In the proofs of Propositions 5.4.1, 5.4.2, and 5.4.3, we will denote by $O(\delta^2)$ any term in $\delta_i\delta_j$. When ρ is very small, we can ignore such higher order terms and keep only the linear terms.

We will consider the bias of the polynomial $\bar{p}$ w.r.t. p as a random variable denoted by γ. We direct the reader to the Appendix for a brief review of the standard probability results which will be used.

There are essentially three steps in computing the interpolated polynomials of Proposition 5.2.2:

($I1$) compute the univariate polynomials $e_{\alpha,i}$,

($I2$) compute the multivariate polynomials E_α, which are products of univariate polynomials $e_{\alpha,i}$,

(*I*3) compute the multivariate polynomials N_ℓ which are sums of some multivariate polynomials.

For each step, we provide properties on the stability analysis of the corresponding calculations. For our study of the relation between $\bar{p}$ and p, we need the following notation.

Notation 2. For $1 \leq k \leq b$, the k-th elementary symmetric function of $x_1, \ldots, x_b$ is given by

$$\sigma^k = \sum_{1 \leq a_1 < a_2 < \ldots < a_k \leq b} x_{a_1} \cdots x_{a_k}, \tag{5.4.6}$$

and let $\sigma^0 := 1$. Observe that we have:

$$p = \prod_{i=1}^{b} (x - x_i) = \sum_{k=0}^{b} (-1)^k \sigma^k x^{b-k}. \tag{5.4.7}$$

Let $1 \leq j \leq b$. We denote by σ_j^k the element of $\mathbb{C}[x_1, \ldots, x_n]$ obtained from σ^k by specializing x_j to 0, that is $\sigma_j^k = \sigma^k \mid_{x_j=0}$. Let l_j be the j-th Lagrange interpolation polynomial. Observe that we have:

$$l_j = \prod_{i=1, i \neq j}^{b} (x - x_i) = \sum_{k=0}^{b-1} (-1)^k \sigma_j^k x^{b-k-1}. \tag{5.4.8}$$

Proposition 5.4.1. *The bias γ of p w.r.t $\bar{p}$ is bounded by*

$$max \left(\frac{\sum_{i=1}^{b} \mid \sigma_i^k x_i \mid}{\mid \sigma^{k+1} \mid}, k = 0, \ldots, b-1 \right) \rho. \tag{5.4.9}$$

We define

$$\varpi_k = \frac{\sqrt{3 \sum_{i=1}^{b} \mid \sigma_i^k x_i \mid^2}}{3 \mid \sigma^{k+1} \mid} \rho \tag{5.4.10}$$

$$\omega = max(\varpi_k, k = 0, \ldots, b-1). \tag{5.4.11}$$

If b is big enough, then γ is bounded by the normal distribution $\mathrm{N}(0, \omega)$. (For the precise meaning of the statement being bounded by a distribution, please refer to Definition 5.7.1 in the Appendix.)

PROOF. By the definitions of $\bar{p}$ and p, we have

$$\begin{aligned}
\bar{p} - p &= \prod_{i=1}^{b}(x - x_i - x_i\delta_i) - \prod_{i=1}^{b}(x - x_i) \\
&= \prod_{i=1}^{b}(x - x_i) - \sum_{i=1}^{b}\prod_{j=1, j\neq i}^{b}(x - x_j)x_i\delta_i + O(\delta^2) - \prod_{i=1}^{b}(x - x_i) \\
&= -\sum_{i=1}^{b} l_i x_i \delta_i + O(\delta^2) \\
&\approx -\sum_{i=1}^{b}(\sum_{k=0}^{b-1}(-1)^k \sigma_i^k x_i \delta_i) x^{b-k-1} \\
&= -\sum_{k=0}^{b-1}(-1)^k(\sum_{i=1}^{b}\sigma_i^k x_i \delta_i) x^{b-k-1},
\end{aligned}$$

and

$$p = \prod_{i=1}^{b}(x - x_i) = \sum_{k=-1}^{b-1}(-1)^{k+1}\sigma^{k+1}x^{b-k-1}.$$

Thus, the absolute value of the bias for each coefficient γ_k, for $k = 0, \ldots, b-1$, is given by

$$\mid \gamma_k \mid = \frac{\mid \sum_{i=1}^{b}\sigma_i^k x_i \delta_i \mid}{\mid \sigma^{k+1} \mid} \leq \frac{\sum_{i=1}^{b} \mid \sigma_i^k x_i \mid}{\mid \sigma^{k+1} \mid}\rho.$$

Hence, to order $O(\delta^2)$

$$\gamma \leq max\left(\frac{\sum_{i=1}^{b} \mid \sigma_i^k x_i \mid}{\mid \sigma^{k+1} \mid}, k = 0, \ldots, b-1\right)\rho.$$

Recall that, by assumption, the random variables $\delta_1, \ldots, \delta_b$ are independent. Also observe that, to order $O(\delta^2)$, the bias of each coefficient of p is a linear combination of these variables. Hence, we can compute the variance ω_k^2 of the bias γ_k of the coefficient x^{b-k-1} , for $k = 0, \ldots, b-1$, by means of the properties given in the

Appendix:

$$\begin{aligned}
\omega_k^2 &= Var\left(\sum_{i=1}^{b} \sigma_i^k x_i \delta_i / \sigma^{k+1}\right) \\
&= Var\left(\sum_{i=1}^{b} \sigma_i^k x_i \delta_i\right) / |\sigma^{k+1}|^2 \\
&= \frac{\sum_{i=1}^{b} |\sigma_i^k x_i|^2}{|\sigma^{k+1}|^2} Var(\delta_i) \\
&\leq \frac{\sum_{i=1}^{b} |\sigma_i^k x_i|^2}{3|\sigma^{k+1}|^2} \rho^2 \\
&= \varpi_k^2.
\end{aligned}$$

When b is big enough, the distribution of γ_k will tend to a normal distribution $\mathrm{N}(0, \omega_k)$, by the results in the Appendix. Let $\omega = max(\varpi_k, k = 0, \ldots, b-1)$, then γ_k is bounded by $\mathrm{N}(0, \omega)$ for each k. Finally, γ is bounded by $\mathrm{N}(0, \omega)$. □

Remark 5.4.2. If γ follows the normal distribution $\mathrm{N}(0, \omega)$ and $x = 2\omega$ then we have $P(\mid \gamma \mid < x) \approx 0.95$. In fact, our experiments show that for $b \geq 10$, the probability $P(\mid \gamma \mid < x)$ is close to 0.95. Thus we can use Formula (5.4.10) to estimate the bias in the coefficients even if b is not very big. From the output of PHC we can estimate δ using condition numbers, compute ω, and finally estimate the bias for the coefficients with confidence level 0.95. In this section assuming b is big enough, then we have:

Proposition 5.4.2. *Given n univariate polynomials, $p_i(x_i) = \sum_k a_{i,k} x_i^k$, $i = 1, \ldots, n$, if each δ_i (the bias of p_i) satisfies $\mathrm{N}(0, \omega)$, then the bias of $\prod_{i=1}^{n} p_i$ is bounded by $\mathrm{N}(0, \sqrt{n}\omega)$ to order $O(\delta^2)$.*

Proof. Write the product of the univariate polynomials as a sum of monomials :

$$p_1 \cdots p_n = \sum f_e X^e,$$

where

$$f_e = f_{e_1,\ldots,e_n} = a_{1,e_1} \cdots a_{n,e_n}.$$

Denote the exact coefficient by

$$\bar{f}_e = (a_{1,e_1} + a_{1,e_1}\delta_1) \cdots (a_{n,e_n} + a_{n,e_n}\delta_n).$$

By the same arguments as above:

$$\begin{aligned}\gamma_e &= \frac{\bar{f_e} - f_e}{f_e} \\ &= \frac{a_{1,e_1} \cdots a_{n,e_n}(\delta_1 + \cdots + \delta_n)}{a_{1,e_1} \cdots a_{n,e_n}} + O(\delta^2) \\ &\approx \delta_1 + \cdots + \delta_n.\end{aligned}$$

Because each δ_i satisfies $\mathrm{N}(0, \omega)$, their sum is also normally distributed (see the Appendix) with distribution function $\mathrm{N}(0, \sqrt{n}\omega)$. So, to order $O(\delta^2)$ the bias of $\prod_{i=1}^{n} p_i$ is bounded by $\mathrm{N}(0, \sqrt{n}\omega)$. □

Proposition 5.4.3. *Let $p_i(X) = \sum f_{i,e} X^e$, $i = 1, \ldots, N$, be multi-variate polynomials such that δ_i (the bias of p_i) is normally distributed with distribution* $\mathrm{N}(0, \omega)$. *Let*

$$\begin{aligned}\omega_e &= \frac{\sqrt{\sum_{i=1}^{N} f_{i,e}^2}}{|\sum_{i=1}^{N} f_{i,e}|}\omega \\ \omega' &= max(\omega_e).\end{aligned} \tag{5.4.12}$$

Then, to order $O(\delta^2)$, the random variable γ for $\sum_{i=1}^{N} p_i(X)$ is bounded by $\mathrm{N}(0, \omega')$.

Proof. Examine the coefficients of the monomials:

$$\begin{aligned}p_1 + \cdots + p_N &= \sum f_e X^e \\ f_e &= f_{1,e} + \cdots + f_{N,e}.\end{aligned}$$

Let the exact coefficient be denoted by

$$\overline{f_e} = (f_{1,e} + f_{1,e}\delta_1) + \cdots + (f_{N,e} + f_{N,e}\delta_N).$$

Again, by the same arguments, the bias γ_e is:

$$\frac{\overline{f_e} - f_e}{f_e} = \frac{f_{1,e}\delta_1 + \cdots + f_{N,e}\delta_N}{f_{1,e} + \cdots + f_{N,e}} + O(\delta^2).$$

Because each δ_i is normally distributed by $\mathrm{N}(0, \omega)$, the distribution of γ_e is still normal and equal to $\mathrm{N}(0, \omega_e)$ (see the Appendix). So γ for the sum is bounded by $\mathrm{N}(0, \omega')$ (again, see the Appendix for the meaning of bounded here). □

Definition 5.4.3. *Given an approximate triangular set T and the bias bound ρ of the approximate roots, let the* bias *of T be bounded by* $\mathrm{N}(0, \omega)$. *Denote the standard deviation of T by sd where $sd = \omega/\rho$.*

Remark 5.4.3. Let $V \simeq_\epsilon \widetilde{V}$. Assume that $\widetilde{V}$ satisfies the strong equivalence condition with tolerance ϵ, in the sense of Definition 5.3.3. Then, it follows from Propositions 5.4.1, 5.4.2, and 5.4.3 that we can determine *sd* and the bias of the approximate

triangular sets (in the approximate equiprojectable decomposition) of $\widetilde{V}$ with a given probability. Moreover, for an approximate system, given a perturbation of the approximate roots, we can estimate the change of the coefficients of the associated approximate triangular sets.

For further computations, using the approximate triangular sets will likely be difficult because of accumulation of errors. However our discussion above also provides a statistical way to estimate this accumulation.

5.5 An illustrative example

Here, we use a simple example to illustrate our concept of an approximate triangular set and our algorithm for determining the standard deviation. Let us consider:

$$sys = [zx^2 - zy, x^2 - 4y + y^2 + 2, -3zy + zy^2 + 3z - 3]. \tag{5.5.1}$$

The exact triangular set of this system with order $z \prec y \prec x$:

$$[z - 3, y^2 - 3y + 2, x^2 - y]. \tag{5.5.2}$$

1. Solving the system by PHC, we get 4 isolated points:
   $[z = 3.0, y = 2.0, x = 1.41421356237309, rco = 0.01511]$
   $[z = 3.0, y = 1.0, x = 1.0, rco = 0.02089]$
   $[z = 3.0, y = 2.0, x = -1.41421356237309, rco = 0.01511]$
   $[z = 3.0, y = 1.0, x = -1.0, rco = 0.02089]$.
   Here rco is the inverse of the condition number of Jacobian matrix at this point.

2. We remark, as we did in the Introduction, that each solved form $[z = 3.0, y = 2.0, x = 1.41421356237309]$, $[z = 3.0, y = 1.0, x = 1.0]$, $[z = 3.0, y = 2.0, x = -1.41421356237309]$, $[z = 3.0, y = 1.0, x = -1.0]$ is an approximate triangular set.

3. We use the condition numbers to estimate d_{max}: $\delta = 1/rco \times 10^{-16} = 6.62 \times 10^{-15}$ and call this the estimated value of ρ. For this example, we know the exact solutions, and the exact distance between roots. In particular ρ should be $\sqrt{2} - 1.41421356237309 = 5.1 \times 10^{-15}$. In practice we don't know the exact solution of the input system, and we only can give an estimated value for ρ. But we need to point out that this estimation works well for many examples. Comparisons are given in next section.

4. By the definition of an approximate equiprojectable decomposition, the projection of the first and third points above are numerically equal since $|2.0 - 2.0| < (2.0/0.01511 + 2.0/0.01511) \times 10^{-16}$.

# roots	# tests	% of trials: rel. err. $> 1\,sd$ (0.32 expected)	% of trials: rel. err. $> 2\,sd$ (0.05 expected)	% of trials: rel. err. $> 3\,sd$ (0.003 expected)
10	1000	0.328	0.0503	0.0168
20	1000	0.312	0.0425	0.0050
30	1000	0.350	0.0579	0.0023
40	800	0.335	0.0517	0.0067
50	500	0.342	0.0474	0.0042

Table 5.1: Experiments for our probabilistic analysis (sd = standard dev., rel. err. is relative error).

Also the projections of the first and second points are not numerically equal since
$|2.0 - 1.0| > (2.0/0.01511 + 1.0/0.02089) \times 10^{-16} = 1.8 \times 10^{-14}$.
In the same way, we get two different projected points $p_1 = (3.0, 2.0)$, $p_2 = (3.0, 2.0)$ on zy-plane, and there are two points on each fiber. The projections of p_1, p_2 onto the z axis is just one point $z = 3.0$. So the variety of sys is approximately equiprojectable. From the cardinality of the fibers, we know that the degree sequence is [1, 2, 2] with respect to the main variables of each polynomial in the triangular set. The degree sequence can be equivalently written as $1 \cdot 2^2$.

5. By formula 5.2.7, we get the approximate triangular set of sys:

$$[-.999999999999986y + 1.0x^2, y^2 - 3.0y + 2.0, z - 3.0]. \tag{5.5.3}$$

The biggest relative error of coefficients is 1.4×10^{-14}. By formula (5.4.10) and (5.4.12) the standard deviation (sd) is 2.89.
So $sd \times \rho = 1.9 \times 10^{-14} > 1.4 \times 10^{-14}$ is a good estimate for the relative error. In the next section we will give more nontrivial examples to support our statement. Due to both input and round off errors in numerical computation, there will be some monomials of approximate triangular sets with very small coefficients that do not appear in the exact triangular sets. Then the biggest relative error of coefficients is 1. So in practice we will consider coefficients which are smaller than a given tolerance as 0.

5.6 Experimental Results

We have conducted two sets of experiments. The first one illustrates the probabilistic analysis of Proposition 5.4.1. Experiments are described in Section 5.6.1, and the results appear in Table 5.1.

Sys	Name	n	d	h	H	$\widehat{H}$	Reference
1	Issac97	4	2	2	71	1498	[37]
2	L3	3	3	1	1	1678	[2]
3	Sendra	2	7	7	59	2421	[37]
4	fabfaux	3	3	13	72	2650	[14]
5	L4	3	4	1	2	3977	[2]
6	Cylohexne	3	4	3	9	4361	[37]
7	Weispfenning94	3	5	0	10	7392	[37]
8	UteshevBikker	4	3	3	88	7908	[37]
9	Fee-1	4	2	2	34	23967	[37]
10	Reimer-4	4	5	1	14	56013	[37]
11	$S9_1$	8	2	2	33	58116	[37]
12	eco6	6	3	0	12	105718	[37]
13	Geneig	6	3	2	82	114466	[37]
14	gametwo5	5	4	8	674	158075	[37]
15	dessin-2	10	2	7	436	360596	[37]
16	eco7	7	3	0	26	387754	[37]
17	Methan61	10	2	16	227	452756	[37]

Table 5.2: Input systems (n =# polys.; d = degree system; h = height input coeffs; H = height output coeffs; $\widehat{H}$ = estimated height output coeffs.).

Sys	Exact equiproj dec. tim. (secs)	Degree configuration	#$\mathbb{C}$-roots	Time to isolate $\mathbb{R}$-roots (secs)	#$\mathbb{R}$-roots
1	164	$16\ 1^3$	16	< 1	0
2	< 1	(1 3 1), (8 1 1), (8 2 1)	27	< 1	5
3	33	46 1	46	5	6
4	28	$27\ 1^2$	27	1	3
5	1	(24 2 1), (16 1 1)	64	< 1	8
6	6	(4 1 2), (8 1 1)	16	< 1	12
7	72	$54\ 1^2 1$	54	< 1	0
8	29201	$36\ 1^3$	36	7	10
9	24	$26\ 1^3$	26	2	6
10	10097	$18\ 2\ 1^2$	36	5	4
11	26	$10\ 1^7$	10	1	4
12	50	$16\ 1^5$	16	< 1	4
13	18	$10\ 1^3$	10	2	10
14	24320	$44\ 1^4$	44	45	12
15	527	$1\ 42\ 1^8$	42	15	1
16	2742	$32\ 1^6$	32	4	8
17	6251	$27\ 1^9$	27	28	13

Table 5.3: Exact equiprojectable triangular decomposition with the `RegularChains` library.

Sys	#ℂ-roots	#ℂ-roots by PHC	PHC tim.(secs)	Estimated ρ	Exact ρ
1	16	16	1	0.448e-14	0.239e-14
2	27	27	1	0.186e-14	0.337e-14
3	46	46	4	0.159e-11	0.274e-14
4	27	27	2	0.224e-14	0.154e-14
5	64	64	1	0.143e-14	0.331e-14
6	16	16	< 1	0.835e-14	0.181e-14
7	54	49	5	0.183e-13	0.336e-14
8	36	36	6	0.767e-12	0.781e-14
9	26	26	5	0.229e-11	0.759e-14
10	36	36	3	0.739e-13	0.544e-14
11	10	10	3	0.107e-13	0.125e-14
12	16	16	3	0.292e-13	0.287e-14
13	10	10	2	0.629e-13	0.105e-13
14	44	43	6	0.665e-12	0.144e-13
15	42	41	11	0.585e-7	0.271e-14
16	32	32	14	0.760e-13	0.264e-14
17	27	13	10	0.846e-6	0.563e-13

Table 5.4: Approximate roots by `PHC` where the *estimate* ρ = condition number $\times 10^{-16}$ and *exact* ρ = largest 2-norm of distance between exact and approx root divided by the 2-norm of approx root.

Sys	sd	Exact $\rho \cdot sd$	δ_{coeff}	$< sd$?	$< 2sd$?	Residual
1	403.3	0.9639e-12	0.197e-12	yes	yes	0.444e-15
2	7.492	0.2529e-13	0.211e-13	yes	yes	0.125e-13
3	1729.2	0.4736e-11	0.542e-11	no	yes	0.89e-11
4	1056.7	0.1625e-11	0.463e-12	yes	yes	0.201
5	59188.4	0.1959e-09	0.248e-09	no	yes	0.555e-7
6	23835.5	0.4314e-10	0.179e-11	yes	yes	0.7e-13
7	NA	NA	NA	NA	NA	NA
8	383.8	0.2996e-11	0.942e-12	yes	yes	0.163e-8
9	151.6	0.1151e-11	0.181e-12	yes	yes	0.504e-13
10	3928.4	0.2137e-10	0.397e-12	yes	yes	0.193e-18
11	45.77	0.5708e-13	0.133e-13	yes	yes	0.188e-15
12	121.7	0.3488e-12	0.184e-12	yes	yes	0.216
13	551.7	0.5815e-11	0.761e-13	yes	yes	0.314e-17
14	NA	NA	NA	NA	NA	NA
15	NA	NA	NA	NA	NA	NA
16	317.7	0.8397e-12	0.154e-11	no	yes	0.218e20
17	NA	NA	NA	NA	NA	NA

Table 5.5: Approximate Triangular Sets: sd = standard dev. defined in Section 5.4; *exact* ρ = largest 2-norm distance between exact and approx root divided by the 2-norm of approx root; δ_{coeff} = largest relative error of coeffs of approx triangular set compared with the exact one.

The second set of experiments deals with the computation of exact and approximate triangular decompositions. Section 5.6.2 presents the exact case whereas Section 5.6.3 reports on the approximate one. Most of the test polynomial systems that we use (see Table 5.2) are well known problems [2, 11, 37]. They are zero-dimensional square systems defined by multivariate polynomials over $\mathbb{Q}$ generating radical ideals. Table 5.3 shows data for the exact triangular decompositions of these systems, the output by `PHC` is collected in Table 5.4, and Table 5.5 shows their approximate triangular decompositions computed from the `PHC` output. The main results for the purposes of this paper are given by this latter table.

5.6.1 Normal distribution test

Let b be a number of roots given in the column *# roots*. We randomly generate b roots, and view them as the exact roots of a polynomial $\bar{p}$ of degree d. Then, we perturb each of these roots by a uniformly distributed random variable, leading to an approximate polynomial p. The two polynomials $\bar{p}$ and p are expanded in order to obtain ε, the largest relative error for a coefficient. We compute the standard deviation sd by formula (5.4.10), and compare it with ε. These experiments are repeated many times (between 500 and 1000, see the column *# tests*) for $b = 10, 20, 30, 40, 50$. The third column is the percentage of times for which the relative error is bigger than one standard deviation. If the relative error is normally distributed, then this percentage should be 0.32, which we verify in our tests.

5.6.2 Exact triangular decomposition

The test polynomial systems are given in Table 5.2. For each input system F, we give n the number of variables, d the total degree of F, the logarithm h of the largest coefficient, the number of digits H appearing in the largest coefficient in the (exact) equiprojectable decomposition of F, and the height $\widehat{H}$ of that coefficient as estimated by the formulas of [11].

In order to compute the exact equiprojectable decomposition, we use the triangular decomposition library `RegularChains` written in MAPLE by Lemaire, Moreno Maza and Xie [22] in which the algorithms of [25, 11]are implemented. Our computations are done on a 2799 MHz Pentium 4 machine. The timings for computing the exact equiprojectable decompositions are given in the first column of Table 5.3. To understand these timings, we should mention that the `RegularChains` code is high-level interpreted code (and not compiled). Moreover, this code is not supported by fast arithmetic, such as FFT-based arithmetic.

Each degree configuration specifies the degree sequences of the triangular sets in the decomposition (see [2] for similar data). Hence, the number of sequences in a degree configuration equals the number of equiprojectable components of the system. In Table 5.3, #$\mathbb{C}$-roots and #$\mathbb{R}$-roots are, respectively, the total number of complex

and real roots of the system. The column labeled "Time to isolate $\mathbb{R}$-roots", gives the total time in seconds to isolate all the real roots to a precision of 2^{-30} using interval arithmetic.

We have also isolated each complex root. This was done by Éric Schost (École Polytechnique, France) using `Magma` as follows. First, the *splitting circle* method of Schönhage was used to separate the complex roots. Then, Newton iteration was used to refine the isolation boxes. A precision of 200 digits could be achieved for our 17 test systems in less than 10 minutes on a Pentium P3 running at 1GHz.

5.6.3 Approximate triangular sets

We used the `PHC` package [38, 24] to compute the approximate isolated roots for each benchmark system. Then we interpolated the approximate triangular sets and give the results of our error analysis for each system. The computations in Tables 5.4 and 5.5 were done on a 1.5 GHz Pentium M machine, and the timings for finding the roots using `PHC` are listed in *PHC Timing* of Table 5.4. In Table 5.4: the first column is the exact number of roots and second column is the number of roots found by `PHC`. For some systems, `PHC` (in black box mode) did not get every root. This simply means that the default settings in the black box version of `PHC` did not solve the system. We did not compute the approximate triangular sets for such systems. Some of these systems could certainly have been solved by using PHCPack, by exploiting the flexibility of its powerful user specified options, designed for more challenging problems. But we did not do that here. The *estimate* ρ is defined as the condition number $\times 10^{-16}$, and *exact* $\rho = max(|x_i - \bar{x}_i|/|x_i|), \bar{x}_i \in V$ where the $\bar{x}_i$ are the "exact" roots, the x_i are the roots given by `PHC`, and the distance is given by the 2 norm. The results show that our estimated distance is often larger than the exact distance.

In Table 5.5: The second column gives the standard deviation of the approximate triangular set, as discussed in Remark 5.4.3. The third column is the product of the exact ρ and one standard deviation. In the fourth column δ_{coeff} is the largest relative error of the coefficients of the approximate triangular set as compared with the exact one. If this relative error is less than exact $\rho \cdot sd$, the element of the fifth column (labeled $< sd$?) is "yes", otherwise it is "no". Moreover, for every approximate triangular set, the relative error is bounded by $2\,sd$ (see column 6). The last column, labeled *residual*, gives the maximum residual of an approximate triangular set at the roots given by `PHC`. The results of this table support the conclusions of Remark 5.4.3.

5.7 Discussion

There are well-developed algorithms for computing exact triangular decompositions and considerable recent improvements in their time complexity [11]. Such representations are desirable, not only because of their triangular solved-form structure, but

also because, in comparison to other exact methods, their space complexity is well controlled [12]. In particular, they use the minimum number of polynomials needed to describe the equi-dimensional components of a polynomial system.

We have extended such methods to approximate systems in the dimension zero case. We have exploited methods from the newly developing area of Numerical Algebraic Geometry [31, 38, 32, 33], together with new techniques based on the so-called equiprojectable decomposition [10] of a zero-dimensional variety.

Throughout this paper we have assumed that the input is zero-dimensional and generates a radical ideal. We briefly discuss the situation where both of these restrictions are removed. The approximate methods in [33, 38] yield isolated points, possibly of higher multiplicity, corresponding to the zero-dimensional equi-dimensional components. Such multiplicities can be removed (deflated) numerically using the techniques of [13] and [23] (see [21] for a symbolic method for the exact case) and subsequently the methods of our paper can be applied.

Our contribution, in the zero-dimensional case, has been to show that the isolated points, given by approximate coordinates, can be interpolated in order to obtain a triangular decomposition which is an approximation of the exact equiprojectable decomposition. The methods [32] yield a numerical irreducible decomposition for this case, and in particular they give a collection of triangular sets, each of them corresponding trivially to an isolated point.

In addition, the co-dimensional one components (hypersurfaces) can be numerically interpolated by [31, 32] to obtain a single polynomial which can be considered as a representation with triangular shape. The methods also give (non-triangular) representations of all of the positive dimensional components using generic points on each component. The above results, together with those in our paper on linearized triangular decompositions [27], represent progress on the general problem of obtaining approximate triangular representations for all components of a given polynomial system.

Often, in applications, polynomial systems have parameters [33]. One is interested in behavior at generic values of the parameters. In practise, one proceeds by selecting generic values for the parameters, and this is often how zero-dimensional polynomial solving arises in applications. In [33], it is shown how once a solution is computed by homotopy continuation for a specific parameter value, then solutions for other parameter values can be obtained efficiently from the given one using a "parameter homotopy". Analogously, we can follow this idea to reduce positive dimensional systems to zero-dimensional ones by setting generic values for the parameters. Then, a parameter homotopy is used to efficiently compute approximate triangular sets for other parameter values. A promising approach to construct triangular sets of positive dimensional components is to use parameter homotopies followed by interpolation by choosing sufficiently many values for the parameters. Thus, our work on the zero-dimensional case is a preparation for the study of the general case. The related exact approaches go back to [39, 16] among others; also see the recent work [9, 29].

Under some choices of interpolation points (e.g. uniformly spread points) the interpolation formulas of [12] may be ill-conditioned [4, 17]. In the zero-dimensional case, we have no control over this, since the locations of the points are fixed. However, the stability analysis of our paper can identify this situation. In particular, a very large standard deviation means that the coefficients are very sensitive to changes in the roots. For such systems, interpolation is not a good method for obtaining approximate triangular sets from the roots.

In [6], the authors compute an exact absolute factorization of a bivariate polynomial from an approximate factorization. It is natural to ask if one could compute an exact equiprojectable decomposition from an approximate one. One preliminary answer is as follows. Let F be an (exact) zero-dimensional polynomial system in $\mathbb{Q}[X_1 \prec \cdots \prec X_n]$ with total degree d and the maximum number of digits of the coefficients h. Then [11], the height of any coefficient of any (exact) triangular set in the equiprojectable decomposition of $V(F) \subseteq A^n(\mathbb{C})$ is $O(h\, n\, d^n)$. This suggests that the numbers d and n must be small for this *reconstruction* (from approximate to exact) to be realistic. However, the question remains open for future work. Indeed, Table 5.2 shows that the actual coefficient size H in the triangular set is much less than the above height upper bound $\widehat{H}$. Another approach is to lift to nearby exact triangular systems which may have moderately sized rational coefficients, in comparison to lifting to exact rational triangular systems. In addition, a linearized sensitivity analysis should yield information on coefficient versus solution changes (e.g. see [35]). This information is valuable in lifting exact results from the approximate triangular decomposition. Such approaches are the topic of future work.

Traditional uses of exact triangular sets include finding the reduced or simplified form of a polynomial with respect to a triangular decomposition, as accomplished by a chain of pseudo-reductions. Standard deviations of the coefficients also provide information about the accumulation of error in such operations. Provided that the chains of reductions are short, and the degrees of the polynomials involved are not too high, some similar uses are possible with our approximate triangular systems. However, we caution the reader that the accumulation of roundoff error means that such operations should be carried out with care.

The roots of a generic zero-dimensional system are equiprojectable and correspond to a normalized triangular set. Following the idea in [35], we can construct a homotopy to study the deformations of triangular sets with special shape (by the *Shape-Lemma*) and the errors in the roots caused by errors in the coefficients. This idea will also be pursued in future work.

Finally, we direct the reader to [34, 35], where fundamental theorems on backward error analysis and sensitivity of the roots under small perturbations of the coefficients are given for polynomials. When the input system F is approximate, although discontinuous phenomena can occur, some continuity aspects are preserved under perturbation [36].

The favorable properties of the equiprojectable decomposition of $V(F)$ under

specialization [11] suggests that the continuity of approximate equiprojectable decomposition needs to be studied in future application to general systems.

Acknowledgement

GR and WW gratefully acknowledge support for this work from the NSF funded IMA Thematic Year on Applications of Algebraic Geometry. All of the authors acknowledge support from the National Sciences and Research Engineering Council of Canada.

The authors would like to thank Yuzhen Xie (University of Western Ontario, Canada) who realized the experiments with the `RegularChains` library [22]. And we also appreciate the following colleagues for their assistance: François Lemaire (University of Lille 1, France) who provided the source code of *Realroots* to compute the real solutions with the `RegularChains` library and Éric Schost (École Polytechnique, France) who isolated each complex root by `Magma`.

Appendix - Brief review of probability theory

In our stability analysis of coefficients, a probability model was introduced. Here we give a brief review of the relevant standard probability knowledge required.

- If δ is a random variable and c is a constant in $\mathbb{R}$ then $Var(c\delta) = c^2 Var(\delta)$.

- If $\delta_1, ..., \delta_b$ are random variables and $\xi = \sum \delta_i$ then the expectation value is additive: $E(\xi) = \sum E(\delta_i)$. Moreover, if they are independent, then the variance of the sum of these random variables is also additive: $Var(\xi) = \sum Var(\delta_i)$.

- Suppose $\delta = \delta_{re} + \delta_{im}\sqrt{-1}$ and δ_{re}, δ_{im} are independent random variables with the same distribution with $c \in \mathbb{C}$. Then $Var(\Re(c\delta)) = |c|^2 Var(\delta_{re}) = Var(\Im(c\delta)) = |c|^2 Var(\delta_{im})$, where $\Re(z)$ and $\Im(z)$ are the real and imaginary parts of z. In this paper we define $Var(\delta) := Var(\delta_{re})$.

- $\mathrm{N}(0,1)$ is the standard normal distribution with mean 0, standard deviation 1, probability density function $p(x) = \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-x^2/2}$ and cumulative density function $\Phi(x) = \int_{-x}^{x} p(x)dx$. Note that $\Phi(1) \approx 0.68$, $\Phi(2) \approx 0.95$.

- Suppose that $\delta_1, ..., \delta_b$ are independent random variables with distribution functions $F_1, ..., F_b$ and $E(\delta_i) = 0$, $0 < Var(\delta_i) < \infty$, $s_b^2 = \sum Var(\delta_i)$. The Lindeberg condition for a sum of independent random variables is that for any $t > 0$:

$$\frac{1}{s_b^2} \sum_{k=1}^{b} \int_{|x| > t s_b} x^2 dF_k(x) \longrightarrow 0 \quad \text{when} \quad b \longrightarrow \infty \tag{5.7.1}$$

From our assumptions about the roots, the bias is uniformly distributed and because $0 < Var(\delta_i) < \infty$ we have $s_b^2 \to \infty$ as $b \longrightarrow \infty$. So for any $t > 0$, there always exists L, when $b > L$ the integral above is 0.

Proposition 5.7.1 (uniform distribution and Lindeberg condition)**.** *If $\delta_1, ..., \delta_b$ are independent random variables which are uniformly distributed, and $E(\delta_i) = 0$, if the variance of each δ_i is nonzero and finite, then this family of random variables satisfies the Lindeberg condition.*

Proposition 5.7.2 (Lindeberg's central limit theorem [30])**.** *Suppose $\delta_1, ..., \delta_b$ are uniformly distributed independent random variables, $E(\delta_i) = 0$ and δ_i satisfies the Lindeberg condition. Let $S_b = \sum_{i=1}^{b} \delta_i$ and $s_b^2 = \sum_{i=1}^{b} Var(\delta_i)$ then when $b \longrightarrow \infty$, the sum of variables divided by its standard deviation is convergent (in distribution) to a standard normal distribution:*

$$\frac{S_b}{s_b} \longrightarrow \mathrm{N}(0,1) \quad as \;\; b \longrightarrow \infty \tag{5.7.2}$$

Definition 5.7.1. *We say a random variable ξ or $|\xi|$ is bounded by $\mathrm{N}(0,\omega)$ if the probability $P(|\xi| < x\omega) > \Phi(x)$.*

When ω is bigger, the probability will also be bigger. In particular if $\omega' > \omega$ then $P(|\xi| < x\omega') > P(|\xi| < x\omega)$, so ξ is also bounded by $\mathrm{N}(0,\omega')$.

Bibliography

[1] P. Aubry, D. Lazard, and M. Moreno Maza. *On the theories of triangular sets.* J. Symb. Comp., 28(1,2):45–124, 1999.

[2] P. Aubry and M. Moreno Maza. *Triangular sets for solving polynomial systems: A comparative implementation of four methods.* J. Symb. Comp., 28(1-2):125–154, 1999.

[3] P. Aubry and A. Valibouze. *Using Galois ideals for computing relative resolvents.* J. Symb. Comp., 30(6):635–651, 2000.

[4] J.P. Berrut and L.N. Trefethen. *Barycentric lagrange interpolation.* SIAM Rev., vol 46(3): 501-517, 2004.

[5] L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and real computation.* Springer, New York, 1997.

[6] G. Chèze and A. Galligo. *From an approximate to an exact absolute polynomial factorization.* Technical report, Universiteé de Nice, 2005.

[7] S.C. Chou. *Mechanical Geometry Theorem Proving..* D. Reidel Publ. Comp., Dordrecht, 1988.

[8] Robert M. Corless, Mark Giesbrecht, Mark van Hoeij, Ilias S. Kotsireas, Stephen M. Watt. Towards factoring bivariate approximate polynomials. In ISSAC'01: 85-92, 2001.

[9] X. Dahan and X. Jin and M. Moreno Maza and É Schost. *Change of Ordering for Regular Chains in Positive Dimension. Maple Conference*, pages 26–34, Canada 2006.

[10] X. Dahan, M. Moreno Maza, É. Schost, W. Wu, and Y. Xie. *Equiprojectable decompositions of zero-dimensional varieties.* In ICPSS, pages 69–71. University of Paris 6, France, 2004.

[11] X. Dahan, M. Moreno Maza, É. Schost, W. Wu, and Y. Xie. *Lifting techniques for triangular decompositions.* In ISSAC'05, pages 108-115, ACM Press 2005.

[12] X. Dahan and É. Schost. *Sharp estimates for triangular sets.* In ISSAC 04, pages 103–110. ACM Press, 2004.

[13] Barry H. Dayton and Zhonggang Zeng. *Computing the Multiplicity Structure in Solving Polynomial Systems.* In ISSAC 05, pages 116–123. ACM Press, 2005.

[14] European Commission. *FRISCO - A Framework for Integrated Symbolic/Numeric Computation.* Esprit Scheme Project No. 21 024, 1996. `http://www.nag.co.uk/projects/FRISCO.html`.

[15] X.S. Gao and Y. Luo. *A characteristic set algorithm for difference polynomial systems.* Int. Conf. Poly. Sys. Sol., Paris, November 2004, pages 24–26.

[16] P. Gianni, B. Trager and G. Zacharias. *Gröbner bases and primary decomposition of polynomial ideals.* J. Symbolic Computation, 6(2-3):149–167, 1988.

[17] N.J. Higham. *The numerical stability of barycentric lagrange interpolation.* IMA Journal of Numerical Analysis, 24(547-546), 2004.

[18] Itnuit Janovitz-Freireich, Lajos Rónyai and Ágnes Szántó. *Approximate radical of ideals with clusters of roots.* In ISSAC 06, pages 146–153. ACM Press, 2006.

[19] M. Kalkbrener. *A generalized euclidean algorithm for computing triangular representations of algebraic varieties.* J. Symb. Comp., 15:143–167, 1993.

[20] D. Lazard. *Solving zero-dimensional algebraic systems.* J. Symb. Comp., 13:117–133, 1992.

[21] G. Lecerf. *Quadratic Newton iteration for systems with multiplicity.* Found. Comput. Math., 2:247–293, 2002.

[22] F. Lemaire, M. Moreno Maza, and Y. Xie. *The* `RegularChains` *library. Maple Conference*, pages 355–368, Canada, 2005.

[23] Anton Leykin, Jan Verschelde, and Ailing Zhao. *Evaluation of jacobian matrices for Newton's method with deflation to approximate isolated singular solutions of polynomial systems.* In SNC 2005 Proceedings, pages 19–28.

[24] A. Leykin and J. Verschelde. *Phcmaple: A maple interface to the numerical homotopy algorithms in phcpack.* In proceedings of ACA'04, pages 139–147, University of Texas at Beaumont, USA, 2004.

[25] M. Moreno Maza. *On triangular decompositions of algebraic varieties.* Technical Report 4/99, NAG, UK, Presented at the MEGA-2000 Conference, Bath, UK, 2000. `http://www.csd.uwo.ca/~moreno.`

[26] M. Moreno Maza and G. Reid and R. Scott and W. Wu. *On approximate triangular decompsoitions I. Dimension zero.* In D. M. Wang and L. Zhi editors. Symbolic-Numeric Computation, page 250-275, Xi'an, China, 2005.

[27] M. Moreno Maza and G. Reid and R. Scott and W. Wu. *On approximate linearized triangular decompositions.* In Symbolic-Numeric Computation, D. M. Wang and L. Zhi editors. Birkhauser, Basel Boston. To appear.

[28] J. F. Ritt. *Differential equations from an algebraic standpoint*, volume 14. American Mathematical Society, New York, 1932.

[29] É. Schost. *Complexity results for triangular sets.* J. Symb. Comp., 36(3-4):555–594, 2003.

[30] A. N. Shiryaev. *Probability.* Springer, 1995.

[31] A.J. Sommese and J. Verschelde. *Numerical homotopies to compute generic points on positive dimensional algebraic sets.* J. Complexity, 16(3):572–602, 2000.

[32] A.J. Sommese, J. Verschelde, and C.W. Wampler. *Numerical decomposition of the solution sets of polynomial systems into irreducible components.* SIAM J. Numer. Anal., 38(6):2022–2046, 2001.

[33] A.J. Sommese and C.W. Wampler. *The Numerical solutions of systems of polynomials.* World Scientific, Singapore, 2005.

[34] Hans J. Stetter. *The nearest polynomial with a given zero, and similar problems.* SIGSAM Bull., 33(4):2–4, 1999.

[35] Hans J. Stetter. *Numerical Polynomial Algebra.* SIAM. Philadelphia, 2004.

[36] Hans J. Stetter and Gunther H. Thallinger. *Singular systems of polynomials.* In ISSAC '98, pages 9–16. ACM Press, 1998.

[37] The symbolicdata project, 2000–2002. `http://www.SymbolicData.org.`

[38] J. Verschelde. *PHCpack: A general-purpose solver for polynomial systems by homotopy continuation.* ACM Transactions on Mathematical Software, 25(2):251–276, 1999.

[39] B.L. Van der Waerden. (2nd). *Modern Algebra.* Frederick Ungar Publishing Co., New York, 1953.

[40] D. M. Wang. *Elimination Methods.* Springer, Wein, New York, 2000.

[41] W. T. Wu. *A zero structure theorem for polynomial equations solving.* MM Research Preprints, 1:2–12, 1987.

Chapter 6

Computing the Rank and Null-space of Polynomial Matrices

Rank and Null-space computation are considered for polynomial matrices i.e. for matrices whose entries are multivariate polynomials. In particular it is shown that several well-known definitions of the rank of a matrix over a polynomial ring are equivalent. An efficient and probabilistic method to evaluate the rank of a polynomial matrix is given. A generalized Sylvester matrix method is presented to compute the null-space of a polynomial matrix numerically by using Singular Value Decomposition (SVD). Null-space bases and syzygy modules of polynomial matrices are discussed. Applications to computation of GCDs of multivariate polynomials and elimination ideals of "quasi-linear" systems are presented in this article.

6.1 Introduction

A classical and important area in linear algebra concerns the solution of linear equations with coefficients in a field (often in $\mathbb{R}$ or $\mathbb{C}$). Rank and null-space computation for matrices are central operations in this area. When the coefficients of a homogeneous systems are multivariate polynomials, the set of all polynomial solutions is called the *syzygy module* of the system (the difference to the null-space is that the null-space is a vector space in rational functions). This subject has been studied for decades by researchers in commutative algebra and various methods such as Gröbner bases have been proposed to compute a basis of the syzygy module [5]. The paper [11] studied the syzygy module using the theory of n-dimensional linear systems.

This paper is more concerned about the problem of rank and null-space basis computation for polynomial matrices. However the connection between the null-space and the syzygy module of a polynomial matrix will be discussed in this paper.

For such problems, much progress on both symbolic (exact) and numerical computation methods has been made. McClellan [12] gave algorithms to compute general solutions of linear equations with polynomial or rational function coefficients by mod-

ular methods. In [21] the problem of computing the rank and a null-space basis of a univariate polynomial matrix is reduced to polynomial matrix multiplication. The authors use Hensel lifting and matrix minimal fraction reconstruction to yield a favorable complexity for symbolic computation. In [25] the authors proposed a numerical method for univariate polynomial matrices to compute the rank by evaluation of several constant matrices and construct null-space bases by Sylvester methods.

In this paper, the method of null-space computation in univariate case [25] is extended to multivariate cases. In addition rank evaluation of a polynomial matrix is reduced to the constant matrix case by choosing one generic (random) point.

In this paper R denotes the polynomial ring $K[x_1, ..., x_s]$, where the field K can be $\mathbb{R}$ (the field of real numbers) or $\mathbb{C}$ (the field of complex numbers). Here R is an integral domain and also a unique factorization domain [7]. In addition $Q(R)$ denotes the quotient field of R (i.e. rational functions in variables $x_1, ..., x_s$).

6.2 The Rank of a Matrix

In this section, first we will review the aspects of matrix theory which are still true over a polynomial ring R. Many properties can be generalized to any commutative ring [1]. Then we discuss a probabilistic method to detect the rank of a polynomial matrix.

Definition 6.2.1. ***[Polynomial Matrix]*** *The set of all $m \times n$ matrices with entries from R will be denoted by $M^{m\times n}(R)$. Each member in $M^{m\times n}(R)$ is called a polynomial matrix over R.*

For $A \in M^{m\times n}(R)$, in this paper $[A]_{ij}$ is used to denote the (i,j)-th entry of A. Often (a_{ij}) is used to denote A. Also the multiplication is defined in usual way: $A \in M^{m\times \ell}$, $B \in M^{\ell\times n}$, then $AB \in M^{m\times n}(R)$ and $[AB]_{ij} = \sum_{k=1}^{\ell} [A]_{ik}[B]_{kj}$.

Let $A = (a_{ij}) \in M^{m\times n}(R)$. The i-th row of A will be denoted by $Row_i(A)$, $Row_i(A) = (a_{i1}, ..., a_{in})$ for $i = 1, ..., m$. Similarly j-th column of A will be denoted by $Col_j(A)$, $Col_j(A) = (a_{1j}, ..., a_{mj})^t$ for $j = 1, ..., n$.

To save the space the following notation is used:

$$A = \begin{pmatrix} Row_1(A) \\ Row_2(A) \\ \vdots \\ Row_m(A) \end{pmatrix} =: (Row_1(A); Row_2(A); ...; Row_m(A)) \tag{6.2.1}$$

Analogously A is partitioned into column vectors as follows:

$$A = (Col_1(A)|Col_2(A)|...|Col_n(A)) \tag{6.2.2}$$

The classical theory of determinants plays an important role in linear algebra which can be generalized to matrices over a polynomial ring. Additionally the Laplace Theorem for determinants is still valid over a polynomial ring [1].

Let us consider the column vectors of a polynomial matrix $A = (\alpha_1|\alpha_2|...|\alpha_n) \in M^{m\times n}(R)$ and assume $y_k \in R$ for $k = 1, ..., n$. If $\sum_{k=1}^{m} y_k\alpha_k = 0^{m\times 1}$ implies $y_k = 0$ for $k = 1, ..., n$, then these vectors are said to be *linearly independent*. Otherwise these vectors are said to be *linearly dependent*. The linear dependence and linear independence of the row vectors of a polynomial matrix are defined similarly.

Definition 6.2.2. ***[Rank]*** *The (column) rank of polynomial matrix $A \in M^{m\times n}(R)$ is the maximum number of linearly independent column vectors of A.*

It is easy to extend this definition to a set of vectors. Let B be a set of polynomial vectors in R^m, we define $\text{rank}(B)$ to be the maximum number of linearly independent vectors of B.

The following theorem is well known.

Theorem 6.2.1. *Let $A \in M^{m\times n}(R)$. Then $\text{rank}(A) = k$ if and only if any $t \times t$ minor of A is zero when $t > k$ and there exists some $k \times k$ nonzero minor.*

Remark 6.2.1. *Define the algebraic rank as the largest t which satisfies the condition in Theorem 6.2.1. This theorem shows that the algebraic rank is equivalent to (geometric) rank defined in this paper (see Definition 6.2.2). Note that this is true if and only if R is an integral domain. In the book [1], the algebraic rank is generalized to arbitrary commutative rings. There is another frequently used definition of rank for linear systems. Over R, it is also equivalent to the two definitions above.*

Theorem 6.2.2. *Let $A \in M^{m\times n}(R)$.*

$$\text{rank}(A) = \max_{p\in K^s} \text{rank}(A_p) \tag{6.2.3}$$

where A_p is matrix A evaluated at point p.

Proof. Assume $\text{rank}(A) = k$. By Theorem 6.2.1, all the $t \times t$ minors of A are zero when $t > k$, they are also zero when evaluated at any point p. Thus $\max_{p\in K^s} \text{rank}(A_p) \leq k$. Since there exist some $k \times k$ nonzero minors which are nonzero polynomials in R. Then there exists a point p such that the nonzero polynomials do not vanish at this point. Consequently $\max_{p\in K^s} \text{rank}(A_p) \geq k$. Hence $\max_{p\in K^s} \text{rank}(A_p) = k$.
□

A probabilistic method shows that the rank evaluation of a polynomial matrix can be reduced to a constant matrix by choosing a random point in K.

Proposition 6.2.1. *For a random point $p_0 \in K^s$, the probability that $\text{rank}(A) = \text{rank}(A_{p_0})$ is 1.*

Remark 6.2.2. *This proposition follows Schwartz-Zippel theorem [16] which is a frequently used tool in probabilistic polynomial identity testing. The result above only requires that the coefficient field contains an infinite number of points. So K could be $\mathbb{Q}$ or even $\mathbb{Z}$. Therefore the (exact) rank of a polynomial matrix is equal to the rank of the matrix evaluated symbolically at some random point (in $\mathbb{Q}$) with probability 1. This method reduces the cost of rank computation dramatically.*

If we apply numerical computation to the rank evaluation at a point, because of round off error the "numerical rank" is more subtle. Sometimes the symbolic result is consistent with the numerical one. However, for some exact matrices, the answers may be different.

EXAMPLE 6.2.1. *Consider an $n \times n$ constant matrix:*

$$A = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \sqrt{99} & 1 & 0 & \cdots & 0 \\ 0 & \sqrt{99} & 1 & \cdots & 0 \\ \vdots & \cdots & \ddots & \cdots & 0 \\ 0 & 0 & \cdots & \sqrt{99} & 1 \end{pmatrix} \tag{6.2.4}$$

Obviously the rank of A is n. But if we replace $\sqrt{99}$ by 9.949874371 then when $n = 10$ the last (smallest) singular value will be 10^{-9}. When $n = 17$, the last singular value will be less than machine epsilon. This matrix has full rank but is very close to a lower rank unstructured matrix in the 2 norm.

As Rob Corless (private communication) pointed out that the structure and symmetry of a matrix also play important roles. Now if we only consider the matrices with bi-diagonal structure, the minimal distance to a lower rank structured matrix is 1. In general it can be difficult to determine the minimal distance to lower rank structured matrices. Arising from diverse applications Structured Low Rank Approximation is hot topic in numerical linear algebra [15, 2, 8].

Now let the diagonal element be x, and the sub-diagonal element be y. If we choose $x, y \in \mathbb{R}$ randomly, then there is 50% chance such that $y > x$. In this case, when n is large enough the numerical rank of A will be $n - 1$. This also unveils a very important fact: although the singular region often has measure 0, the numerically difficult region (close to singular), which depends on the structure of the input, can be unacceptably large for numerical computation.

6.3 Rank of Approximate Polynomial Matrices

In last section, the rank of a polynomial matrix is defined by the maximum number of linearly independent vectors of the matrix. However this definition is not so meaningful for the polynomial matrices with approximate coefficients since a tiny

perturbation on the input almost always leads to a full rank matrix. To adapt the definition to the approximate case, we introduce a concept of ϵ-rank in this section.

Definition 6.3.1. *Let $A \in M^{m\times n}(R)$ and $\epsilon > 0$. The ϵ-rank of A with a given norm is*

$$min\{r : \text{rank}(\bar{A}) = r, \|A - \bar{A}\| < \epsilon\} \tag{6.3.1}$$

which is denoted by $\text{rank}(A, \epsilon)$.

To define a norm of polynomial matrix, we first introduce a norm of a polynomial with monomial basis. Here the norm of a polynomial $f = \sum_{i\in I} c_i x^i$ is defined to be $\sum_{i\in I} |c_i|$. Note that $\|fg\| \leq \|f\| \cdot \|g\|$, so it is *sub-multiplicative*. There many norms to measure a matrix. In this paper the norm of a polynomial matrix is a *Frobenius* norm. More precisely, let $A \in M^{m\times n}(R)$. The norm of A is defined by:

$$\|A\| := \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} \|A_{ij}\|^2} \tag{6.3.2}$$

It is easy to check that this norm is also sub-multiplicative.

By this definition, it is difficult to directly compute the ϵ-rank of an approximate polynomial matrix. In the exact case, the rank at a random point can be used to detect the rank of a polynomial matrix with probability 1. In the approximate case, it also can provide us some information about the ϵ-rank.

Remark 6.3.1. *In the following proof, we have to compute the SVD in $\mathbb{C}$. So the orthogonal matrix in the SVD computation is replaced by a unitary matrix. The truncated SVD is still minimizing the Frobenius norm of the difference between A and $\bar{A}$ in $\mathbb{C}^{m\times n}$, where $\bar{A}$ has a given rank exactly.*

Theorem 6.3.1. *Let $A \in M^{m\times n}(R)$ and $\epsilon > 0$. Suppose $p_0 = (a_1, ..., a_s) \in \mathbb{C}^s$ with $|a_i| = 1$ for each $1 \leq i \leq s$ and $\{\sigma_i\}$ are the singular values of A_{p_0}. Let $\omega_r = \sqrt{\sum_{i=r}^{min(m,n)} \sigma_i^2}$. If $\omega_r > \epsilon > \omega_{r+1}$, then* $\text{rank}(A, \epsilon) \geq r$.

Proof. Suppose $\text{rank}(A, \epsilon) = r' \leq r - 1$. Then there exist $\bar{A} \in M^{m\times n}(R)$ such that $\text{rank}(\bar{A}) = r'$ and $\|\bar{A} - A\| < \epsilon$. Then $\text{rank}(\bar{A}_{p_0}) \leq \text{rank}(\bar{A}) = r'$ and the Frobenius norm of $\bar{A}_{p_0} - A_{p_0}$ is greater than or equal to ω_r since ω_r is the minimal distance to lower rank matrices [6]. So $\epsilon < \omega_r \leq \|\bar{A}_{p_0} - A_{p_0}\|$. Because $p_0 = (a_1, ..., a_s)$ and $|a_i| = 1$, for any polynomial $f(x_1, .., x_s) = \sum_{i\in I} c_i x^i$, it is easy to show that $|f(p_0)| \leq \sum_{i\in I} |c_i| = \|f\|$. Hence $\epsilon < \|\bar{A}_{p_0} - A_{p_0}\| \leq \|\bar{A} - A\|$. This contradicts with the fact that$\|\bar{A} - A\| < \epsilon$. Therefore $\text{rank}(A, \epsilon) \geq r$. □

If A is a constant matrix, it is easy to show that $\text{rank}(A, \epsilon) = r$. However one difficulty is that we may obtain different lower bounds when the rank of a non-constant matrix is evaluated at different points.

EXAMPLE 6.3.1. *Let us consider a polynomial matrix given by approximate data:*

$$\begin{bmatrix} (2\,x-1) & (-1.0088+1.9958\,x+0.0029\,y) & (-1.000091+2.000011\,x-0.00001\,y) \\ (-3\,y+x) & (-0.0041+0.9908\,x-3.0046\,y) & (0.000070+0.999959\,x-3.000045\,y) \\ (y+1) & (1.0093+0.0027\,x+1.0018\,y) & (0.999916-0.000002\,x+1.000067\,y) \end{bmatrix}$$

Suppose we choose $\epsilon = 0.01$. *At* $(x, y) = (-.960 - .279\,i, .529 + .849\,i)$ *the singular values are* $[8.90, 0.00710, 0.000077]$ *yielding the numerical rank* 1. *But the singular values are* $[5.56, 0.016, 0.000074]$ *at* $(x, y) = (.667 + .745\,i, -.178 + .984\,i)$, *where the numerical rank is* 2, *which is a lower bound of* ϵ*-rank. If we let the third column equal the first column, then this new matrix has the distance to the input matrix less than* ϵ *and it has rank* 2 *exactly, which is an upper bound on the* ϵ*-rank. So* ϵ*-rank for this matrix is* 2. *In general it can be difficult to construct a matrix close to the input matrix with rank deficiency.*

The basic reason for failing at the first point is that the distance of this matrix to rank 1 *matrices (let the second and third columns equal the first column, then the distance to the input matrix less than* 0.027*) is only slightly larger than the tolerance* 0.01. *And it is also difficult to know the minimal distance to the matrices with a given rank.*

However for random matrices with some small perturbation, our experiments show that numerical rank with tolerance ϵ at a random point is equal to r.

EXAMPLE 6.3.2. *In the experiments, exact matrices with* n *rows and* r *columns were generated randomly in Maple 10. In particular the entries of the matrices were random real polynomials of* 3 *variables with degree* d *and coefficient range* c. *We used random linear combinations of existing columns and appended them to make a square matrix* A. *Then a full rank random perturbation matrix with norm* ϵ, *denoted by* ΔA *was generated. Obviously the upper bound on the rank of* $A + \Delta A$ *is* r. *We compute the numerical rank* r_- *of* $A + \Delta A$ *with tolerance* ϵ *at a random point* p *by using the SVD, and compare* r_- *with upper rank bound* r.

The experiments were repeated for 1000 *times for each combination of different values of* n, r, d, c *(we choose* $n = 3, 4, .., 10$, $d = 3, 5$, $c = 1, 10$ *and all the possible values of* r *for each* n*). In the experiments,* $r_- = r$ *and no exceptions occurred. The results indicate that numerical rank at a random point can be a good estimate of the* ϵ*-rank of an approximate polynomial matrix.*

As we showed in Example 6.3.1, this method can fail. So after we obtain a lower rank bound r of an input matrix A by Theorem 6.3.1, we need to check whether the method fails or not. If r is underestimated, then the dimension of the null-space will be overestimated. The construction of the approximate null-space is carried out using the Sylvester Method, which is given in Section 6.5. If we cannot construct

the null-space with dimension $n-r$ or the output $\tilde{N}$ (an approximation of the null-space) is not acceptable, it means the norm of the remainder $A \cdot \tilde{N}$ is larger than the tolerance. Then the method fails. So this method is a heuristic way to estimate the ϵ-rank of an approximate polynomial matrix. And more rigorous probability analysis should be studied in the future work.

6.4 Null-space and Syzygy Module

In this section the null-space and syzygy module of a polynomial matrix are discussed.

Given a polynomial matrix $A \in M^{m\times n}(R)$, let

$$Null(A) := \{f \in Q(R)^n | Af = 0^{m\times 1}\} \tag{6.4.1}$$

and call $Null(A)$ the null-space of matrix A. If there is no confusion, it is denoted by N in this paper. Because $Q(R)$ is a field, the null-space N is a vector subspace. The rank-nullity theorem asserts $\text{rank}(N) + \text{rank}(A) = n$ in the usual way.

A related concept in commutative algebra and algebraic geometry [5] is the *syzygy module*. In particular

$$Syz(A) := \{g \in R^n | Ag = 0^{m\times 1}\} \tag{6.4.2}$$

is called the syzygy module of the matrix A.

Because R is an integral domain and by the distributivity of the ring, it is easy to prove that:

Proposition 6.4.1. *Given a nonzero vector $f \in Syz(A)$ with $f = ag$, where $a \in R$, $a \neq 0$ and $g \in R^n$, then $g \in Syz(A)$.*

Now let us consider the syzygy module of a polynomial matrix A. In fact, $\text{rank}(Syz)$ and $\text{rank}(A)$ also have this relation:

Theorem 6.4.1. *Let $A \in M^{m\times n}(R)$. Then*

$$\text{rank}(Syz(A)) + \text{rank}(A) = n \tag{6.4.3}$$

Proof. Assume $\text{rank}(A) = k$, then it is always possible to choose k linearly independent columns from A. Without loss of generality, let $\{Col_1(A), ..., Col_k(A)\}$ be such a set.
Then $\{Col_1(A), ..., Col_k(A), Col_{k+i}(A)\}$ is linearly dependent, where $1 \leq i \leq n-k$. So for each i, there exist $c_{i,1}, ..., c_{i,k}, c_{i,k+i}$ such that $\sum_{j=1}^{k} Col_j(A)c_{i,j} + Col_{k+i}(A)c_{i,k+i} = 0$ and $c_{i,k+i} \neq 0$. Let $f_i = (c_{i,1}, ..., c_{i,k}, 0, ..., 0, c_{i,k+i}, 0, ..., 0)^t \neq 0$, then $Af_i = 0$. It is easy to check that $G := \{f_1, ..., f_{n-k}\}$ is linearly independent and each f_j is in $Syz(A)$. Hence $\text{rank}(Syz(A)) \geq n-k$.
If $\text{rank}(Syz(A)) > n-k$ and $\text{rank}(A) = k$, then by Proposition 6.2.1 there must exist a generic point p, such that $\text{rank}(Syz(A)_p) > n-k$ and $\text{rank}(A_p) = k$. But

this contradicts with Rank-Nullity Theorem for constant matrices.
Therefore $\text{rank}(Syz(A)) + \text{rank}(A) = n - k + k = n$. □.

Given a set of polynomial vectors $F = \{f_1, ..., f_r\}$, define

$$span(F, R) = \left\{ \sum_{i=1}^{r} a_i f_i \in R^n : a_i \in R, f_i \in F \right\} \tag{6.4.4}$$

Then $span(F, R)$ is the set of all linear combinations of F in R. It is the R-submodule of R^n generated by F in R.

Similarly, the linear subspace generated by F in $Q(R)$ is

$$span(F, Q(R)) = \left\{ \sum_{i=1}^{r} c_i f_i \in Q(R)^n : c_i \in Q(R), f_i \in F \right\} \tag{6.4.5}$$

If F is linearly independent, then $span(F, R)$ and $span(F, Q(R))$ are denoted by $\langle F \rangle_R$ and $\langle F \rangle_{Q(R)}$ respectively.

Let N be the null-space of A and $Syz(A)$ be the syzygy module of A. If $span(F, Q(R)) = N$ (or $span(F, R) = Syz(A)$), we call F a set of *generators* of N (or $Syz(A)$). In fact it is possible to choose a set of linearly independent vectors G from F to generate N. In particular G is called a *basis* of the null-space, and $N = \langle G \rangle_{Q(R)}$.

In Commutative Algebra, if a module M has a linearly independent set of generators G ($M = \langle G \rangle_R$) , then G is called a *module basis* and the module is *free module*. It can be difficult to determine whether the syzygy module $Syz(A)$ is free. But the generators of $Syz(A)$ can be obtained by computing the *module Gröbner basis* of A [5]. However it is expensive (it can be of double exponential cost) and unsuitable for inexact input. The reason is that computing the Gröbner basis requires an input order of the variables. This may cause trouble when we determine that the leading coefficient is very small. Another reason that may also cause trouble is numerical polynomial division (to reduce the S-polynomial by existing polynomials) may not be stable even for univariate polynomials.

Suppose $\text{rank}(Syz(A)) = r$. It is always possible to choose $G = \{g_1, ..., g_r\}$ from $Syz(A)$, which is a set of r linearly independent column vectors. It is also easy to see that $\langle G \rangle_R \subseteq Syz(A)$. The set of all the vectors which are linearly dependent on G is also a R-module, denoted by $\langle\langle g_1, ..., g_r \rangle\rangle$. The following proposition shows it is equal to the syzygy module.

Proposition 6.4.2. *Suppose $Syz(A)$ is the syzygy module of a polynomial matrix $A \in M^{m \times n}(R)$ with rank r and $G = \{g_1, ..., g_r\}$ is a set of r linearly independent column vectors of $Syz(A)$. Then $\langle\langle G \rangle\rangle = Syz(A)$.*

Proof. For any $g \in Syz(A)$, the set $\{g, g_1, ...g_r\}$ must be linearly dependent (otherwise $\text{rank}(Syz) > r$). So $Syz(A) \subseteq \langle\langle g_1, ..., g_r \rangle\rangle$.

Conversely, for any $g \in \langle\langle g_1, ..., g_r \rangle\rangle$, there exists a linear combination such that $\sum_{i=1}^{r} a_i g_i + a_0 g = 0$. So $a_0 g = -\sum_{i=1}^{r} a_i g_i \in Syz(A)$. Also linear independence of G implies $a_0 \neq 0$. By Proposition 6.4.1, this implies $g \in Syz(A)$. Hence $Syz(A) = \langle\langle G \rangle\rangle$ □

Suppose A is a polynomial matrix and G is a basis of the null-space of A. Then $\langle G \rangle_R \subseteq \langle\langle G \rangle\rangle = Syz(A) \subseteq \langle G \rangle_{Q(R)} = Null(A)$ and G may not be a module basis of $Syz(A)$ over R. But it is a basis of the null-space over the quotient field $Q(R)$. In this paper "basis" does not mean module basis.

EXAMPLE 6.4.1 ([5] Exercise 25, page 193)**.** *Let $R = K[x, y]$ and consider the matrix $A = (1 + x, 1 - y, x + xy) \in M^{1\times 3}(R)$. Let*

$$f_1 = \begin{pmatrix} 1-y \\ -1-x \\ 0 \end{pmatrix}, \quad f_2 = \begin{pmatrix} x+xy \\ 0 \\ -1-x \end{pmatrix}, \quad f_3 = \begin{pmatrix} 0 \\ x+xy \\ -1+y \end{pmatrix}$$

It can be checked that $F = \{f_1, f_2, f_3\}$ generates the syzygy module $Syz(A)$. But $\text{rank}(A) + rank(Syz) = 3$*, so* $\text{rank}(F) = 3 - 1 = 2$*. This generating set is linearly dependent. If we choose $G = \{f_1, f_2\}$, then* $\text{rank}(G) = 2$ *and $Null(A) = \langle G \rangle_{Q(R)}$. On the other hand $f_3 \in \langle\langle f_1, f_2 \rangle\rangle$. Hence $Syz(A) = \langle\langle f_1, f_2 \rangle\rangle$. Actually by the Quillen-Suslin Theorem [5], $Syz(A)$ is free since the ideal generated by $\{1 + x, 1 - y, x + xy\}$ is the whole polynomial ring. It means there exist g_1 and g_2, such that $Syz(A) = \langle g_1, g_2 \rangle_R$. Such a row is often called unimodular row. Quillen and Suslin showed that the syzygy module of a unimodular row is free.*

6.5 Generalized Sylvester Method

In this section, a numerical method is given to compute the null-space of any polynomial matrix approximately. It is a generalized Sylvester Matrix method together with the SVD.

6.5.1 Sylvester Matrices and the Algorithm

Let $R = K[x]$ be a polynomial ring with variables $x_1, ..., x_s$. There is a natural bijection: $M^{m\times n}(K[x]) \leftrightarrow M^{m\times n}(K)[x]$, where K is the coefficient field of the polynomial ring. Here $M^{m\times n}(K)$ is the set of matrices with entries in K. Hence, equivalently a polynomial matrix can be considered as a polynomial with matrix coefficients, a so-called *matrix polynomial*.

Let $T(d) = \binom{s+d}{d}$ where for notational simplification the parameter s which is number of variables in the polynomial ring is omitted. A polynomial matrix A can be written in terms of increasing total degree order of monomials of x: $A(x) = \sum_{i=1}^{T(d_1)} A_i x^{\alpha_i}$. Here d_1 is the maximum total degree of the entries of A and $T(d_1)$

is maximum number of terms of $A(x)$. Assume $f \in Syz(A)$ with degree d_2, then similarly we obtain $f(x) = \sum_{j=1}^{T(d_2)} f_j x^{\beta_j}$. Hence

$$A(x)f(x) = \sum_{k=1}^{T(d_1+d_2)} C_k x^{\gamma_k} = 0^{m\times 1} \tag{6.5.1}$$

where $C_k := \sum_{\alpha_i+\beta_j=\gamma_k} A_i f_j$. This equation is equivalent to each coefficient $C_k = 0$.

Naturally, the coefficients of $f(x)$ are written as a vector: $v_f := [f_1, ..., f_{T(d_2)}]^t$. It is not hard to find a matrix M_A, a so-called *convolution matrix or generalized Sylvester matrix*, whose entries are the coefficients of $A(x)$, such that

$$M_A^{mT(d_1+d_2)\times nT(d_2)} \cdot v_f^{nT(d_2)\times 1} = 0^{mT(d_1+d_2)\times 1} \tag{6.5.2}$$

To reduce polynomial algebra to linear algebra by choosing certain polynomial basis is a frequently used technique in both symbolic and numerical computations [9, 3].

The relations above are illustrated by the diagram below:

$$A \xrightarrow{\phi} A(x) \xrightarrow{\psi_{d_2}} M_A^{d_2}, \qquad A \xrightarrow{\omega_{d_2}} M_A^{d_2}, \qquad M_A^{d_2} \xrightarrow{\omega_{d_2}^{-1}} A \tag{6.5.3}$$

where ϕ and ψ_{d_2} are bijections as described above, $\omega_{d_2} = \psi_{d_2} \circ \phi$ and $\omega_{d_2}^{-1}$ is the inverse map of ω_{d_2}. Note that this map depends on d_2 the degree of f.

The SVD is used to compute the null-space $N_A^{d_2}$ of the generalized Sylvester matrix (convolution matrix) $M_A^{d_2}$. Each vector in $N_A^{d_2}$ corresponds to a polynomial vector with degree less than or equal to d_2. Let $Syz(A)^{d_2}$ be the set of all the polynomial vectors with degree less than or equal to d_2 in $Syz(A)$. It is easy to show that $Syz(A)^{d_2} = \omega_{d_2}^{-1}(N_A^{d_2})$. But the images of linearly independent vectors in $N_A^{d_2}$ are not necessarily linearly independent in $Syz(A)^{d_2}$.

Let N be the null-space of A and $r = rank(N)$. The process to approximate a basis starts from degree 0 polynomial vectors, $G = \omega_0^{-1}(N_A^0)$. Then the degree d_2 is increased by one at each iteration. For each iteration, it is always possible to choose vectors from $\omega_{d_2}^{-1}(N_A^{d_2})$ which are linearly independent of all the vectors in G. The vectors that are chosen are then appended into G. If G has r vectors, then the process stops and the G is an approximate basis for the null-space.

EXAMPLE 6.5.1.

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & (2\,x+1) \\ 0 & -1 & 0 & (2\,x+1) & (2\,x+1) & 0 \\ -1 & 1 & (2\,x+1) & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6.5.4}$$

The rank of A can be estimated using a random point to obtain $\text{rank}(A) = 4$. *Note that we use random point in $\mathbb{R}$ to detect the rank rather than the ϵ-rank. In this example, when the degree of polynomial vector f is* 0, *there is one vector (column 3) in $Null(A)$, and when the degree is increased to* 1, *another two vectors, the first two columns of the following matrix are obtained.*

$$\begin{bmatrix} 0 & 0 & 0 \\ (-0.3317 - 0.6635\,x) & (0.2093 + 0.4185\,x) & 0 \\ 0.3317 & -0.2092 & 0 \\ -0.543 & -0.4934 & -0.7071 \\ 0.2114 & 0.7027 & 0.7071 \\ 0 & 0 & 0 \end{bmatrix}. \tag{6.5.5}$$

Since $\text{rank}(N) = 2$, *the second and third columns can be chosen from this matrix to form a basis for $Null(A)$.*

Will the algorithm terminate after finitely many steps? Henrion in his Ph.D. Thesis "Reliable Algorithms for Polynomial Matrices" gave a degree bound for such null-space bases. Using the Laplace Theorem a similar result is given here.

Proposition 6.5.1. *Given a polynomial matrix $A \in M^{m\times n}(R)$, let* $\text{rank}(A) = k < n$, *$r = n - k$ and let $deg(Col_i(A))$ be the maximum degree of all the elements in the i-th column of A. The order of columns can always be changed to satisfy $deg(Col_1(A)) \geq deg(Col_2(A)) \geq \cdots \geq deg(Col_n(A))$. Then there exists G which is a basis of the null-space of A, such that*

$$degree(G) \leq d_A = \sum_{i=1}^{k} deg(Col_i(A)) \,. \tag{6.5.6}$$

PROOF. From the proof in Theorem 6.4.1, let $G = \{g_1, ..., g_r\}$ be a basis of the null-space of A. Now it only remains to estimate the degree bound for each g_i. By the Laplace Theorem, each g_i can be constructed from $k \times k$ minors of A whose degrees are bounded by d_A. Hence the degree of G is bounded by d_A. □

This yields Algorithm 6.5.1 to compute a basis of the null-space of a polynomial matrix approximately. The subroutine **Choose** is to choose the independent vectors from $\omega_i^{-1}(N_A^i)$ using rank computation and append them to G one by one. When the input is exact, we can choose a random exact point p to compute the rank of A_p symbolically. Otherwise we apply SVD to compute the numerical rank $\mathbf{rank}(A_p, \epsilon)$. If $\text{rank}(A, \epsilon)$ is underestimated, it means that the number of independent vectors in the null-space is overestimated. If we could not find such vectors within the degree bound d_A or the norm of the remainder $(A \cdot G)$ is larger than the tolerance then the algorithm will return 'FAIL'.

Algorithm 6.5.1. $G = \boldsymbol{NullSpace}(A, \epsilon)$

Input : A, *a polynomial matrix*
ϵ, *tolerance*
Output: G, *a set of column vectors*
fail, when $\text{rank}(A, \epsilon)$ *is underestimated*

$p := random(a_1, .., a_s)$ — *a random point in* $\mathbb{C}^s$ *with* $|a_i| = 1$
$k := \boldsymbol{rank}(A_p, \epsilon)$ — *for exact input, replaced by* $\text{rank}(A_p)$
$r := n - k$
$d := \max\{deg(Col_i(A))\}$
$d_A := \sum_{i=1}^{k} deg(Col_i(A))$ — $d_A \leq (n-1)d$
$G := \{\}$
Repeat from $i = 0$ *to* d_A
  $M_A^i := \omega_i(A)$ — *size of matrix* $L \leq mT(d_A + d)$
  $N_A^i := \boldsymbol{SVD}(M_A^i)$ — *cost:* $O(L^3)$
  $G := \boldsymbol{choose}(\omega_i^{-1}(N_A^i), G)$
  if $|G| = r$ *then*
    if $\|A \cdot G\| < \epsilon$ *then*
      return G
    else
      return fail
    end if
  end if
end loop
return fail.

Now let us review Example 6.3.1.

EXAMPLE 6.5.2 (Detect the underestimation of the ϵ-rank)**.** *When* $\epsilon = 0.01$. *We obtain numerical rank* 1 *at* $(x, y) = (-.960 - .279\,i, .529 + .849\,i)$ *which underestimates the* ϵ*-rank of input matrix* A. *In this case we construct the (approximate) null-space*

N, which has two vectors:

$$\begin{bmatrix} -0.406 & -0.708 \\ 0.815 & 0.00335 \\ -0.412 & 0.705 \end{bmatrix} \tag{6.5.7}$$

However

$$A \cdot N = \begin{bmatrix} -0.008\,x - 0.005 + 0.0023\,y & -0.00029\,x + 0.0000026\,y \\ 0.003\,y - 0.0098\,x - 0.0033 & -0.00009\,y - 0.000073\,x + 0.000035 \\ -0.0007\,y + 0.005 + 0.0022\,x & -0.000035\,y - 0.0001 + 0.0000076\,x \end{bmatrix} \tag{6.5.8}$$

The norm of the remainder is 0.0238 *(> 0.01) which indicates the method fails.*

Given a polynomial matrix A, Algorithm 6.5.1 will find the minimal degree basis of the null-space of A. This basis is not necessarily a module basis of the syzygy module. But if $\text{rank}(N) = 1$, then the output G of this algorithm is a module basis of $Syz(A)$ because R is a *unique factorization domain.*

Proposition 6.5.2. *Given a polynomial matrix $A \in M^{m\times n}(R)$, let $Syz(A)$ be the syzygy module of A and* $\text{rank}(N) = 1$. *Then a polynomial vector $\{f\}$ can be found by algorithm **NullSpace**, which is a module basis of $Syz(A)$. This basis is unique up to multiplication by numbers in K.*

Proof. Let f be the output of **NullSpace**. It is a vector in $Syz(A)$ with the minimal degree otherwise the algorithm will stop earlier and the output is not f. For any $g \in Syz(A)$, the polynomial vectors f and g must be linearly dependent. So there exist $a, b \in R$ such that $GCD(a, b) = 1$ and $af = bg$. Hence every entry of f must have factor b. If $deg(b) > 0$, then f is not the vector with minimal degree. Therefore $deg(b) = 0$ and g is generated by f. This means $\langle f\rangle_R = Syz(A)$. To show the uniqueness, if $\langle g\rangle_R = Syz(A)$, then there exist a, b, $f = bg$ and $g = af$, so $ab = 1$ and $a, b \in K$. □

In general, a basis of the null-space may not be a module basis. In [11], the author proves that the syzygy module being free is equivalent to the existence of a basis of the null-space, which generates a *minor right prime matrix.* The following result provides a method for checking whether or not a basis is a module basis.

Proposition 6.5.3. *Given a polynomial matrix A, let the null-space $Null(A) = \langle g_1, ..., g_r\rangle_{Q(R)}$ and the matrix $G = (g_1|...|g_r)$. If all the $r \times r$ minors of G are relatively prime, then $Syz(A) = \langle g_1, ..., g_r\rangle_R$*

Please see the Proposition 6 in [11] for a proof.

6.5.2 Algorithmic Analysis

Given a polynomial matrix $A \in M^{m\times n}(R)$, if each $deg(Col_i(A)) = d$ and rank of A is k, then $d_A = kd$. So the maximum size of the Sylvester (convolution) matrix M_A is $m\binom{s+d+kd}{s} \times n\binom{s+kd}{s}$. Assume $m \approx n$ and $kd \gg s$. Then the size of this matrix is bounded by $n(k+1)^s d^s$. Since $k < n$, it follows that the bound is $n^{s+1}d^s$. When $s = 1$, the paper [21] reports that the cost to compute the rank and null-space is $\tilde{O}(n^{2.7}d)$ and it is same as the cost of multiplication of matrices. Here $\tilde{O}$ indicates missing logarithmic factors of form $\alpha(\log n)^\beta(\log d)^\gamma$ for three positive real constants α, β, γ. In the numerical case, we know that the Sylvester matrix M_A is always sparse with block Toeplitz structure. Zuniga and Henrion [25] give an algorithm with complexity $O(n^3 d)$ using blocked LQ factorization.

However when $s > 1$, the block Toeplitz structure of M_A is much more complicated. To design an efficient and numerical stable algorithm for computing the null-space basis of multivariate polynomial matrix is an important problem and needs further study.

For random dense matrices, the degree bound and size bound is sharp. But for sparse matrices, the cost may be much less. In Example 6.5.1, the degree bound $d_A = 4$, but a basis is obtained at degree 1. This means when entries of A are sparse polynomials and A is a sparse matrix, all the linear independent polynomial vectors could be found when the degree of f is much lower than the degree bound.

6.6 Applications

6.6.1 Approximate GCD of two multivariate polynomials

As surveyed in [23], GCD-finding is one of the basic operations in algebraic computation with a wide range of applications. For example the multivariate GCD can be applied to engineering problems such as image restoration where the given polynomials contain noise. However the existing symbolic GCD-finders are usually not be suitable for inexact polynomials since GCD computation is infinitely sensitive to perturbations. Therefore, many researchers propose various methods to compute a well-defined GCD and also give the associated error analysis [3, 23, 10]. This is not the main point of the current paper. Only a brief discussion and a simple example (example 3 in [24]) are given here to illustrate the basic idea.

Since R, a polynomial ring over a field, is a unique factorization domain (UFD), any two multivariate polynomials p and q in R have a GCD.

Let $A = (p, q) \in M^{1\times 2}(R)$ and N be the null-space of A. Obviously rank$(N) = 1$, so N has basis $(a; b)$ such that $GCD(a, b) = 1$ and $ap = -bq$. This vector can be computed by the algorithm **NullSpace**. Again let $B = (p, b)$. The method yields a basis of $Null(B)$, which is $(1; g)$. Alternately a least squares method can be used to obtain g. Hence $g = -\frac{p}{b} = \frac{q}{a} = GCD(p, q)$. Note that numerical division of

polynomials may not be stable.

EXAMPLE 6.6.1. *For $n = 10, 20, 30, 40$, let u_n be a dense bivariate polynomial of degree n with random coefficients in $\mathbb{R}$. Let $p = u_n(1 + x + y + xy)$ and $q = u_n(1 - x + y - xy + y^2)$. So with probability 1, $GCD(p, q) = u_n$. First construct a matrix $A = (p, q) \in M^{1\times 2}(K[x, y])$ and then compute a basis by **NullSpace**:*

$$\begin{pmatrix} -0.3333 - 0.3333\,y + 0.3333\,x - 0.3333\,y^2 + 0.3333\,xy - 8.326 \times 10^{-17}\,x^2 \\ 0.3333 + 0.3333\,y + 0.3333\,x + 3.122 \times 10^{-17}\,y^2 + 0.3333\,xy + 1.387 \times 10^{-17}\,x^2 \end{pmatrix}$$

Then let $B = (p, b)$, where $b = 0.3333 + 0.3333\,y + 0.3333\,x + 0.3333\,xy$ and compute a basis of $Null(B)$. This numerical method obtains a good approximation of the exact GCD of p, q, which is expected to be u_n. In particular, when $n = 40$ the largest matrix to process in our algorithm is 1034×12.

6.6.2 Projection of the Variety of Quasi-linear Polynomial Systems

A special class of polynomial systems, so-called *quasi-linear polynomial systems* is considered. In [17], such equations are called *parametric linear systems*. These systems have variables $\{x_1, ..., x_s, y_1, ..., y_n\}$, and can be written as $AY + \mathbf{b} = 0$, where $A \in M^{m\times n}(\mathbb{C}[x_1, ..., x_s])$, $Y = [y_1, ..., y_n]^t$ and $\mathbf{b} \in M^{m\times 1}(\mathbb{C}[x_1, ..., x_s])$ is a column polynomial vector. The matrix A, which contains the key information, is called the coefficient matrix of the quasi-linear polynomial systems.

Let $I = \langle Row_1(A) \cdot Y + \mathbf{b}_1, ..., Row_m(A) \cdot Y + \mathbf{b}_m \rangle \subseteq \mathbb{C}[x_1, ..., x_s, y_1, ..., y_n]$ and $V = V(I)$. We consider the projection of the variety into X space: $V_x = \pi_x(V)$ and the elimination ideal $I_x = I \cap \mathbb{C}[x_1, ..., x_s]$. It is well known that $V(I_x) = \overline{V_x}$.

The projection can be obtained by computing the Gröbner basis of I or using triangular decomposition with an appropriate order of variables [4]. For the solutions of quasi-linear polynomial systems, a symbolic algorithm based on the minors of the coefficient matrix is given in [17]. That algorithm identifies all cases (including degenerate cases) in parametric space (the X space) and constructs the uniform solution in $Q(R)$ for each case.

Here we will show that the null-space of A^t gives us an alternative way to do the elimination. But this method cannot always guarantee success. However, the following theorem gives us a way to check whether or not the correct projection is successfully determined. Some techniques of the "numerical algebraic geometry", (e.g. witness sets) are used here.

The concept of *generic point* over $\mathbb{C}$ plays an essential role in "Numerical Algebraic Geometry". Suppose some property P is satisfied everywhere except on a proper algebraic subset U of an irreducible variety V. We call the points in $V\backslash U$ generic points. Then $\dim V > \dim U$, so $V\backslash U$ is dense in V (with the standard

Lebesgue measure 1). So we say P holds with *algebraic probability one* for a random point of V.

In [20] Sommese and Wampler introduce the concept of *Witness Sets* of an algebraic variety V, denoted by $W(V)$, which is the key data in a numerical irreducible decomposition. A witness set for a k-dimensional solution component consists of k random hyperplanes and all isolated solutions in the intersection of the component with those hyperplanes. The degree of the solution component equals the number of witness points. If each point in $W(V)$ is contained in other variety V', then $V \subseteq V'$ with probability 1. This nice property can be used to execute numerical radical ideal membership testing [14].

Theorem 6.6.1. *Let* $\{g_1, ..., g_r\}$ *be a basis of* $Null(A^t)$ *and* $I' := \langle g_1^t \cdot \boldsymbol{b}, ..., g_r^t \cdot \boldsymbol{b} \rangle$ *then:*

1. $V(I_x) \subseteq V(I')$
2. $\forall\ p \in W(V(I'))$, $\text{rank}(A_p) = \text{rank}([A_p, \boldsymbol{b}_p]) \Rightarrow V(I_x) = V(I')$

Proof. Proof of (1): Because each $g_i \in Null(A^t)$, it follows that $g_i^t \cdot (AY + \mathbf{b}) = g_i^t \cdot \mathbf{b} \in I$. Also $g_i^t \cdot \mathbf{b}$ only involves the variables $x_1, ..., x_s$, so $g_i^t \cdot \mathbf{b} \in I_x$. Hence $I' \subseteq I_x$, which implies (1).
Proof of (2): it is only necessary to prove that $V(I') \subseteq V(I_x)$. For any generic point $p \in V(I')$ in X-space, $\text{rank}(A_p) = \text{rank}([A_p, \mathbf{b}_p])$. It means that this linear equation after fixing the value of x must have at least one solution y_p in Y-space. So (p, y_p) must be in V. This implies $p \in V(I_x)$. This is true at generic point of each component of $V(I')$, so (2) is true. □

Remark 6.6.1. *In [19, 20], a new field "Numerical Algebraic Geometry" was described which led to the development of homotopies to describe all irreducible components of the solution set of a polynomial system.*

The key tool to numerically solve polynomial systems is homotopy continuation. Homotopy methods define families of systems, embedding a system to be solved in a homotopy, connecting it to a start system whose solutions are known. Continuation methods are then applied to track the paths defined by the homotopy, leading to the solutions. By random choices of constants in the homotopy one can prove that, except for an algebraic set of bad choices of constants, singularities and diverging paths can only occur at the end of the paths, when the system to be solved has singular solutions or fewer solutions than the generic root count.

The paper [13] shows that the projection of a variety can be constructed by combining an interpolation and a homotopy method. They first compute the witness set of the variety by using some random linear equations only involving the variables of X space. Then they use enough projected points of witness set to interpolate the polynomials which only involve the variables of X space. But the difficulty of this

approach is that as the number of equations increase the number of homotopy paths will grow exponentially.

More equations in the system means more rows in A (more columns in A^t). Suppose d does not change. Based on the analysis in section 6.5.2, when the number of rows in A changes from m to m_1, the bound of the size of the Sylvester matrix changes from $m^{s+1}d^s$ to $m_1^{s+1}d^s$. The cost of the algorithm in this paper grows polynomially. However, the method in the paper cannot identify degenerate cases.

EXAMPLE 6.6.2. *Let us consider a quasi-linear polynomial system with* 3 *equations* 4 *variables and degree* 3*:*

$$\begin{pmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} \tag{6.6.1}$$

where $a_i, b_i, c_i, i = 1, 2, 3$ *are some polynomials in the variables* x, y*. These polynomials are chosen randomly with degree* 2 *(with coefficients in* $\mathbb{R}$*). The method of this paper is used to compute a null-space basis and construct a projected polynomial* $f(x, y)$ *with degree* 6*. Maple 10 is used to compute a Gröbner basis of the ideal* $\langle a_1z_1+b_1z_2-c_1, a_2z_1+b_2z_2-c_2, a_3z_1+b_3z_2-c_3\rangle$ *with the order* $x \prec y \prec z_1 \prec z_2$*. The elimination ideal is generated by one polynomial* g*. We find that the distance between* f, g *(both have leading coefficient* 1*) is less than* 10^{-9}*. For this example, the largest constant matrix processed in our algorithm is* 56×45 *and the time for computing the SVD is less than 0.1 second. However the computer (CPU 1.5GHZ and RAM 512MByte) used* 72.3 *seconds to calculate the Gröbner basis for this system.*

6.7 Discussion

Several equivalent definitions of the rank of polynomial matrices have been studied. A simple probabilistic algorithm is proposed to evaluate the rank only using one generic point. For an exact matrix, it can provide the correct answer with probability 1. In the approximate case, we only estimate the lower bound of the ϵ-rank. Further work is needed to study the rank of polynomial matrices given by approximate data.

Using generalized Sylvester methods, the computation of the null-space basis of a polynomial matrix is reduced to computing the SVD of some constant matrices. A degree bound for the null-space basis is given and termination of the method is demonstrated.

Another result of this article is that the relation between null-space basis and syzygy module is given explicitly. In the special case of null-space with rank one, the null-space basis is the syzygy module basis and it can be obtained by the algorithm presented in this paper.

Two applications to GCD and elimination ideals are mentioned briefly. A detailed discussion about its application to differential elimination method for partial

differential equations is given in another paper by Wu and Reid [22].

The complexity and stability of the method will be studied in future work. Since the Sylvester matrices have block Toeplitz structure, hopefully Zúñiga and Henrion's methods can also be applied to speed up the required multivariate polynomial matrix solving.

Acknowledgements

I appreciate my supervisor Greg Reid for his helpful comments and proofreading which greatly assisted me with the paper. I have also enjoyed and benefited from the AMS/SIAM Special Session on Symbolic-Numeric Computation and Applications in San Antonio, organized by Agnes Szanto, Jan Verschelde, and Zhonggang Zeng. I would like to thank Prof. Zeng for a good motivating example to show the instability of polynomial division. I also want to express my thanks to Mark Giesbrecht, George Labahn and Arne Storjohann for their helpful comments on this paper.

Bibliography

[1] William C. Brown. *Matrices Over Commutative Rings*. Marcel Dekker, New York, 1992.

[2] M. T. Chu, R. E. Funderlic, and R. J. Plemmons. Structured low rank approximation. Linear Algebra and Applications, 366, pp. 157C172, 2003.

[3] Robert M. Corless, Patrizia M. Gianni, Barry M. Trager, Stephen M. Watt The Singular Value Decomposition for Polynomial Systems. Proceedings of ISSAC'95, pages: 195 - 207, 1995.

[4] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms*. Springer-Verlag, New York, 1992. Undergraduate Texts in Mathematics.

[5] D. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry*. Springer-Verlag, New York, 1998. Graduate Texts in Mathematics.

[6] C. Eckart, and G. Young. The approximation of one matrix by another of lower rank. Psychometrika, I, pages 211-218, 1936.

[7] Ralf Fröberg. *An Introduction to Gröbner Bases*. John Wiley and Sons, England, 1997. Pure and Applied Mathematics.

[8] Erich Kaltofen, Zhengfeng Yang, and Lihong Zhi. Structured low rank approximation of a Sylvester matrix. In Symbolic-Numeric Computation, pages 69-83, 2006.

[9] Daniel Lazard. Resolution des Systemes d'Equations Algebriques. Theor. Comput. Sci. 15: 77-110, 1981.

[10] Bingyu Li, Zhengfeng Yang, and Lihong Zhi. Fast low rank approximation of a Sylvester matrix by structured total least norm. *Journal of Japan Society for Symbolic and Algebraic Computation*, 11(3,4):165–174, 2005.

[11] Zhiping Lin. On syzygy module for polynomial matrices. *Linear Algebra and its Applications*, 298:73–86, 1999.

[12] Michael T. McClellan. The exact solution of systems of linear equations with polynomial coefficients. *J. of the Association for Computing Machinery*, 20(4):563–588, 1973.

[13] G.J. Reid, C. Smith, and J. Verschelde. Geometric completion of differential systems using numeric-symbolic continuation. *SIGSAM Bulletin*, 36(2):1–17, 2002.

[14] Greg Reid, Jan Verschelde, Allan Wittkopf, and Wenyuan Wu. Symbolic-numeric completion of differential systems by homotopy continuation. In *Proc. 2005 Internat. Symp. Symbolic Algebraic Comput. ISSAC' 05*, pages 269–276, 2005.

[15] J. B. Rosen, H. Park, and J. Glick. Total least norm formulation and solution for structured problems, SIAM J. Matrix Anal. Appl., 17, pp. 110C128, 1996.

[16] J. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27:701–717, 1980.

[17] William Y. Sit. A theory for parametric linear systems. In S. M. Watt, editor, *Proc. 1991 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'91)*, pages 112–121, New York, 1991. ACM Press.

[18] A.J. Sommese and J. Verschelde. Numerical homotopies to compute generic points on positive dimensional algebraic sets. *Journal of Complexity*, 16(3):572–602, 2000.

[19] Andrew J. Sommese and Charles W. Wampler. Numerical algebraic geometry. In *Proceedings of the AMS SIAM Summer Seminar in Applied Mathematics*, pages 749–763, 1995.

[20] Andrew J. Sommese and Charles W. Wampler. *The Numerical Solution of Systems of Polynomials*. World Scientific, Singapore, 2005.

[21] Arne Storjohann and Gilles Villard. Computing the rank and a small nullspace basis of a polynomial matrix. In *Proc. 2005 Internat. Symp. Symbolic Algebraic Comput.(ISSAC'05)*, pages 309–316, 2005.

[22] Wenyuan Wu and Greg Reid. Application of numerical algebraic geometry and numerical linear algebra to PDE. In Jean-Guillaume Dumas, editor, *Proc. 2006 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'06)*, pages 345-353 2006.

[23] Z. Zeng. The approximate GCD of inexact polynomials. part I: a univariate algorithm. Manuscript, 2004.

[24] Z. Zeng and B. H. Dayton. The approximate GCD of inexact polynomials part II: a multivariate algorithm. In Jaime Gutierrez, editor, *Proc. 2004 Internat. Symp. Symbolic Algebraic Comput.(ISSAC'04)*, pages 320–327, New York, 2004. ACM Press.

[25] Juan C. Zuniga and Didier Henrion. Block Toeplitz methods in polynomial matrix computations. In *International Symposium on Mathematical Theory of Networks and Systems*, 2004.

Chapter 7

Conclusion and Future Work

General (e.g. under and over-determined) systems of polynomially nonlinear PDE are rapidly becoming more common in diverse applications. Analyzing such systems requires prolonging them, and detecting and including their integrability conditions. Existing symbolic methods are very expensive and only apply to exact input systems and not the approximate ones occurring in applications.

In this Thesis, new symbolic-numeric completion methods have been introduced and their fundamental mathematical properties have been investigated. We have introduced two different types of general methods for completion of PDE: hybrid symbolic-numeric methods and pure numerical methods. In general the hybrid method can be applied to large and sparse exact systems. The pure numerical method is more suitable for small dense systems with approximate input. For a special class of PDE, called t-dominated systems, we proposed a fast t-prolongation method to compute a Riquier Basis of the input system in an implicit form. A common feature of these new methods is the strong use of geometry to obtain numerically stable methods.

7.1 Conclusion and Main Results

Let us conclude by stating the main results we have obtained:

As the first symbolic-numeric completion method, **HybridRif** divides the completion task into two complementary parts: determining the leading linear and leading nonlinear subsystems. These subsystems are updated finitely many times during the application of **HybridRif**. The basic idea is that the leading linear PDE are processed by a symbolic method (a modified version of **rifsimp**) and the leading nonlinear part is processed by a numerical method (homotopy continuation). A key improvement in the method is that the radical ideal membership testing required by **HybridRif** can be tested by only using the witness points on components. This can be very expensive in symbolic approaches, because it requires computing the Gröbner basis of a radical ideal. The other key method is the Diagonal Homotopy

Method. This allows an incremental equation by equation strategy, which allows one to exploit system structure and compute with dramatically fewer continuation paths.

For dense systems with approximate input, it is usually numerically unstable to apply symbolic methods directly. We presented a pure numerical method for fully non-linear approximate PDE, which exploited a polynomial matrix theory. The first main result of this method is to provide a simple criterion for completion by comparing the ranks of the Symbol Matrix and Augmented Symbol Matrix at generic points. The second main contribution is the construction of projected constraints in a numerically stable way by computing the (left) null-space of the Symbol Matrix. The main idea is to transform the polynomial matrix to a matrix polynomial and then reduce it to constant matrix by looking at the coefficients of all monomials. However this method is very difficult to apply to large system, since the size of the Sylvester matrix increases exponentially with the number of independent variables. Note that the Sylvester matrix is highly structured and exploiting this to improve efficiency is an important task for future research.

More recently we developed an extremely efficient method for a certain class of PDE, called *t-dominated systems*. This class is a general and natural type of PDE/DAE system with many applications in multi-body mechanics and chemical dynamics. We propose the first numerically stable method to find all the hidden constraints of square t-dominated PDE without expression explosion and complicated differential elimination. From a theoretical point of view, this work is also related to the famous *Jacobi Conjecture* of Differential Algebra. Another contribution is to provide a general framework to generalize numerical techniques for DAE/ODE to t-dominated PDE.

At the end of each chapter (paper), some future research directions have been discussed separately. The geometric study of PDE by using symbolic-numeric computations as carried out in this Thesis forms part of a new area called "Numerical Jet Geometry".

Unlike algebraic systems, differential systems can generate more and more (new) equations after prolongations. But such differential problems can be reduced to algebraic ones after we obtain the involutive forms, and this should be the central task of Numerical Jet Geometry. In numerical algebraic geometry, the solutions (including isolated and positive dimensional components) can be represented by witness sets. Following Sommese and Wampler's idea, we can use witness points on the zero sets of an involutive form to represent all the solutions. In particular each witness point gives an approximation of a (truncated) formal power series solution of the original PDE at a given point x^0 after specifying some initial data (the number of the initial data is equal to the local dimension at this point).

So the key data structure of Numerical Jet Geometry is $[\text{Invol}, A, \text{ID}, x^0]$ where: Invol is the involutive form of an input system; A is a set of witness points on a component of the zero set of Invol; ID is the specified initial data at a point x^0.

Involutive form is a local concept. For different components we may obtain

different involutive forms. Usually to compute the generic case is easier (e.g. the methods in Chapter 3 and 4 only pursue the generic cases). However the singular cases are also very important but more challenging.

7.2 Future Research Directions

In general, differential problems are more complicated and more interesting than algebraic ones and lead to many deep research questions. Finally, we choose some of the most important research directions of this new area and sketch them in an informal way.

Involutivity and Regularity As discussed in Chapter 1 and Chapter 3, the involutivity of the Symbol is the key for termination of Cartan-Kuranishi method. Cartan's test and Spencer's Cohomology method both have their own advantages and disadvantages. One direction is to combine them together to yield a stable and efficient method to check the involutivity of the Symbol.

Another direction is more related to the concept of "Castelnuovo-Mumford Regularity" in Commutative Algebra, which is a very important invariant in Commutative Algebra and Algebraic Geometry. As we discussed in Chapter 1, at a point of the zero set of a PDE system R with m dependent variables and n independent variables, the Symbol can be considered as A-module, a submodule of A^m, where A is the polynomial ring $\mathbb{F}[x_1, ..., x_n]$. Eisenbud [5, 6] defined the Regularity of a module by using "Minimal Free Resolutions" which is an exact sequence by computing syzygy modules. Mansfield [9] proposed a simple criterion for involutivity by using the dual of the δ sequence of the Symbol which essentially is also computing syzygies. Certainly involutivity and regularity are closely related [4, 3]. Recently, Malgrange shows that Cartan involutivity is equivalent to Mumford regularity in [8]. Regularity is a hot topic in the areas of commutative algebra and algebraic geometry and many approaches have been developed [2, 1]. One research project is to adapt Bayer-Stillman's criterion for detecting m-regularity to numerical computation. Hopefully, we can "transplant" these techniques into our area to study PDE.

Non-square t-dominated systems The current fast t-prolongation method can only be applied to square t-dominated system. A natural question will be how can we extend it to non-square systems.

For an under-determined system ($\ell < m$), we can choose ℓ dependent variables and consider the others as parameters to produce a square system. There are $\binom{m}{\ell}$ different choices, but a feasible choice must satisfy the condition: existence of maximum transversal value for the chosen $\ell \times \ell$ sub-matrix of the signature matrix. This problem can also be formulated as an integer linear programming

problems with two stages. At the first stage, we look for a feasible choice of dependent variables:

$$\left\{\begin{array}{rll} \text{Maximize} & D=\sum_{i,j}\sigma_{ij}\xi_{ij}, & \\ \text{where} & \sum_j \xi_{ij}=1, & \text{for each } i \\ & 0\leq\sum_i \xi_{ij}\leq 1, & \text{for each } j \\ & \xi_{ij}\geq 0. & \end{array}\right. \tag{7.2.1}$$

The value of D will give us the maximum transversal value and the values of ξ_{ij} will indicate which columns we need to choose (if $\sum_i \xi_{ij}=1$, then column j is chosen). After the first stage we will have a square system, the methods we have discussed in Chapter 4 can be applied to solve it.

The case of over-determined systems ($\ell > m$) is more challenging. One idea is to seek square sub-systems and apply the fast prolongation method to them separately. Then the output needs to be intersected with the remaining equations (perhaps by some type of generalized diagonal differential homotopy).

Singular Components of Differential Systems The methods introduced in Chapter 3 and Chapter 4 only pursue the generic components of differential systems. An important question in Numerical Jet Geometry is to compute the singular components. Let us consider Hubert's example: $u_x^2+xu_x-u=0$ with unknown function $u(x)$. It has a family of general solutions: $u=cx+c^2$ (c is a parameter depending on the initial value) and a singular solution: $-1/4x^2$, which is an envelope. For square systems, one idea is to add the Symbol equation $(2u_x+x)v_{xx}=0$ to the original system and compute the witness points of the embedding system in the higher dimensional space, $J_2\times\mathbf{S}^2T^*$. If the v_{xx} coordinate of a witness point is nonzero (equivalently the Symbol matrix is singular), then the projection of the component containing this point is singular in J_2 with probability 1. Otherwise it is nonsingular.

Unlike the binary tree splitting in algebraic exact methods (e.g. **Rifsimp** and DiffAlg), the decomposition in $J_q\times\mathbf{S}^qT^*$ discussed above has more flavor of geometry and it is more natural and consequently more stable.

Approximate Polynomial Algebra The spirit of this Thesis is to study stable methods for approximate computation with PDE. So approximate computation in polynomial algebra is a subfield of Numerical Jet Geometry .

A property of a system is said to be *stable*, if it retains this property under small perturbations. In other words, all the systems in an "open set" containing this system also have this property. So this given system must be in some kind of generic position. This is why genericities are so important in numerical solving. For example, a generic hyperplane can drop the dimension of an algebraic set by one. In addition, a generic choice of all the nonzero coefficients

of a square polynomial system guarantees the number of roots is equal to the mixed volume of the system. In Chapter 4, we also see that genericity is the key for t-dominated systems and the fast t-prolongation method.

On the other hand, consider the case where a given system satisfies some unstable properties, (e.g. two univariate polynomials have nontrivial GCD, or an $n \times n$ matrix has rank less than n). Such systems are said to be degenerate. The numerical computation of such degenerate systems will be difficult and unstable. And the problem itself is ill-posed!

However sometimes we are interested in unstable properties (e.g. the existence of nontrivial GCD, see Zeng [10] for a numerical approach). In particular we might say that a system approximately has such an unstable property up to some tolerance. By our use of the term "*approximately*" here, we mean that there exists a "*nearby*" system which exactly has such property. In fact, this is the backward stability problem. Now let us describe how to transform an ill-posed problem to a well-posed one by using the following general approach.

Let Σ be a parameter space in which any system can be embedded. Suppose we can introduce a metric $\mathrm{dist}(\cdot\ ,\cdot)$ in Σ. Suppose we are interested with a family of properties $[P_1, P_2, ...]$. Let $\Phi(P_i)$ be the set of all the systems which has property P_i. Each $\Phi(P_i)$ is a closed set in Σ, since P_i is a unstable property. Consequently $\Phi(P_1 \cap \cdots \cap P_i) = \Phi(P_1) \cap \cdots \cap \Phi(P_i) =: C_i$ is also a closed set. So given a system s, we can define $\mathrm{dist}(s, C_i) := \min_{s' \in C_i} \mathrm{dist}(s, s')$.

Given an input system s and tolerance ϵ, let

$$k := \max\{i : \mathrm{dist}(s, C_i) < \epsilon, \quad \text{and} \quad \mathrm{dist}(s, C_{i+1}) > \epsilon\}. \tag{7.2.2}$$

Then we say the system s has properties $\{P_1, ..., P_k\}$ approximately with tolerance ϵ. Now this definition is well-defined, because a sufficient small perturbation will not change the value of k.

A future research direction is to apply the general approach that we introduced above to particular classes of approximate problems.

Sparse structured matrices From a computational point of view, problems in Approximate Polynomial Algebra need to be reduced to linear algebra. There are two ways to transform a nonlinear system into a linear system. The first way is by linearization (only retaining local information). The second way is to embed the problem into a higher dimensional vector space (e.g. monomial basis), whereby more degrees of freedom are introduced. Without losing nonlinear information, the embedded system is highly structured (e.g. represented by Sylvester matrices).

We know the SVD gives a measurement and construction of the low rank approximation of a given matrix [7]. But it is challenging to find a low rank

approximation of a given structured matrix with the same structure.

The second problem is that the dimension of the embedding space is often very large. Obviously, the general methods in (numerical) linear algebra cannot apply to such large systems. However, it is so sparse that it generates a linear subspace with low dimension! Structure preserving techniques need to be developed to guarantee computation in these linear subspaces.

Bibliography

[1] I. Bermejo and P. Gimenez. Saturation and Castelnuovo-Mumford regularity. J. Algebra, Vol 303, pages 592–617, 2006.

[2] David Bayer and Michael Stillman. A criterion for detecting m-regularity. Invent. Math. 87. pages 1–11, 1987.

[3] R. Bryant and P. Griffiths. Characteristic cohomology of differential systems I. J. Amer. Math. Soc. 8 (1995), 507-596.

[4] R. Bryant, S.-S. Chern, R. Gardner, P. Griffiths and H. Goldschmidt. Exterior Differential Systems, MSRI Publications, Springer, 1989.

[5] D. Eisenbud. Commutative Algebra. With a view toward Algebraic Geometry. Springer, 1995.

[6] David Eisenbud. The geometry of syzygies. A second course in commutative algebra and algebraic geometry. New York: Springer-Verlag, 2005.

[7] G.H. Golub, C.F. Van Loan. Matrix Computation. The Johns Hopkins University Press, 1996.

[8] Bernard Malgrange. Cartan involutiveness = Mumford regularity. Commutative algebra, Contemp. Math. 331, Amer. Math. Soc., 2003.

[9] E. Mansfield. A simple criterion for involutivity. J. London Math. Soc., Vol 54, pages 323-345, 1996.

[10] Z. Zeng and B. H. Dayton. The approximate GCD of inexact polynomials part II: a multivariate algorithm. In Jaime Gutierrez, editor, Proc. 2004 Internat. Symp. Symbolic Algebraic Comput.(ISSAC'04), pages 320–327, New York, 2004. ACM Press.

Curriculum Vitae

Wenyuan Wu

Birth: Nov. 9 1976, Chengdu, China.
Ph.D. student of the Department of Applied Mathematics
University of Western Ontario, London, ON. Canada
Supervisor: Prof. Greg Reid
Office Address: Middlesex College 275K.

Research Background

Numerical Jet Geometry

My recent work concerns Numerical Jet Geometry, and involves the application of algebraic geometry to systems of partial differential polynomials with approximate coefficients. In this new area, I focus on two directions. One is to apply geometric methods to PDE to yield involutive systems. A new theory for polynomial matrices needed to be developed as a consequence of this work. Another direction is to identify a class of PDE with nice properties. For this class, we have efficient and stable methods based on combinatorial analysis and linear programming.

Symbolic Computation in Polynomial Rings

This is the study of solutions of polynomial systems with exact input, which often appear in non-linear phenomena. I mastered the fundamental computational techniques e.g., Gröebner bases, triangular decompositions, rational univariate representations. Moreover, we introduced equiprojectable decomposition and developed an efficient Maple package for solving zero dimensional polynomial systems using the equiprojectable decomposition combined with modular methods.

Approximate Polynomial Algebra

Motivated by realistic problems, a trend from pure symbolic computation to symbolic-numerical computation has occurred. I am mainly concerned with adapting the concepts in exact polynomial algebra to the approximate case and developing numerical stable methods to obtain structural information characterizing polynomial ideals. Also I am interested with approximate triangular representations of algebraic varieties and the associated sensitivity analysis.

Education

- 2003 - present: Ph.D. student in Department of Applied Mathematics, University of Western Ontario, Canada.
- 2006.09 - 2006.12: NSF funded long term visitor to the Institute of Mathematics and its Applications, University of Minnesota, Minneapolis.

- 1999 - 2002 : M.S. of Computer Science, Graduate School of Chinese Academy of Sciences, Chengdu, China.
- 1995 - 1999 : B.S. of Mathematics, Department of Mathematics, Peking University, Beijing, China.
- 1997 - 1999 : (second) B.S. of Economics, Chinese Economics Research Center, Peking University.

Publications

1. Wenyuan Wu, Greg Reid: *Symbolic-numeric Computation of Implicit Riquier Bases for PDE*, Proceedings of the 2007 International Symposium on Symbolic and Algebraic Computation (ISSAC'07), pages 377-385, ACM 2007.
2. Marc Moreno Maza, Greg J. Reid, Robin Scott and Wenyuan Wu: *On Approximate Triangular Decompositions in Dimension Zero*, Journal of Symbolic Computation, Vol 42(7), page 693-716, 2007.
3. Marc Moreno Maza, Greg J. Reid, Robin Scott and Wenyuan Wu: *On Approximate Linearized Triangular Decomposition.* Symbolic-Numeric Computation book, Birkhauser Basel Boston, in press.
4. Wenyuan Wu, Greg Reid: *Application of Numerical Algebraic Geometry and Numerical Linear Algebra to PDE.* Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation (ISSAC'06), July 9-12 2006, Genova, Italy. Edited by Jean-Guillaume Dumas, pages 345-353, ACM 2006.
5. Wenyuan Wu, Zhenbing Zeng: *Reachability Analysis of Petri Net.* Journal of System Simulation, 2005 Vol 17 page 17-26 (Chinese).
6. Greg Reid, Jan Verschelde, Allan Wittkopf, and Wenyuan Wu: *Symbolic-Numeric Completion of Differential Systems by Homotopy Continuation.* Proceedings of the 2005 International Symposium on Symbolic and Algebraic Computation (ISSAC'05), July 24-27 2005, Beijing, China. Edited by Manuel Kauers, pages 269-276, ACM 2005.
7. Xavier Dahan, Marc Moreno Maza, Eric Schost, Wenyuan Wu and Yuzhen Xie: *Lifting techniques for triangular decompositions.* Proceedings of the 2005 International Symposium on Symbolic and Algebraic Computation (ISSAC'05), pages 108-115, ACM 2005.
8. Marc Moreno Maza, Greg J. Reid, Robin Scott and Wenyuan Wu: *On Approximate Triangular Decompositions I Dimension Zero.* In proc. of International Workshop on Symbolic-Numeric Computation, page 250-275, 2005.

9. Marc Moreno Maza, Greg J. Reid, Robin Scott and Wenyuan Wu: *On Approximate Triangular Decompositions II Linear Systems.* In proc. of International Workshop on Symbolic-Numeric Computation, page 276-296, 2005.

10. X. Dahan, M. Moreno Maza, E. Schost, W. Wu and Y. Xie. *Equiprojectable decompositions of zero-dimensional varieties.* In proc. of International Conference on Poynomial System Solving, page 69-71, France, 2004.

11. Wenyuan Wu, Zhenbing Zeng, Hongguang Fu: *Knowledge base of elementary geometry based on Ontology.* Journal of Computer Applications 2002, Vol 22 page 9-14 (Chinese).

12. Master Thesis: *On Reachability of Petri Net.* Masters Thesis, Graduate School of Chinese Academy of Sciences, 2002 (Chinese). Advisor: Prof. Lu Yang.

Work in Progress

- Wenyuan Wu, Greg Reid: On Approximate Ideal and Approximate Ideal Membership Test.
- Wenyuan Wu: Computing the Rank and Null-space of Polynomial Matrix.

Academic Honors

- Fields Institute Postdoctoral Fellowship in Applied Mathematics at the University of Western Ontario, 2007.
- Chinese Government Award for Outstanding Students Abroad, $5000. Awarded at General Chinese Consulate in Toronto, on April 25, 2007.
- General Membership of Institute for Mathematics and its Applications, University of Minnesota, Sept - Dec 2006, research funding $4000.
- Western Graduate Thesis Research Award of the University of Western Ontario, 2006, research grant $1500.
- Distinguished student paper award, ACM ISSAC 2005, Beijing China.
- Best Poster Award, ACM ISSAC 2005, Beijing China.
- *Special University Scholarship, Graduate Research Assistantship, Teaching Assistantship, Western Graduate Research Scholarships and International Graduate Student Scholarship* from the Dept. of Applied Mathematics, University of Western Ontario starting Sept. 2003 till now, approx. $29000/year .
- PhD Entrance Research Award, University of Western Ontario, 2004, $4732.

Presentations

1. *"Symbolic-numeric Computation of Implicit Riquier Bases for PDE"*, AMS Special Session on Differential Algebra, New York, April 2007.

2. *"On Approximate Triangular Decomposition in Dimension Zero"*, IMA, the University of Minnesota, Nov 15, 2006.

3. *"Application of Numerical Algebraic Geometry and Numerical Linear Algebra to PDE"*, ISSAC 2006, Genova Italy. July 9-12 2006.

4. *"Numeric Algebraic Geometric Methods for PDEs"*, invited talk at University of Technology in Helsinki Institute of Mathematics, April 12, 2006.

5. *"Introduction to Application of Numerical Algebraic Geometry to PDE"*, Formal theory of partial differential equations and their application Workshop at University of Joensuu, Finland, April 2-9, 2006.

6. *"Differential elimination for approximate PDE systems"*, AMS Special Session: Symbolic-Numeric Computation and Applications on January 15 2006 at San Antonio, Texas.

7. *"Progress on Symbolic and Numeric Differential Elimination Methods for Differential Systems"*, invited talk at Chengdu Institute of computer applications, Chinese Academy of Sciences, Aug 2005.

8. *"On Approximate Triangular Decompositions I: Dimension Zero"*, SNC 2005, Xi'an, China. July 19-21 2005.

9. *"Determination of the dimension of a variety and some applications"*, AMS Special Session on Solving Polynomial Systems, October 23-24 2004, Northwestern University, Chicago.

Poster Presentations

1. *"Differential elimination of PDEs by numerical algebraic geometry and numerical linear algebra"*, Blackwell-Tapia Conference, Minnesota, 2006.

2. *"Symbolic and numerical methods for partial differential equations"*, Software for Algebraic Geometry Workshop, IMA, Minnesota, 2006.

Teaching Experience

1. Winter term 2007: Organized a seminar on "Regularity of Ideals" and gave a series of lectures.

2. Winter term 2007: Teaching assistant for AM325 *Optimization* and AM213 *Linear Algebra II*. Gave two AM213 lectures.

3. Winter term, 2006: Teaching assistant for AM325 *Optimization*.

4. Winter term, 2006: Gave a series of 5 lectures in course of AM586 *Geometric and Algebraic Aspects to PDEs*.

5. Fall term, 2005: Teaching assistant for AM315 *Partial Differential Equations* and AM301 *Complex Variables with Applications*. Gave one AM315 lecture.

6. Winter term, 2005: Teaching assistant for *Advanced Calculus II*.

7. Fall term, 2004: Teaching assistant for *Applied Mathematics for Engineers*.

8. Fall term, 2003: Teaching assistant for *Advanced Calculus I*.

Programming Experience

- 2006.9 - 2007.1: DAE/PDAE solving using fast completion method and Linear Programming, in Maple.

- 2006,1 - 2006.4: Course Project on Scientific Parallel Computation using C and LaPack on Sharcnet (http://www.sharcnet.ca).

- 2005.8 - 2006.7: Project of Numerical Differential Elimination using homotopy methods and polynomial matrix, in Maple.

- 2005.1 - 2005.5: Project of Approximate triangular set by polynomial interpolation and PHCpack, in Maple.

- 2004.10 - 2005.1: Symbolic-numerical Completion using homotopy methods, in Maple.

- 2004.8 - 2004.10: Regular Chain Maple package for symbolic triangular decomposition using modulo and lifting method.

- 2002.1 - 2002.5: Hopfield Neutral Network simulation for solving reachability of Petri net, in Matlab.

- 2000.9 - 2001.5: industrial software development of automated reasoning engine for elementary geometry, in Lisp.