Linköping Studies in Science and Technology
Thesis No. 540

Algebraic Methods for Modeling and Design in Control

Mats Jirstrand

REGLERTEKNIK
AUTOMATIC CONTROL
LINKÖPING

Division of Automatic Control
Department of Electrical Engineering
Linköping University, S–581 83 Linköping, Sweden
WWW: `http://www.control.isy.liu.se`
Email: `matsj@isy.liu.se`

Linköping 1996

Algebraic Methods for Modeling and Design in Control

Department of Electrical Engineering,
Linköping University,
S–581 83 Linköping,
Sweden.

ISBN 91-7871-676-4
ISSN 0280-7971
LiU-TEK-LIC-1996:05

To Emma

Abstract

Computer algebra or symbolic computation has been recognized as an important tool in many engineering disciplines and continues to find new fields of application. In this thesis we investigate a number of constructive methods implemented as parts of computer algebra packages and utilize them both theoretically and practically in connection with problems in modeling and design of control systems.

A large class of systems can be described by a set of polynomial differential equations and there are different approaches to analyze systems of this kind. In this thesis we utilize mathematical tools from commutative and differential algebra such as Gröbner bases and characteristic sets to study input-output relations of systems given in state space form.

We show that despite the fact that the state space representation of a system can be described by a finitely generated differential ideal, this is generally not true for the corresponding differential ideal of input-output relations. However, the input-output relations up to a fixed order can be computed and represented using non-differential tools. Using characteristic sets we also show how the above problem of finite representation of the input-output relations can be resolved.

Many problems in control theory can be reduced to finding solutions of a system of polynomial equations, inequations and inequalities, a so called *real polynomial system*. The *cylindrical algebraic decomposition* method, which was invented in the mid seventies is an algorithm that can be utilized to find solutions to such systems. The extension of real polynomial systems to expressions involving Boolean operators ($\vee, \wedge, \neg, \rightarrow$) and quantifiers ($\exists, \forall$) is called the first-order theory of real closed fields. There are algorithms to perform *quantifier elimination* in such expressions, i.e., to derive equivalent expressions without any quantified variables.

We show how these algorithms can be used to solve problems in control such as choosing parameters in a controller to stabilize an unstable system with parameter uncertainties; feedback design of linear, time-invariant systems; computation of bounds on static nonlinearities in feedback systems that ensure stability; computation of (asymptotically stable) equilibrium points for nonlinear systems subjected to constraints on the control and state variables; and curve following in the state space of a nonlinear system.

Acknowledgments

First of all I would like to thank my supervisor, Professor Torkel Glad, for his proficient guidance. During the development of this work we have had many valuable discussions and inspiring talks.

I would also like to thank Professor Lennart Ljung for providing me the opportunity to join the group and for creating such a stimulating research atmosphere.

The Automatic Control group in Linköping is really an extraordinary working place. Besides the serious research activities I have to mention the often incredible coffee room discussions, the `JunkMail` email list, the Junk Seminars, and the Friday lunch floorball sessions, just to single out a few things. I thank all the people in the group for their contribution.

I am also grateful to Krister Forsman who introduced me to commutative algebra and its applications to control theory. Thanks to him I began the work which led to this thesis. The SyCo (Symbolic Computation) group seminars and their participants have also been a great source of inspiration. A special thank also to Håkan Fortell and Roger Germundsson for inspiring research discussions.

The following people have read the manuscript during various stages of preparation: Håkan Fortell, Johan Gunnarsson, Niclas Bergman, and Krister Edström. I thank you all for your valuable remarks and suggested improvements, which I really appreciated. I am also thankful to Krister Edström, my office colleague, for many clarifying discussions regarding both courses and research.

I am indebted to Peter Lindskog for always answering my questions about LaTeX and for providing me with valuable macros, for improving the visual appearance of this document.

This work was supported by the Swedish Research Council for Engineering Sciences (TFR), which is gratefully acknowledged. Furthermore I would like to thank Dr. Hoon Hong at Research Institute for Symbolic Computation (RISC) in Austria for providing me with the program QEPCAD, which were used to perform quantifier elimination in many of the examples in the thesis.

I am also grateful to my family for their love and never ending support in whatever I have done.

Finally, I would like to thank Emma for all the love and support she has given me and the patience she has shown during my work with this thesis.

Linköping, February 1996

Mats Jirstrand

Contents

Notation

Symbols

$\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{A}$	The sets of natural, integer, rational, and algebraic numbers.
$\mathbb{R}$, $\mathbb{C}$	The sets of real and complex numbers.
$k[x_1, \ldots, x_n]$	Polynomial ring in the variables $x_1, \ldots, x_n$, coefficients from a field k, usually $\mathbb{Q}$, $\mathbb{R}$ or $\mathbb{C}$.
I, $\sqrt{I}$	Ideal, radical ideal.
$\langle f_1, \ldots, f_s \rangle$	Ideal generated by the polynomials $f_1, \ldots, f_s$.
$\mathbf{V}(f_1, \ldots, f_s)$	Variety (zero set) of the polynomials $f_1, \ldots, f_s$.
$k\{x_1, \ldots, x_n\}$	Differential polynomial ring in the variables $x_1, \ldots, x_n$, coefficients from a field k, usually $\mathbb{Q}$, $\mathbb{R}$ or $\mathbb{C}$.
$[f_1, \ldots, f_s]$	Differential ideal generated by the polynomials $f_1, \ldots, f_s$.
Σ, Σ^c	State space differential ideal, input-output differential ideal.
$\mathbf{A}$	Autoreduced set of differential polynomials.

Operators and Functions

$A \subset B$, $A \subsetneq B$	A is a subset of B, A is a proper subset of B.
$A \cap B$, $A \cup B$	Intersection and union of sets A and B.

$A \setminus B$	The set $\{x \mid x \in A \text{ and } x \notin B\}$.
$>$	Monomial ordering or differential ranking of variables.
$\gcd(f, g)$	Greatest common divisor of two polynomials f and g.
$\mathrm{lcm}(f, g)$	Least common multiple of two polynomials f and g.
$\mathrm{mdeg}(f)$, $\mathrm{tdeg}(f)$	Multi-degree and total degree of a multivariate polynomial f.
$\mathrm{lc}(f)$, $\mathrm{lm}(f)$, $\mathrm{lt}(f)$	Leading coefficient, leading monomial, and leading term of f.
$\mathrm{rem}(f, F)$	Remainder of a polynomial f on division by $F = \{f_1, \ldots, f_s\}$.
$S(f, g)$	S-polynomial of f and g.
$\mathrm{GB}(I)$	Reduced Gröbner basis of the ideal I.
∂	Derivation operator.
$\dot{x}, \ddot{x}, x^{(i)}$	First, second and i^{th} derivative of x w.r.t. t.
y_i, u_i	The i^{th} derivatives w.r.t. t of input and output variables.
L_f	Extended Lie-derivative operator computed with respect to f.
$\mathrm{wt}(p)$, $\mathrm{type}(p)$	Weight and type of an isobaric differential polynomial p.
$\mathrm{ld}(f)$, $\mathrm{lv}(f)$	Leader and leading variable of the differential polynomial f.
I_f, S_f, H_f	Initial, separant of the differential polynomial f. $H_f = I_f S_f$.
$[\mathbf{A}] : H_{\mathbf{A}}^{\infty}$	Set of differential polynomials whose remainder w.r.t. $\mathbf{A}$ is zero.
$\mathrm{prem}(f, g)$	Pseudo-remainder of a polynomial f on division by g.
$\mathrm{subres}_i(f, g)$	The i^{th}-subresultant of two polynomials f and g.
$\mathrm{psc}_i(f, g)$	The i^{th}-principal subresultant coefficient of f and g.
$\mathrm{proj}(f_1, \ldots, f_s)$	Projection operator w.r.t. the set $\{f_1, \ldots, f_r\}$ of polynomials.
$\mathrm{var}(a_1, \ldots, a_s)$	Number of sign variation in a sequence of real numbers.
$f \triangleleft g$	The differential polynomial f is reduced w.r.t. g.
$\wedge$, $\vee$, $\neg$, $\rightarrow$	Conjunction, disjunction, negation, and implication.
$\exists$, $\forall$	Existential and universal quantifier.
$\mathcal{S}$, $\mathcal{S}(x)$	Semialgebraic set and its defining Boolean formula.

Abbreviations

CAD	Cylindrical Algebraic Decomposition.
QE	Quantifier Elimination.
PRS	Polynomial Remainder Sequence.
EPRS	Euclidean Polynomial Remainder Sequence.
PPRS	Primitive Polynomial Remainder Sequence.
SPRS	Subresultant Polynomial Remainder Sequence.
MIMO	Multiple Input Multiple Output.
SISO	Single Input Single Output.
UFD	Unique Factorization Domain.

1

Introduction

Computer algebra or symbolic computation has been recognized as an important tool in many engineering disciplines and continues to find new fields of application. In this thesis we present the mathematical background and a number of methods, or algorithms, that can be utilized for modeling and design of control systems.

In control and systems theory symbolic computation offers solutions to many problems both of theoretical and practical value, e.g., it is easy to manipulate systems of equations to reveal connections between different variables. Another example is computation of analytical expressions for system properties and their dependence on parameters together with graphical presentations in various ways.

The methods studied in this document are constructive algorithms for elimination of variables in systems of differential and non-differential polynomial equations; finding solutions of systems of polynomial equations, inequations and inequalities; and elimination of variables in formulas containing Boolean operators and quantifiers. We also use the methods to treat some theoretical issues in differential algebraic systems theory and demonstrate their applicability to a number of problems both in linear and nonlinear control theory.

The aim of the thesis is twofold. On one hand it should serve as an introduction to constructive algebra and symbolic computation and on the other hand present their applicability to some problems in control and systems theory.

1.1 Differential Algebraic Systems Theory

A large class of dynamical systems can be described by a set of differential equations. In the case when all the differential equations are linear most of the questions about the system can be answered using linear algebra. In the non-linear case the situation is more complicated and there are no general methods. If we constrain ourselves to the fairly large class of systems which can be described by polynomial differential equations, the frameworks of commutative algebra and differential algebra offer insight to some questions.

Commutative algebra is a mathematical branch that arose from the study of polynomial equations. Differential algebra was developed mainly by J.F. Ritt [72] in the forties and is an extension of commutative algebra in the sense that derivation is also taken into account. Differential algebra made its way into systems theory via the discoveries by Fliess in the mid eighties and has now grown quite extensive. A good survey of the topic is given in [27].

Due to the development of powerful computers and advanced mathematical software for symbolic computations, the treatment of polynomial differential equations with commutative and differential algebra has been considerably simplified. As examples we here mention the symbolic math softwares MAPLE [14] and MATHEMATICA [84], which has been of vital importance in this work.

In Part I an introduction to commutative and differential algebra is given together with two constructive methods, Gröbner bases and characteristic sets. These tools are then used in Part II to investigate some problems that arise when treating polynomial differential equations with concepts from commutative and differential algebra.

1.2 Quantifier Elimination

A formula containing polynomial equations and inequalities combined with Boolean operators $\wedge$ (and) and $\vee$ (or) where some of the variables are quantified using the quantifiers $\exists$ (there exists) and $\forall$ (for all) constitutes a sentence in the first-order theory of real closed fields. If all variables in such a formula are quantified the problem of deciding its truth value is denoted a *decision problem*. Suppose that some variables are bounded by quantifiers but others are not. The search for an equivalent expression in the unquantified variables only is called a *quantifier elimination* problem. In the late forties Tarski showed that there is a procedure that solves the above problems in a finite number of steps [79]. Although Tarski gave a constructive proof, the resulting algorithm is impractical even with the power of todays computers. In the mid seventies Collins invented *cylindrical algebraic decomposition* which given a set of n-variate polynomials decomposes $\mathbb{R}^n$ into regions over which the polynomials have constant sign [18]. Utilizing this method the problem of eliminating quantifiers could be carried out for small examples. Recently, effective algorithms for quantifier elimination have made it possible to solve non-trivial problems on workstations.

For *real polynomial systems*, i.e., systems of polynomial equations and inequalities the method of cylindrical algebraic decomposition has a value of its own. Once a decomposition is found, a solution (if there exists one) to any real polynomial system in the defining polynomials can be found.

In Part I we present cylindrical algebraic decomposition, quantifier elimination, and subalgorithms and tools utilized in these methods such as computation of subresultants, principal subresultant coefficients, Sturm chains, and representation of real algebraic numbers.

One of the first attempts to apply quantifier elimination techniques to problems in control was made by Anderson et al. [2], where the possibility of stabilizing an unstable finite-dimensional linear system by static linear output feedback was studied. Since then, both algorithmic improvements and a dramatically increased computer power have made it possible to implement quantifier elimination algorithms.

In Part III we formulate a number of problems in both linear and nonlinear control theory as quantifier elimination problems and illustrate their solution in examples using the program QEPCAD developed by Hoon Hong et al. [19] at RISC (Research Institute for Symbolic Computation) in Austria.

1.3 Contributions

Part I of the thesis serves as an introduction to mathematical concepts and constructive algorithms in commutative, differential, and real algebra but contains essentially nothing new. However, most of the material is scattered over many articles and books and a similar collection of introductory material does not exist to the author's knowledge.

The contributions of Part II consist of some results regarding the description of the input-output structure of systems represented by state space equations. Although the state space representation of a system obeys a description in terms of a finitely generated differential ideal, the description of the corresponding input-output relations is a more intricate problem. We show that the differential ideal describing the input-output relations is, in general, not finitely generated. We also show how this problem can be resolved. One possibility is to consider the differential structures as non-differential ones in infinitely many variables and then use tools from commutative algebra. A technical Lemma 6.6 is proved, which facilitates this approach.

In Part III the contributions are the applications of the quantifier elimination method to problems in control theory such as constructive stability investigations using the circle criterion, computation of equilibrium points and their stability properties of systems subjected to control constraints, and curve following. We also observe that many classical problems in control theory can be formulated using the concepts of semialgebraic sets and formulas in the first-order theory of real closed fields. This implies that these problems can, at least theoretically, be solved in a finite number of steps.

1.4 Outline of the Thesis

The thesis is divided into three parts. The purpose of Part I is to introduce the reader to basic commutative, differential and real algebraic terminology together with a number of constructive methods. In Chapter 2 concepts in commutative algebra such as rings, fields and ideals are introduced. Polynomial rings and their ideals play a special role, since there are many algorithms developed and implemented in computer algebra packages that make computations with these objects possible. We introduce Gröbner bases as the main constructive tool of commutative algebra and show parallels to multivariate polynomial division. Chapter 3 deals with differential algebra. We consider differential rings, fields and ideals and especially differential rings of polynomials of the same reason as in the non-differential case. The constructive methods in this chapter concern characteristic sets and Ritt's algorithm. Chapter 4 is devoted to real algebra, cylindrical algebraic decomposition, and quantifier elimination. After a short introduction to real closed fields, semi-algebraic sets and decompositions induced by zero sets of polynomials, a detailed introduction to constructive tools in real algebra such as subresultants, Sturm chains and cylindrical algebraic decomposition is given. The last topic in this chapter is quantifier elimination, which is extensively used in Part III of the thesis.

In Part II the input-output structure of systems represented by state space equations are investigated from a differential algebraic perspective. An introduction to differential algebraic representation of dynamical systems is given and input-output relations are discussed in Chapter 5. In Chapter 6 we investigate some finiteness issues in differential algebraic systems theory. We point at some problems that arise when treating systems of polynomial differential equations using the framework of differential algebra. We also give a non-differential algebraic approach to the input-output description problem and a treatment with characteristic sets that resolve much of the problems. The chapter ends with some extensions and conclusions of the discussed issues. The results of Part II were carried out with Krister Forsman as supervisor and were published in [29].

Part III presents a number of applications of quantifier elimination to problems in control theory. Chapter 7 serves as a brief introduction to the subject. Applications of quantifier elimination to linear systems such as stabilization and treatment of model uncertainties are presented in Chapter 8. Here we also give examples of the applicability of quantifier elimination in design of controllers and investigations of stability of feedback systems with static nonlinearities. In Chapter 9 quantifier elimination is utilized to compute descriptions of equilibrium points of (nonlinear) dynamical systems subjected to state and control constraints. The question if a given curve can be followed by the states of a (nonlinear) dynamical system and a constrained form of computable reachability is also discussed in this chapter. Finally some possible extensions and conclusions of the quantifier elimination approach in control theory are given in Chapter 10.

Part I

An Introduction to Constructive Algebra

2

Commutative Algebra and Gröbner Bases

In this chapter we will introduce some concepts and theorems from commutative algebra. We first study two basic algebraic objects called rings and fields. The reader will probably recognize the properties of these structures and think of them as abstract generalizations of some well known sets. The ring of multivariate polynomials is paid special attention to due to its usefulness in applications and importance later on in the thesis. Next we treat ideals which are subsets of rings with some special properties. One reason for studying ideals is the close relation between ideals in polynomial rings and systems of polynomial equations, which are abundant in engineering applications. The last section treats Gröbner bases which is a useful way of describing ideals. The computation of a Gröbner basis of an ideal can be seen as a generalization of Gaussian elimination for systems of linear equations to systems of polynomial equations.

An excellent first book on constructive commutative algebra which links commutative algebra and computational algebraic geometry is [23]. Another comprehensive treatment of ideal theory, especially polynomial ideals can be found in [75]. The books [45, 46, 52] and [60] are comprehensive and rigorous but also more abstract than the abovementioned.

2.1 Rings and Fields

A ring is an algebraic structure that captures the properties of the integers under the usual operations of addition and multiplication but also rational and complex numbers, polynomials and rational functions under natural addition and multiplication. Here follows a formal axiomatic definition.

Definition 2.1 *A ring is a set R together with two binary operations addition "+" and multiplication "·".*

(i) $a + (b + c) = (a + b) + c, \quad \forall a, b, c \in R$ *(associative).*

(ii) $a + b = b + a, \quad \forall a, b \in R$ *(commutative).*

(iii) There is an element denoted 0 in R such that $0 + a = a, \quad \forall a \in R$.

(iv) For every $a \in R$, there is an element $-a \in R$ such that $(-a) + a = 0$.

(v) $a \cdot (b \cdot c) = (a \cdot b) \cdot c, \quad \forall a, b, c \in R$ *(associative).*

(vi) $a \cdot b = b \cdot a, \quad \forall a, b \in R$ *(commutative).*

(vii) There is an element denoted 1 in R such that $1 \neq 0$ and

$$1 \cdot a = a \cdot 1 = a, \quad \forall a \in R.$$

(viii) Multiplication is distributive over addition, i.e.,

$$a \cdot (b + c) = a \cdot b + a \cdot c, \quad \forall a, b, c \in R.$$

Some authors define rings in a slightly different manner. The correct name for the structure above would rather be a *commutative ring with multiplicative identity.* The term commutative originates from the commutativity of the ring multiplication. In this document we will only consider commutative rings and henceforth when referring to a ring we mean a commutative ring. Note that the requirements in the definition are not independent[1].

Observe that the operations themselves are an important part of the structure. However, when the underlying operations are the natural ones we will not state them explicitely.

Example 2.1
The set of integers, $\mathbb{Z}$ is a ring.
The set of complex numbers, $\mathbb{C}$ is a ring.
The set of rational numbers, $\mathbb{Q}$ is a ring.
The set of natural numbers, $\mathbb{N}$ is not a ring.

We see that a ring is an abstract way of generalizing the properties of some well known sets and it will give us a whole new machinery to deal with such sets.

Another more sloppy definition of a ring could be: A ring is a set where we can add, subtract and multiply the elements according to the "usual" laws of these operations. We then get an element of the ring as the result, i.e., the ring is closed under its defined operations.

Since our main interest will be polynomials we now take a closer look on a special kind of rings called rings of polynomials and their construction.

[1]The commutativity of addition (ii) follows from the other postulates: apply the distributive law in two different ways to $(a + b) \cdot (1 + 1)$.

Rings of Polynomials

A polynomial of degree N in the *indeterminate* or variable x with coefficients from the ring R is a formal expression of the form

$$a_0 + a_1 x + a_2 x^2 + \ldots + a_N x^N,$$

where $a_0, a_1, a_2, \ldots, a_N \in R$ and $N \in \mathbb{N}$. The set of all such expressions is denoted $R[x]$. Note that two polynomials are equal if and only if they are of the same degree and $a_i = b_i$ for every $i \leq N$.

Let addition and multiplication in $R[x]$ between two polynomials of the same degree be defined as usual by

$$\left(\sum_{n=0}^{N} a_n x^n\right) + \left(\sum_{n=0}^{N} b_n x^n\right) = \sum_{n=0}^{N} (a_n + b_n) x^n,$$

$$\left(\sum_{n=0}^{N} a_n x^n\right)\left(\sum_{n=0}^{N} b_n x^n\right) = \sum_{r=0}^{2N} \sum_{\substack{i+j=r \\ i,j \leq N}} a_i b_j x^r.$$

If the polynomials have different degrees we just extend the lower degree polynomial with powers of x with zeros coefficients.

That $R[x]$ is a ring with addition and multiplication defined as above follows from the ring structure of R. The zero and identity element can be identified with the ones in R and it is easy to show that $R[x]$ fulfills the axioms of Definition 2.1.

It is now easy to extend a ring of polynomials in a single indeterminate to rings of polynomials in several indeterminates. We start with the ring $R[x]$ and construct a new ring exactly in the same way as we constructed $R[x]$ from the ring R above. The new ring, $R[x][y]$ will be the set of all polynomials in the indeterminate y with coefficients from $R[x]$ which is the same as the set of all polynomials in two indeterminates. Another way to denote this set is $R[x, y]$. Continuing in this way it is possible to construct rings of polynomials in an arbitrary number of indeterminates. This gives us the following definition.

Definition 2.2 *A ring of polynomials in the indeterminates* $x_1, \ldots, x_n$ *is the set of polynomials in* $x_1, \ldots, x_n$ *with coefficients from a ring* R *together with the usual addition and multiplication of polynomials. This ring is denoted* $R[x_1, \ldots, x_n]$.

When dealing with polynomials in a small number of variables we will usually use the letters x, y and z instead of indexed variables, e.g., $R[x, y, z]$.

Example 2.2 *The polynomial* $x^2 y + (1 + 2i)z$ *belongs to the ring* $\mathbb{C}[x, y, z]$.

Fields

We have not mentioned anything about division yet. When we now introduce it we will get another algebraic structure which is a special kind of ring.

Definition 2.3 *A field,* k *is a ring where every element except* 0 *has a multiplicative inverse, i.e.,*

$$\exists b : ab = ba = 1 \qquad \forall a \in k \setminus \{0\}.$$

From now on we let the symbol k denote an arbitrary field.

Example 2.3
$\mathbb{Q}$, $\mathbb{R}$ *and* $\mathbb{C}$ *are fields.*
$\mathbb{Q}[x]$ *is not a field since there is no polynomial,* $f \in \mathbb{Q}[x]$ *such that* $f\,x = 1$.

Sometimes one needs larger fields. There is a simple way of constructing an extended field.

Definition 2.4 *Let* α *be an indeterminate not in the field* k*. Then the smallest field containing* $k \cup \alpha$ *is called the field obtained by adjoining* α *to* k *and is denoted* $k(\alpha)$.

This is just a way to introduce a new symbol, which obeys the field operations, i.e., addition, subtraction, multiplication and division.

Example 2.4
The field $\mathbb{R}(i)$, *where* $i^2 = -1$ *is the field of complex numbers, i.e.,* $\mathbb{C}$.
Let k *be a field. The field* $k(t)$ *is the field of all rational functions of* t *with coefficients from* k.

After defining the above general algebraic structures we now turn to an important substructure of a ring called ideals.

2.2 Ideals

An ideal is an important substructure of a ring. It plays a central role in many applications of ring theory, e.g., finding the zeros of a system of polynomial equations.

Definition 2.5 *An ideal* I *of a ring* R *is a subset of* R *satisfying*

$$a, b \in I \;\Rightarrow\; a + b \in I \quad \textit{and} \quad r \in R,\, a \in I \;\Rightarrow\; r\,a \in I.$$

In the rest of this section we let R denote an arbitrary ring and I an arbitrary ideal in R. From the above definition it is clear that any element of the form

$$r_1 a_1 + \ldots + r_n a_n \in I$$

for any $r_1, \ldots, r_n \in R$ and $a_1, \ldots, a_n \in I$.

Let $P \subset R$. We use the notation $\langle P \rangle$ for the smallest ideal containing P and we call P a *generating set* or *basis* for the ideal in question.

Example 2.5 *It is easily verified that* $I = \{ a \in \mathbb{Z} \mid a \text{ is even} \} \subset \mathbb{Z}$ *is an ideal in* $\mathbb{Z}$*. Furthermore,* $I = \langle 2 \rangle$ *and it can be shown that all ideals in* $\mathbb{Z}$ *can be generated by a single element.*

Definition 2.6 *An ideal* I *of* R *is said to be finitely generated if there is a finite set* $P \subset R$ *such that* $I = \langle P \rangle$*.*

Note that there are always several different generating sets for one ideal. In Example 2.5 we could equally well have chosen $\{2, 4\}$ as a generating set of I.

Ideals in Polynomial Rings

Ideals in polynomial rings appears to be a very fruitful concept in the study of systems of polynomial equations. Therefore, from now on our main interest will be in rings of polynomials and ideals in such rings.

Lemma 2.1 *Let* $f_1, \ldots, f_s$ *be polynomials in* $k[x_1, \ldots, x_n]$*. The ideal generated by these polynomials is*

$$\langle f_1, \ldots, f_s \rangle = \left\{ \sum_{i=1}^{s} g_i f_i \,\middle|\, g_i \in k[x_1, \ldots, x_n] \right\}.$$

Proof *The proof follows almost immediately from the definition of an ideal, see [75].* □

Later on we need some results concerning a special kind of polynomial ideals so called *monomial ideals*. A *monomial* is an arbitrary product of the variables of the ring, e.g., $xy^2z \in k[x, y, z]$.

Definition 2.7 *An ideal* I *in* $k[x_1, \ldots, x_n]$ *is a monomial ideal if there is a generating set for* I *consisting only of monomials.*

Example 2.6 *Consider the ideal* $I = \langle x - y, y \rangle \in k[x, y, z]$*. By definition* I *consists of all polynomials* f *in* $k[x, y, z]$ *which can be expressed as*

$$f = g_1(x - y) + g_2 y = g_1 x + (g_2 - g_1) y$$

where $g_i \in k[x, y, z]$*. The expression after the last equality sign is by definition an element in* $\langle x, y \rangle$*, hence* $I \subset \langle x, y \rangle$*. The other inclusion can be proved similarly which shows that* $I = \langle x - y, y \rangle = \langle x, y \rangle$ *is a monomial ideal.*

Algebra-Geometry Connection

Above we have only considered symbolic or algebraic properties of polynomials. Next we will give a definition which links polynomials with geometry. This link is one reason for the power of the algebraic framework for systems theory.

Definition 2.8 *Let* k *be a field and* $\mathcal{F} \subset \mathsf{k}[x_1, \ldots, x_n]$ *a set of multivariate polynomials. Then we define*

$$\mathbf{V}(\mathcal{F}) \triangleq \left\{ (a_1, \ldots, a_n) \in \mathsf{k}^n \,|\, f(a_1, \ldots, a_n) = 0 \quad \forall f \in \mathcal{F} \right\}.$$

We call $\mathbf{V}(\mathcal{F})$ *the variety defined by* $\mathcal{F}$.

This is nothing else but the set of solutions of a system of polynomial equations. In applications one is often more interested in the solutions of a system of equations than the equations defining the system. This leads us to search for a representation of the solution set of a system of equations where the defining equations is not that central. We will see that this has a close relation to the ideal generated by the equations of the system.

There is a close connection between the ideal $\langle f_1, \ldots, f_s \rangle$ and the corresponding system of polynomial equations. Given $f_1, \ldots, f_s \in \mathsf{k}[x_1, \ldots, x_n]$, we can form the following system of equations

$$\begin{aligned} f_1 &= 0 \\ &\vdots \\ f_s &= 0 \end{aligned}$$

From these equations, one can derive others by algebraic manipulations. For example, if we multiply the first equation by g_1, the second by g_2, etc. where $g_i \in \mathsf{k}[x_1, \ldots, x_n]$ and then add the resulting equations, we obtain

$$g_1 f_1 + \ldots + g_s f_s = 0,$$

which is a consequence of our original system. Notice that the left hand side of this equation is exactly an arbitrary element of the ideal $\langle f_1, \ldots, f_s \rangle$ according to Lemma 2.1. Thus, we can think of $\langle f_1, \ldots, f_s \rangle$ as being the set of all polynomials corresponding to those equations that can be formed by multiplying the original equations with arbitrary polynomials and then sum such products. The important observation here is that any solution of the original system is a solution to all equations derived in this way.

Lemma 2.2 *The set of common zeros of the polynomials* $f_1, \ldots, f_s \in \mathsf{k}[x_1, \ldots, x_n]$ *is the same as the set of common zeros of all polynomials in the ideal* $\langle f_1, \ldots, f_s \rangle$ *or in other words*

$$\mathbf{V}(\{f_1, \ldots, f_s\}) = \mathbf{V}(\langle f_1, \ldots, f_s \rangle)$$

Proof *By definition any polynomial in* $\langle f_1, \ldots, f_s \rangle$ *can be written as* $\sum_{i=1}^{s} g_i f_i$ *where* $g_i \in k[x_1, \ldots, x_n]$ *and*

$$f_1 = \ldots = f_s = 0 \quad \Rightarrow \quad \sum_{i=1}^{s} g_i f_i = \sum_{i=1}^{s} g_i 0 = 0.$$

Hence a common zero of $f_1, \ldots, f_s$ *is a zero of any polynomial in* $\langle f_1, \ldots, f_s \rangle$*.*

A zero which is common to all polynomials in $\langle f_1, \ldots, f_s \rangle$ *is of course also a zero of the generators,* $f_1, \ldots, f_s$ *and the lemma follows.* □

The connection between ideals and systems of polynomial equations will be shown to be very important. It lets us explore the inherent structure of polynomial systems with the power and strength of commutative algebra.

Properties of Ideals

Ideals can be classified in a number of different ways according to their properties. In this document we only need two such classifications; prime and radical ideals.

Definition 2.9 *The ideal* $I \subset R$ *is prime if for all* $a, b \in R$

$$a\,b \in I \quad \Rightarrow \quad a \in I \quad \text{or} \quad b \in I.$$

We have the convention that R itself is not a prime ideal. Clearly an ideal in $k[x_1, \ldots, x_n]$ generated by a single polynomial is prime if and only if its generator is *irreducible*, i.e., it cannot be written as a product of two polynomials in $k[x_1, \ldots, x_n]$. In polynomial rings it depends on the coefficient ring if an ideal is prime or not.

Example 2.7 *The ideal* $\langle x^2 + y^2 \rangle$ *is prime in* $\mathbb{R}[x,y]$ *but not in* $\mathbb{C}[x,y]$*, since* $x^2 + y^2 = (x + iy)(x - iy)$*.*

The proposition that an ideal with irreducible generators is prime only holds for ideals that can be generated by a single element.

Example 2.8 *The ideal* $I = \langle x^2 + 1, y^2 + 1 \rangle \in \mathbb{R}[x,y]$ *is not prime, since*

$$1 \cdot (x^2 + 1) - 1 \cdot (y^2 + 1) = (x + y)(x - y) \ \in I$$

but $x + y \ \notin I$ *and* $x - y \ \notin I$*.*

Next we introduce an operation on ideals that may give a larger ideal as result.

Definition 2.10 *Let* I *be an ideal. The radical ideal of* I *is*

$$\sqrt{I} = \left\{ a \mid \exists m \in \mathbb{N} \text{ such that } a^m \in I \right\}.$$

It is easy to show that $\sqrt{I}$ is an ideal. Clearly $I \subset \sqrt{I}$ for any ideal I.

Example 2.9 *Consider the ideal* $I = \langle (x+1)^2, y^3 \rangle$. *The radical ideal of* I *is*

$$\sqrt{I} = \langle x+1, y \rangle.$$

Definition 2.11 *An ideal I is a radical ideal if* $\sqrt{I} = I$.

Notice that the operation of taking the radical of a polynomial ideal does not change the solution set or variety of the corresponding equations, i.e., $\mathbf{V}(I) = \mathbf{V}(\sqrt{I})$. Furthermore, a prime ideal is always radical but the reversed statement is not true in general.

The following property of chains of sets is useful to ensure that algorithms terminate. The sets A_i are said to satisfy the *ascending chain condition* if

$$A_1 \subset A_2 \subset A_3 \subset \dots \quad \Rightarrow \quad \exists i : A_{j+1} = A_j \quad \text{for all } j \geq i. \tag{2.1}$$

Definition 2.12 *A ring is Noetherian*[2] *if its ideals satisfy the ascending chain condition.*

The following example of a non-Noetherian ring can be found in [28].

Example 2.10 *Let* R *be the ring of all real-valued functions defined on the positive real axis:*

$$R = \left\{ f \mid f : \mathbb{R}^+ \to \mathbb{R} \right\}.$$

For every $j \in \mathbb{N}$

$$I_j = \left\{ f \mid f(x) = 0, \forall x \in [0, \frac{1}{j}] \right\}$$

is an ideal, and

$$I_1 \subset I_2 \subset I_3 \subset \dots$$

is a strictly ascending, infinite chain of ideals. Hence R *is not Noetherian.*

Example 2.11 *The rings* $\mathbb{Z}, \mathbb{Q}, \mathbb{R}$ *and* $\mathbb{C}$ *can all be shown to be Noetherian. In fact, any field,* k *is Noetherian since the only ideals are the so called improper ones* $\langle 0 \rangle = \{0\}$ *and* $\langle 1 \rangle = k$.

Theorem 2.1 *The ring* R *is Noetherian if and only if all ideals of* R *are finitely generated.*

Proof *See [60].* □

[2] Emmy Noether (1882-1935), one of the founders of abstract algebra.

Theorem 2.2 (Hilbert's basis theorem) *If the ring* R *is Noetherian then so is* $R[x_1, \ldots, x_n]$.

Proof *See [52].* □

This tells us that the ideals in a polynomial ring $R[x_1, \ldots, x_n]$ are finitely generated if R is Noetherian. R could for example be the ring of integers $\mathbb{Z}$ or any field k, e.g., $\mathbb{Q}, \mathbb{R}$ or $\mathbb{C}$.

2.3 Gröbner Bases

In the previous section we noted that systems of polynomial equations can be linked to algebraic objects called ideals. Here we will study a particularly useful set of generators (or basis) of an ideal called a Gröbner basis, which lets us solve many important problems concerning ideals and systems of polynomial equations algorithmically. The presentation in this section leans heavily on the one in [23].

A calculation of a Gröbner basis of an ideal almost immediately solves the following problems:

(i) The Ideal Description Problem: Does every ideal $I \subset k[x_1, \ldots, x_n]$ have a finite generating set?

(ii) The Ideal Membership Problem: Given $f \in k[x_1, \ldots, x_n]$ and an ideal $I = \langle f_1, \ldots, f_s \rangle$, determine if $f \in I$.

(iii) The Problem of Solving Polynomial Equations: Find all common solutions in k^n of a system of polynomial equations: $f_1 = f_2 = \ldots = f_s = 0$.

(iv) The Elimination Problem: Given an ideal $I \in k[x_1, \ldots, x_n]$, determine a generating set of the ideal $I \cap k[x_r, \ldots, x_n]$, where $1 \leq r \leq n$.

To understand the need for Gröbner basis we will study a generalization of the division algorithm for polynomials in one variable to the multivariable case. We start this section by considering orderings of monomials which are essential to the division algorithm.

2.3.1 Monomial Orderings

In the division algorithm for polynomials in one variable (or indeterminate) *ordering of terms*, plays an important role. It is essential that the terms of the involved polynomials are written in decreasing degree for the algorithm to work properly.

The ordering of terms becomes even more important when we consider the analog operations on polynomials in several variables. This is the reason for introducing an ordering, denoted $>$, on the monomials in $k[x_1, \ldots, x_n]$. First some convenient notations.

Definition 2.13 *Let $\alpha \in \mathbb{N}^n$. For monomials in $k[x_1, \ldots, x_n]$ we use the notation:*

$$x^\alpha = x_1^{\alpha_1} \ldots x_n^{\alpha_n}, \quad \alpha = \mathrm{mdeg}(x^\alpha), \quad |\alpha| = \sum_{i=1}^{n} \alpha_i.$$

We will refer to the mapping from the monomials of $k[x_1, \ldots, x_n]$ to $\mathbb{N}^n$ as the *multidegree* and $|\alpha|$ as the *total degree.*

Example 2.12 *Let $f = x_1^2 x_2 x_3^3 \in k[x_1, x_2, x_3, x_4]$, then*

$$f = x^{(2,1,3,0)}, \quad \mathrm{mdeg}(f) = (2,1,3,0), \quad |(2,1,3,0)| = 6$$

The multidegree defines a one-to-one mapping between the elements of $\mathbb{N}^n$ and the monomials in $k[x_1, \ldots, x_n]$. Due to this fact every ordering $>$ we establish on $\mathbb{N}^n$ automatically carries over to an ordering of the monomials in $k[x_1, \ldots, x_n]$. If $\alpha > \beta$ we will also say that $x^\alpha > x^\beta$.

However, an ordering on $\mathbb{N}^n$ has to obey some special conditions to be a useful ordering on the monomials in $k[x_1, \ldots, x_n]$. If we multiply a polynomial (with its terms written in decreasing order) with a monomial, we do not want the terms to "jump around". A critical fact in the division algorithm is that the highest ordered term still is the highest ordered after multiplication with a monomial. How to define the highest ordered term or the leading term of a polynomial in $k[x_1, \ldots, x_n]$ will be shown later. If $x^\alpha > x^\beta$ and x^γ is any monomial we require that $x^\alpha x^\gamma > x^\beta x^\gamma$. This prevents the terms from "jumping around" as mentioned above.

Definition 2.14 *A monomial ordering on $\mathbb{N}^n$ is any relation $>$ between elements in $\mathbb{N}^n$ or equivalently, any relation between the monomials $x^\alpha, \alpha \in \mathbb{N}^n$, which satisfies:*

(i) $>$ is a total ordering on $\mathbb{N}^n$, i.e., exactly one of $\alpha > \beta$, $\alpha = \beta$, $\beta > \alpha$ should be true.

(ii) If $\alpha > \beta$ and $\gamma \in \mathbb{N}^n$, then $\alpha + \gamma > \beta + \gamma$ (prevents from "jumping around").

(iii) $>$ is a well ordering on $\mathbb{N}^n$. This means that every nonempty subset of $\mathbb{N}^n$ has a smallest element under $>$.

There are many ways of defining orderings of monomials in several variables. We only consider the *lexicographic* order (or lex order, for short). For other orderings see, e.g., [23] or [1].

Definition 2.15 *The lexicographic term ordering $>_{\text{lex}}$ is defined by*

$$\alpha >_{\text{lex}} \beta \quad \Leftrightarrow \quad \exists j : \alpha_j > \beta_j, \quad \forall i < j : \alpha_i = \beta_i$$

It can be shown that this actually defines a monomial ordering. See for example [23, page 56].

Example 2.13

$$(1,2,3) >_{\text{lex}} (0,4,5) \quad\Leftrightarrow\quad x_1x_2^2x_3^3 >_{\text{lex}} x_2^4x_3^5$$
$$(1,2,3) >_{\text{lex}} (1,2,2) \quad\Leftrightarrow\quad x_1x_2^2x_3^3 >_{\text{lex}} x_1x_2^2x_3^2$$

$$(1,0,\ldots,0) >_{\text{lex}},\ldots,>_{\text{lex}} (0,0,\ldots,1) \quad\Leftrightarrow\quad x_1 >_{\text{lex}},\ldots,>_{\text{lex}} x_n$$

To decide if a $\alpha >_{\text{lex}} \beta$ we just compare the elements in α and β from left to right. The first non-equal pair will determine the order. Sometimes the letters x, y, z are used instead of x_1, x_2, x_3. The above example then shows that $x >_{\text{lex}} y >_{\text{lex}} z$ and the lexicographic ordering will be the same as the one used in dictionaries, which explains its name.

Given a polynomial $f = \sum_\alpha a_\alpha x^\alpha$ in $k[x_1,\ldots,x_n]$ and a monomial ordering we can arrange the terms of f in an unambiguous way.

Example 2.14 *Let $f = xy^2z^3 + 5y^4z^5 - 3x^2$. If we order the terms in decreasing order using* lex *order we will get $f = -3x^2 + xy^2z^3 + 5y^4z^5$.*

We close this subsection by defining a few notations concerning ordering of the terms in polynomials.

Definition 2.16 *Let $f = \sum_\alpha a_\alpha x^\alpha \in k[x_1,\ldots,x_n]\setminus\{0\}$ and let $>$ be a monomial ordering.*

(i) The multidegree of f is $\text{mdeg}(f) \triangleq \max(\alpha \in \mathbb{N}^n : a_\alpha \neq 0)$

(ii) The leading coefficient of f is $\text{lc}(f) \triangleq a_{\text{mdeg}(f)} \in k$

(iii) The leading monomial of f is $\text{lm}(f) \triangleq x^{\text{mdeg}(f)}$

(iv) The leading term of f is $\text{lt}(f) \triangleq \text{lc}(f)\cdot\text{lm}(f)$.

Example 2.15 *Let $f = -3x^2z + xy^2z^3 + 5y^4z^5$, then using* lex *order*

$$\text{mdeg}(f) = (2,0,1),\quad \text{lc}(f) = -3,\quad \text{lm}(f) = x^2z,\quad \text{lt}(f) = -3x^2z.$$

2.3.2 Division in $k[x_1, \ldots, x_n]$

As a prelude to Gröbner bases we introduce a generalization of the division algorithm for polynomials in a single variable to the multivariate case. We want to divide $f \in k[x_1, \ldots, x_n]$ by $f_1, \ldots, f_s$, which analogously to the single variable case means that f can be expressed as

$$f = g_1 f_1 + \ldots + g_s f_s + r,$$

where the "quotients" g_i and the remainder r belong to $k[x_1, \ldots, x_n]$.

The basic idea is the same as in $k[x]$: In each step of the algorithm we try to cancel the leading term of the latest achieved polynomial by multiplying some f_i by an appropriate monomial and subtract. As in the single variable case we have to order the terms in the polynomials in decreasing order using lex order.

Example 2.16 *We will divide* $f = x^2y + y^2$ *by* $f_1 = xy - 1$ *and* $f_2 = y - 1$ *where the terms of the polynomials are ordered using lex order with* $x > y$*. The divisors* f_1, f_2 *and the quotients* g_1, g_2 *are listed vertically, so we start with*

$$\begin{array}{rl}
g_1: & \\
g_2: & \\
xy-1 & \\
y-1 & \sqrt{x^2y+y^2}
\end{array}$$

Analogously to the single variable case we should try to cancel the leading term of f*. Here this can be done in two ways, since* $\mathrm{lt}(f)$ *is divisible by both* $\mathrm{lt}(f_1)$ *and* $\mathrm{lt}(f_2)$*. We use* f_1 *which gives*

$$\begin{array}{rr}
g_1: & x \\
g_2: & \\
xy-1 & x^2y+y^2 \\
y-1 & \underline{x^2y-x} \\
 & x+y^2
\end{array}$$

Now, it is not possible to cancel the leading term of the partial remainder since neither $\mathrm{lt}(f_1) = xy$ *nor* $\mathrm{lt}(f_2) = y$ *divides* x*. Despite this* $x + y^2$ *can be simplified further since* $\mathrm{lt}(f_2) = y$ *divides* y^2*. This never occurs in the single variable case since the algorithm stops when the leading term of the divisor no longer divides the leading term of the dividend. We now create a new column where we put terms that belongs to the remainder, in this case* x*.*

$$\begin{array}{rrcc}
g_1: & x & & \underline{r} \\
g_2: & & & \\
xy-1 & x^2y+y^2 & & \\
y-1 & \underline{x^2y-x} & & \\
 & \underline{x+y^2} & & \\
 & y^2 & \rightarrow & x
\end{array}$$

Now we use f_2 *to cancel* y^2.

$$
\begin{array}{rlcl}
g_1: & x & & \underline{r} \\
g_2: & y & & \\
xy-1 & \overline{x^2y+y^2} & & \\
y-1 & \underline{x^2y-x} & & \\
 & \underline{x+y^2} & & \\
 & y^2 & \rightarrow & x \\
 & \underline{y^2-y} & & \\
 & y & &
\end{array}
$$

After the last two steps we get

$$
\begin{array}{rlcl}
g_1: & x & & \underline{r} \\
g_2: & y+1 & & \\
xy-1 & \overline{x^2y+y^2} & & \\
y-1 & \underline{x^2y-x} & & \\
 & \underline{x+y^2} & & \\
 & y^2 & \rightarrow & x \\
 & \underline{y^2-y} & & \\
 & y & & \\
 & \underline{y-1} & & \\
 & \underline{1} & & \\
 & 0 & \rightarrow & x+1
\end{array}
$$

The algorithm stops when we get zero under the line. Hence $f = x^2y + y$ *can be written in the form* $f = x(xy-1)+(y+1)(y-1)+x+1$. *Notice that the remainder is a sum of monomials, none of which is divisible by the leading terms of* f_1 *and* f_2.

In the first step above we made a choice between using f_1 or f_2. This freedom will in general lead to different final expressions. In this case we would have ended up with the representation $f = 0(xy-1) + (x^2+y+1)(y-1) + x^2 + 1$ of f if we had used f_2 instead of f_1. We notice the important fact that the remainder is not uniquely determined as in the single variable case.

The division algorithm for multivariate polynomials presented in the above example always terminates since the multidegree of the partial remainder drops in every step and according to property (iii) in Definition 2.14 this cannot go on for ever.

A natural question is whether this algorithm solves the ideal membership problem? In the single variable case this is true. By repeated use of the division algorithm a single generator of any ideal in $k[x]$ can be found (the greatest common divisor of the original generators). A polynomial belongs to an ideal in $k[x]$ if and only if the remainder on division by this single generator is zero (the remainder is unique in this case).

Do we always get a remainder equal to zero in the multivariate case too? Clearly, if the polynomial is written $f = g_1f_1 + \ldots + g_sf_s$ then it belongs to the ideal

$\langle f_1, \ldots, f_s \rangle$. The question is whether $f \in \langle f_1, \ldots, f_s \rangle$ implies $r = 0$ when the division algorithm is used. The answer is no which we demonstrate with an example.

Example 2.17 *Let* $f = x^2y - xy^2$, $f_1 = y^2 - 1$ *and* $f_2 = xy - 1$. *Dividing* f *by* f_1, f_2, *using* lex *order, gives the result* $f = -x(y^2 - 1) + x(xy - 1)$ *which implies that* $f \in \langle f_1, f_2 \rangle$. *On the other hand, by cancelling terms in another order during the division we obtain* $f = (x - y)(xy - 1) + x - y$ *where the remainder is not equal to zero and we cannot draw the above conclusion from this calculation.*

The example points at a big disadvantage with the division algorithm for polynomials in several variables. It does not solve the membership problem. Since we are studying ideals one can ask whether there might be another generating set for a given ideal, such that the division algorithm always gives a zero remainder if f is a member of the ideal? The answer to this question is yes and we will see that Gröbner bases solve the problem.

2.3.3 Gröbner Bases

According to Hilbert's basis Theorem 2.2 every ideal $I \subset k[x_1, \ldots, x_n]$ has a finite basis. In this subsection we introduce an especially useful basis of a polynomial ideal called a Gröbner basis. We start with a definition, where $\mathrm{lt}(I)$ denotes the set of all leading terms of the polynomials in the ideal I.

Definition 2.17 *A finite subset* $\mathbf{G} = \{g_1, \ldots, g_t\}$ *of an ideal* I *is said to be a Gröbner basis if*

$$\langle \mathrm{lt}(g_1), \ldots, \mathrm{lt}(g_t) \rangle = \langle \mathrm{lt}(I) \rangle.$$

Equivalently, a set $\mathbf{G} = \{g_1, \ldots, g_t\} \subset I$ is a Gröbner basis of I if and only if the leading term of any element of I is divisible by one of the $\mathrm{lt}(g_i)$.

If we impose some further conditions on what we have defined as a Gröbner basis it can be shown that every ideal, except $\langle 0 \rangle$, has an unique Gröbner basis obeying these conditions.

Definition 2.18 *A reduced Gröbner basis of a polynomial ideal* I *is a Gröbner basis* $\mathbf{G}$ *for* I *such that:*

(i) $\mathrm{lc}(p) = 1 \quad \forall p \in \mathbf{G}$.

(ii) No monomial of p *lies in* $\langle \mathrm{lt}(\mathbf{G} \setminus p) \rangle \quad \forall p \in \mathbf{G}$.

Theorem 2.3 *Fix a monomial ordering. Then every polynomial ideal* I *has a unique reduced Gröbner basis.*

Proof *See [23].* □

Hence, this gives us a unique description of an ideal once a monomial ordering has been chosen. Unless otherwise is stated, we mean a reduced Gröbner basis whenever talking about Gröbner bases in the rest of this document. A reduced Gröbner basis of an ideal I is denoted $\mathrm{GB}(I)$.

One important property of Gröbner bases is presented in the following theorem.

Theorem 2.4 *Let* $I \subset k[x_1, \ldots, x_n]$ *be a polynomial ideal,* $\mathrm{GB}(I) = \{g_1, \ldots, g_t\}$ *and* $f \in k[x_1, \ldots, x_n]$*. Then there is a unique* $r \in k[x_1, \ldots, x_n]$ *with the following properties:*

(i) No term of r *is divisible by any of the* $\mathrm{lt}(g_i)$*.*

(ii) There is a $g \in I$ *such that* $f = g + r$*.*

Proof *See [23].* □

This theorem tells us that we get the same remainder in the division algorithm irrespectively of how the elements in $\mathrm{GB}(I)$ are listed. Hence we can speak of the remainder r of a polynomial f modulo the ideal $\langle f_1, \ldots, f_s \rangle$.

In the previous section we saw that even if a polynomial was a member of an ideal the remainder, when using the division algorithm, did not have to be zero. Gröbner bases removes this ambiguity.

Definition 2.19 *We write* $\mathrm{rem}(f, F)$ *for the remainder on division of* f *by the* s*-tuple* $F = (f_1, \ldots, f_s)$ *using* lex *order.*

We observe that if $F = \mathrm{GB}(I)$, then we can regard F as a set without any particular order according to Theorem 2.4.

Theorem 2.5 *Let* $\mathrm{GB}(I) = \{g_1, \ldots, g_t\}$ *and* $f \in k[x_1, \ldots, x_n]$*. Then*

$$f \in I \quad \Leftrightarrow \quad \mathrm{rem}(f, \mathrm{GB}(I)) = 0.$$

Proof *See [23].* □

This theorem gives a solution to the membership problem once we have a Gröbner basis of the ideal. Just compute $\mathrm{rem}(f, \mathrm{GB}(I))$.

Buchberger's Algorithm

Given a set of generators, f_i of an ideal $I = \langle f_1, \ldots, f_s \rangle$. How does one compute a Gröbner basis $\mathrm{GB}(I) = \{g_1, \ldots, g_t\}$?

In the previous subsection we noted that the remainder of f on division with a sequence of polynomials $(f_1, \ldots, f_s)$ is not uniquely determined. This was not the case if the sequence is a Gröbner basis according to Theorem 2.4. The ambiguity is introduced in the steps of the division algorithm where more than one $\mathrm{lt}(f_i)$ can be used to cancel $\mathrm{lt}(f)$. This observation together with the insight that it may be possible to construct polynomials in I with $\mathrm{lt}(p)$ of lower order than $\mathrm{lt}(f_i)$ gives a hint on how to enlarge $\{f_1, \dots, f_s\}$ to a Gröbner basis of $\langle f_1, \dots, f_s \rangle$. The key ingredient is so called S-polynomials.

Definition 2.20 *Let $f, g \in k[x_1, \dots, x_n]$ and $\alpha, \beta, \gamma \in \mathbb{N}^n$, where $\alpha = \mathrm{mdeg}(f)$, $\beta = \mathrm{mdeg}(g)$, $\gamma_i = \max(\alpha_i, \beta_i)$. We call x^γ the least common multiple of $\mathrm{lm}(f)$ and $\mathrm{lm}(g)$, denoted $x^\gamma = \mathrm{lcm}(\mathrm{lm}(f), \mathrm{lm}(g))$.*

Example 2.18 *Let $f = 3x^2yz^2 - xy^2$ and $g = 2x^3y + xy$. Then $\mathrm{lm}(f) = x^2yz^2$ and $\mathrm{lm}(g) = x^3y$ which gives $\alpha = (2,1,2)$ and $\beta = (3,1,0)$. Hence $\gamma = (3,1,2)$ so $\mathrm{lcm}(\mathrm{lm}(f), \mathrm{lm}(g)) = x^3yz^2 = x \cdot \mathrm{lm}(f) = \mathrm{lm}(g) \cdot z^2$.*

In other words, the least common multiple is the monomial of least multidegree which contains $\mathrm{lm}(f)$ and $\mathrm{lm}(g)$ as factors, not necessarily both at the same time.

Definition 2.21 *Let $f, g \in k[x_1, \dots, x_n]$. The S-polynomial of f and g is*

$$S(f,g) = \frac{\mathrm{lcm}(\mathrm{lm}(f), \mathrm{lm}(g))}{\mathrm{lt}(f)} \cdot f - \frac{\mathrm{lcm}(\mathrm{lm}(f), \mathrm{lm}(g))}{\mathrm{lt}(g)} \cdot g$$

We observe that if $f, g \in I$ are two generators of the ideal I then $S(f,g) \in I$. The S-polynomial is constructed to cancel the leading terms of f and g. Hence it may have a leading term of lower order than f and g but not necessarily. This is maybe best understood by an example.

Example 2.19 *Let f and g be the same as in the previous example. Then,*

$$\begin{aligned} S(f,g) &= \frac{x^3yz^2}{3x^2yz^2} \cdot (\underline{3x^2yz^2} - xy^2) - \frac{x^3yz^2}{2x^3y} \cdot (\underline{2x^3y} + xy) \\ &= \underline{\frac{x}{3} \cdot 3x^2yz^2} - \frac{x^2y^2}{3} - \underline{\frac{z^2}{2} \cdot 2x^3y} - \frac{xyz^2}{2} \\ &= -\frac{1}{3}x^2y^2 - \frac{1}{2}xyz^2. \end{aligned}$$

Notice the cancelation of the underlined terms. The S-polynomial of f and g is constructed to produce cancelation of the leading terms of f and g.
Let $h = 6x^3y^2z^2 + 1$ and divide h by (f, g). In the first step of the division algorithm we can use either f or g to cancel $\mathrm{lt}(h)$

$$\begin{aligned} h - 2xy \cdot f &= \underline{6x^3y^2z^2} + 1 - 2xy(\underline{3x^2yz^2} - xy^2) = 2x^2y^2 + 1 \\ h - 3yz^2 \cdot g &= \underline{6x^3y^2z^2} + 1 - 3yz^2(\underline{2x^3y} + xy) = -3xy^2z^2 + 1 \end{aligned}$$

The ambiguity introduced in this step is

$$(h - 2xy \cdot f) - (h - 3yz^2 \cdot g) = -6y(\frac{x}{3}f - \frac{z^2}{2}g) = -6y\, S(f,g).$$

It can be shown that this is the only way ambiguity can be introduced in the division algorithm. To remove this problem it is natural to include the S*-polynomials (or in fact reduced versions of them) among the divisors.*

The S-polynomials let us decide if an ideal basis also is a Gröbner basis.

Theorem 2.6 *Let* $I = \langle F \rangle$ *be a polynomial ideal, where* $F = \{f_1, \ldots, f_s\}$*. Then*

$$F = GB(I) \qquad \Leftrightarrow \qquad \text{rem}(S(f_i, f_j), F) = 0, \quad i \neq j.$$

Proof *See [23].* □

Example 2.20 *Let* $I = \langle f_1, f_2 \rangle$*, where* $f_1 = x^2 - x$ *and* $f_2 = x - y$*. We want to decide if* $\{f_1, f_2\}$ *is a Gröbner basis of* I*. Computing the corresponding* S*-polynomial gives*

$$S(f_1, f_2) = \frac{x^2}{x^2} \cdot (x^2 - x) - \frac{x^2}{x} \cdot (x - y) = xy - x$$

and the remainder on division by f_1 *and* f_2 *is*

$$\text{rem}(S(f_1, f_2), \{f_1, f_2\}) = y^2 - y.$$

Since we get a nonzero remainder Theorem 2.6 tells us that $\{f_1, f_2\}$ *is not a Gröbner basis of* I*.*

We can now describe an algorithm due to Buchberger for computing the Gröbner basis of an ideal.

Let $I = \langle F \rangle$, where $F = (f_1, \ldots, f_s)$. Calculate the S-polynomial for all pairs f_i, f_j and their remainders, $\text{rem}(S(f_i, f_j), F)$ until there is a nonzero remainder. Add this remainder to the generating set and continue. Eventually the process stops and the generating set fulfills the conditions in Theorem 2.6. The algorithm terminates since the corresponding sequence of monomial ideals generated by the leading monomials of the successively enlarged basis forms an ascending chain. According to the ascending chain condition (2.1) this sequence of monomial ideals saturates and the algorithm terminates.

Example 2.21 *We continue the previous example and calculate a Gröbner basis for* I*. We include the remainder* $f_3 = y^2 - y$ *in the generating set. Now* $\{f_1, f_2, f_3\}$ *gives us three* S*-polynomials to investigate*

$$\begin{aligned} S_1 &= S(f_1, f_2) = xy - x, \quad \text{rem}(S_1, F) = 0, \\ S_2 &= S(f_1, f_3) = x^2y - xy^2, \quad \text{rem}(S_2, F) = 0, \\ S_3 &= S(f_2, f_3) = xy - y^3, \quad \text{rem}(S_3, F) = 0. \end{aligned}$$

Hence, $\{x^2 - x, x - y, y^2 - y\}$ *is a Gröbner basis of* I*, but it is not reduced.*

The next theorem solves the problem of elimination, i.e., given a system of polynomial equations, what conditions does it impose on a given subset of the variables?

Theorem 2.7 (The Elimination Theorem)
Let $I \subset k[x_1, \ldots, x_n]$ *be an ideal and* $\mathrm{GB}(I)$ *a Gröbner basis of* I *with respect to lex order where* $x_1 > x_2 > \ldots > x_n$. *Then for every* $1 \leq r \leq n$

$$I \cap k[x_r, \ldots, x_n] = \langle \mathrm{GB}(I) \cap k[x_r, \ldots, x_n] \rangle.$$

Proof *See [23].* □

Example 2.22 *Consider the following system of equations*

$$\begin{aligned} z^2 - 5xz - xy^2 + 2x^2y + 3y &= 0, \\ xy^2 - 2x^2y + x^2 - 2x + 1 &= 0, \\ xz^2 - 5x^2z + 3xy &= 0 \end{aligned} \tag{2.2}$$

and let $f_1, \ldots, f_3$ *denote the corresponding polynomials. We compute a Gröbner basis of the ideal* $\langle f_1, f_2, f_3 \rangle$ *w.r.t. the order* $z > y > x$ *using the function* `gbasis` *in* MAPLE

$$\mathrm{GB}(\langle f_1, f_2, f_3 \rangle) = \{ z^2 - 5zx + 3y,\ y^2 - 2xy,\ x^2 - 2x + 1 \}.$$

According to Lemma 2.2 the polynomials in the Gröbner basis have the same common zeros as f_i , *which means that the following system is equivalent to (2.2).*

$$\begin{aligned} z^2 - 5xz - 3y &= 0, \\ y^2 - 2xy &= 0, \\ x^2 - 2x + 1 &= 0. \end{aligned} \tag{2.3}$$

The simple "triangular" structure of system (2.3) follows from Theorem 2.7 and makes it easy to solve by back substitution.

We have seen that Gröbner bases solve the problems we posed in the beginning of the section. The ideal description problem is solved by computing the unique reduced Gröbner basis of an ideal. The ideal membership problem is solved by applying polynomial division w.r.t. a Gröbner basis of the ideal. We get a zero remainder if and only if the polynomial belongs to the ideal. Solving systems of polynomial equations and elimination can be carried out according to Theorem 2.7.

3

Differential Algebra and Characteristic Sets

Many physical systems can be modeled by a set of polynomial differential equations. We have seen that Gröbner bases are an excellent tool to deal with ideals in rings of polynomials and hence systems of polynomial equations. Is it possible to treat systems of polynomial differential equations in a similar way as in the non-differential case? What differs is the concept of derivation, which we will see leads to a more complex theory. In this chapter we give a brief introduction to this subject, called differential algebra. The standard references are [72] and [50] but for most of our needs the material on the subject in [27, 28] is enough.

The chapter is organized as follows. In Section 3.1 we define the formal derivation operator, differential rings and fields. Differential ideals and some of their properties are treated in Section 3.2. Finally, characteristic sets and division in differential rings are presented in Section 3.3.

3.1 Rings and Fields

Two main concepts of differential algebra are differential rings and differential fields, respectively. To see how they relate to the definitions of rings and fields in commutative algebra we need to define a formal derivation.

Definition 3.1 *Let* R *be a ring. A derivation is a operator* $\partial : \mathsf{R} \to \mathsf{R}$ *such that*

$$\forall x, y \in \mathsf{R}: \qquad \partial(x+y) = \partial x + \partial y, \quad \partial(xy) = (\partial x)y + x(\partial y).$$

Example 3.1 *Consider the ring* $R = k[x]$. *Then we could define* $\partial \triangleq \frac{d}{dx}$ *as the usual derivative w.r.t.* x. *This operator satisfies Definition 3.1 and is thus a derivation. For example* $\partial(x^2+3x) = 2x+3 = \partial(x^2)+\partial(3x)$. *Another way of defining a derivation on* R *is to consider* x *as a function of* t *and define* $\partial \triangleq \frac{d}{dt}$. *Then* $\partial(x^2+3x) = 2x\frac{dx}{dt}+3\frac{dx}{dt} = x\,\partial x + \partial x\, x + 3\partial x$.

Notice the difference between this formal algebraic definition of derivation and the one in calculus. In the algebraic definition no limit process is involved.

Definition 3.2 *The elements* $a \in R$ *for which* $\partial a = 0$ *are called constants.*

Example 3.2 *From the definition of a derivation we have*

$$\partial 1 = \partial(1\cdot 1) = \partial(1)\cdot 1 + 1\cdot\partial(1) = \partial 1 + \partial 1 \quad \Leftrightarrow \quad \partial 1 = 0.$$

Definition 3.3 *A ring* R *is said to be a differential ring if there is a derivation defined on* R *and* R *is closed under derivation.*

A differential ring can also be described as a set where we can add, subtract, multiply and differentiate elements. The results of these operations will still be elements of the set.

After renaming the derivatives, a differential ring can be considered as a ring in infinitely many variables. Since there are infinitely many variables there is also infinite, strictly ascending chains of ideals, e.g.,

$$\langle z_1 \rangle \subset \langle z_1, z_2 \rangle \subset \langle z_1, z_2, z_3 \rangle \subset \dots$$

Differential rings are sometimes referred to as non-Noetherian.

Definition 3.4 *A field* k *is said to be a differential field if there is a derivation defined on* k *and* k *is closed under derivation.*

Example 3.3 *The fields* $\mathbb{Q}, \mathbb{R}$ *and* $\mathbb{C}$ *are differential fields where all elements are constants.*

A ring of polynomials in $x_1, \dots, x_n$ can be extended to a differential ring if the variables are considered as functions of some variable t and the derivation is defined by $\partial \triangleq \frac{d}{dt}$. This differential ring is called the ring of differential polynomials in $x_1, \dots, x_n$ and is denoted $k\{x_1, \dots, x_n\}$. In many applications the dot notation is used to denote derivations w.r.t. time, i.e.,

$$\partial x_1 = \frac{d}{dt}x_1 = \dot{x}_1, \quad \partial^2 x_1 = \frac{d^2}{dt^2}x_1 = \ddot{x}_1 \quad \text{and} \quad \partial^\nu x_1 = \frac{d^\nu}{dt^\nu}x_1 = x_1^{(\nu)}, \nu > 2.$$

Example 3.4 *The differential polynomials*

$$p_1 = 3\dot{x}_1 x_2 + \ddot{x}_3^3, \quad p_2 = x_2^{(4)} x_2 x_3^2 + 2x_1 + 1$$

are examples of elements in $\mathbb{Q}\{x_1, \dots, x_n\}$. *Derivatives of elements in* $\mathbb{Q}\{x_1, \dots, x_n\}$ *is easily calculated using the chain rule, e.g.,*

$$\dot{p}_1 = 3\ddot{x}_1 x_2 + 3\dot{x}_1 \dot{x}_2 + 3\ddot{x}_3^2 x_3^{(3)}.$$

As in the non-differential case it is possible to consider extensions of differential fields.

Definition 3.5 *Let* α *be an indeterminate not in the differential field* k. *Then the smallest differential field containing* $k \cup \alpha$ *is called the differential field obtained by adjoining* α *to* k *and is denoted* $k\langle\alpha\rangle$.

Example 3.5 *The differential field extension* $\mathbb{Q}\langle u\rangle$ *of the rational numbers, where* $u \notin \mathbb{Q}$ *is an indeterminate, consists of all rational functions in* u *and its derivatives. The differential ring* $\mathbb{Q}\langle u\rangle\{x, y\}$ *is the set of all differential polynomials in* x *and* y *with coefficients that are quotients of differential polynomials in* u, *e.g.,*

$$\frac{2\dot{u}+1}{1+u^2} x\ddot{y} - \dot{u}^2\dot{x} + 1 \in \mathbb{Q}\langle u\rangle\{x, y\}.$$

3.2 Ideals

As for rings and fields there are objects in differential algebra corresponding to ideals called differential ideals.

Definition 3.6 *A differential ideal is an ideal, which is closed under derivation.*

We now want to describe differential ideals in a similar way as we did with ideals in the non-differential case, namely with generating sets.

Consider a set of differential polynomials $P \subset k\{x_1, \dots, x_n\}$. We use the notation $[P]$ for the smallest differential ideal containing P and we call P a generating set for the differential ideal in question. The differential ideal $[P]$ is formed in the same way as $\langle P\rangle$ except that we also allow derivation of elements as well.

We saw above that the ring of polynomials $k[x_1, \dots, x_n]$ considered as functions of t together with the derivation $\partial \triangleq \frac{d}{dt}$, was a differential ring, $k\{x_1, \dots, x_n\}$. Let $P \in k\{x_1, \dots, x_n\}$ be a set of differential polynomials. Then the elements of $[P]$ is the same as the elements of

$$[P] = \langle \{ p, \dot{p}, \ddot{p}, \dots \mid p \in P \} \rangle \in k[x_1, \dots, x_n, \dot{x}_1, \dots, \dot{x}_n, \dots],$$

i.e., a differential ideal can be seen as a non-differential ideal in a non-differential ring in infinitely many variables.

Example 3.6 *Consider the differential ideal*

$$[x^2] = \langle x^2, 2x\partial x, 2x\partial^2 x + 2(\partial x)^2, \dots \rangle.$$

If $x = x(t)$ *and* $\partial = \frac{d}{dt}$ *denoted by a dot, we get*

$$[x^2] = \langle x^2, 2x\dot{x}, 2x\ddot{x} + 2(\dot{x})^2, \dots \rangle.$$

We can classify ideals in an analogous way of what we did in the non-differential case.

Definition 3.7 *A prime differential ideal is a prime ideal, which is closed under derivation.*

We adopt the convention that a differential ring R is not considered to be a prime differential ideal itself.

Definition 3.8 *A radical differential ideal is a radical ideal, which is closed under derivation.*

In Section 2.2 we saw that to every system of polynomial equations we could associate a polynomial ideal. This allowed us to use the framework of commutative algebra to draw conclusions about for example solutions of systems, equivalent systems and elimination problems.

In the same way we now associate a differential ideal to every system of polynomial differential equations and hope that differential algebra will give us answers to similar questions as posed above.

So far we have not seen any major differences between commutative and differential algebra. Unfortunately, differential ideals in differential rings do not behave as "well" as ideals in non-differential rings.

Theorem 3.1 *The differential ideals in* $k\{x_1, \dots, x_n\}$ *do not satisfy the ascending chain condition.*

Proof *According to [47, page 45] and [65] the chain*

$$[x^2] = I_0 \subset I_1 \subset I_2 \subset \dots,$$

where

$$I_j = [x^2, (\partial x)^2, (\partial^2 x)^2, \dots, (\partial^j x)^2], \quad j > 0$$

is an infinite strictly ascending chain of differential ideals.

The differential ideal $[x^2]$ *is also treated in [54] and [62].* □

This theorem tells us that the differential ring $k\{x_1, \dots, x_n\}$ is non-Noetherian, which leads to the fact that there are differential ideals which are not finitely generated. We will explore this fact in Chapter 6 where we also examine other messy properties of differential ideals, especially those ideals arising from state space descriptions of dynamical systems.

3.3 Characteristic Sets

In this section we present a way of describing sets of differential polynomials, especially differential ideals. The method of characteristic sets, developed by J.F.Ritt[1] is such a description and can be found in [72]. The material in this section mainly comes from [36], which is a good introduction to the subject.

We will consider variables which are time dependent and denote derivation with respect to time with a dot or $\frac{d}{dt}$. Higher order derivatives is denoted with superscripts within parentheses in the usual way, i.e.,

$$x, \dot{x}, \ddot{x}, x^{(3)}, x^{(4)}, \ldots$$

3.3.1 Differential Polynomials and Rankings

In differential algebra we often consider systems of polynomial differential equations, that is

$$p_1 = 0, \ldots, p_s = 0,$$

where $p_1, \ldots, p_s$ are *differential polynomials*. By a differential polynomial we mean a polynomial in a finite number of variables and their derivatives. In the framework of differential algebra a differential polynomial is an element of the differential ring $k\{x_1, \ldots, x_n\}$. In this context we sometimes use the term polynomial but still mean differential polynomial.

Example 3.7 *The following are examples of differential polynomials in* x, y, z

$$\dot{x}y^2 + z, \quad \frac{d}{dt}(\dot{x}y^2 + z) = \ddot{x}y^2 + 2\dot{x}y\dot{y} + \dot{z}, \quad 5\dot{x}\ddot{y}z + 17 \quad \in \mathbb{Q}\{x, y, z\}.$$

In the non-differential case an ordering on the variables and monomials was essential to the development of algorithms such as the division algorithm. To develop a similar algorithm for differential polynomials we also need some kind of ordering. In the differential case the conventional term is *ranking*.

A differential equation is often considered more complex if it involves high order derivatives. For one variable it seems natural to rank the derivatives according to

$$x < \dot{x} < \ddot{x} < x^{(3)} < x^{(4)} < \ldots$$

If we have two variables we can choose between, e.g.,

$$x < y < \dot{x} < \dot{y} < \ddot{x} < \ddot{y} < \ldots$$

or

$$x < \dot{x} < \ddot{x} < \ldots < y < \dot{y} < \ddot{y} < \ldots$$

In the algorithms we will study, it turns out that the ranking has to obey two conditions:

[1] Joseph Fels Ritt (1893-1951), American mathematician.

(i) $x^{(\nu)} < x^{(\nu+\sigma)}, \quad \nu, \sigma \in \mathbb{N}$

(ii) $x^{(\mu)} < y^{(\nu)} \quad \Rightarrow \quad x^{(\mu+\sigma)} < y^{(\nu+\sigma)}, \quad \mu, \nu, \sigma \in \mathbb{N}$

The reason for imposing these conditions are similar to the one in the non-differential case. It prevents terms from "jumping around" after the derivation operator has been applied to a differential polynomial.

Example 3.8 *If we consider three variables x, y, z we have many possible rankings, e.g.,*

$$x < y < z < \dot{x} < \dot{y} < \dot{z} < \ddot{x} < \ddot{y} < \ddot{z} < \dots \tag{3.1}$$

$$x < \dot{x} < \ddot{x} < \dots < y < \dot{y} < \ddot{y} < \dots < z < \dot{z} < \ddot{z} < \dots \tag{3.2}$$

$$x < \dot{x} < \ddot{x} < \dots < y < z < \dot{y} < \dot{z} < \ddot{y} < \ddot{z} < \dots \tag{3.3}$$

In algorithms one often generates sequences of derivatives of strictly descending order. The next result is very useful when showing that such an algorithm eventually terminates.

Theorem 3.2 *A sequence of derivatives, each one ranked lower than the preceding one, can only have a finite length.*

Proof *See [36].* □

The ranking of variables is easy to extend to a ranking of differential polynomials.

Definition 3.9 *The leader of a differential polynomial is the highest ranked derivative. The corresponding variable is called the leading variable.*

Example 3.9 *Let the ranking be given according to (3.2) and consider the differential polynomials*

$$f = x\ddot{y} + \dot{z}, \quad \mathrm{ld}(f) = \dot{z} \text{ and } \mathrm{lv}(f) = z,$$

$$g = x\dot{x} + (\ddot{y})^3, \quad \mathrm{ld}(g) = \ddot{y} \text{ and } \mathrm{lv}(g) = y,$$

where we have introduced the notation ld *and* lv *for the leader and leading variable, respectively.*

Definition 3.10 *Let* f *and* g *be two differential polynomials.*

(i) If f *and* g *has different leaders, the one with highest ranked leader is ranked highest.*

(ii) If f *and* g *has the same leader, the one with the highest power of the leader is ranked highest.*

Example 3.10 *Let the ranking be given according to (3.2). Then,*
$x\ddot{y} + \dot{z}$ *is higher ranked than* $x\dot{x} + (\ddot{y})^3$, *since* $\ddot{y} < \dot{z}$
$x\dot{x} + (\ddot{y})^3$ *is higher ranked than* $x^2\ddot{x} + y(\ddot{y})^2$, *since* $(\ddot{y})^3$ *has a higher power than* $(\ddot{y})^2$
$x\dot{x} + (\ddot{y})^3$ *has the same rank as* $\ddot{x} + (\ddot{y})^3$, *since neither of the conditions of Definition 3.10 is fulfilled.*

For differential polynomials there is a result similar to Theorem 3.2.

Theorem 3.3 *A sequence of differential polynomials, each one ranked strictly lower than the preceding one, can only have a finite length.*

Proof *See [36].* □

Later on we want to compare two differential polynomials in the following way.

Definition 3.11 *Let* $f, g \in k\{x_1, \ldots, x_n\}$ *and let the leader of* g *be* $x_k^{(\nu)}$. *Then* f *is said to be reduced with respect to* g *if*

(i) f *contains no higher derivative of* x_k *than* $x_k^{(\nu)}$ *and*

(ii) f *is a polynomial of lower degree than* g *in* $x_k^{(\nu)}$.

Notation (non-standard): $f \lhd g$.

Example 3.11 *Let the ranking be given according to (3.2) and consider the same differential polynomials as in the previous example*

$$x\ddot{y} + \dot{z} \lhd x\dot{x} + (\ddot{y})^3 \quad \textit{and} \quad x\dot{x} + (\ddot{y})^3 \lhd x\ddot{y} + \dot{z},$$

$$x\dot{x} + (\ddot{y})^3 \textit{ is not reduced w.r.t. } x^2\ddot{x} + y(\ddot{y})^2 \textit{ but } x^2\ddot{x} + y(\ddot{y})^2 \lhd x\dot{x} + (\ddot{y})^3.$$

Observe that two differential polynomials can be mutually reduced with respect to each other. Given two polynomials not reduced w.r.t. each other, $f \lhd g$ say, we need an algorithm to carry out the reduction. This algorithm will be similar to the division algorithm in the non-differential case with one important difference, we also allow division by the derivatives of g.

3.3.2 Division in $k\{x_1, \ldots, x_n\}$

The concepts from the preceding section makes it possible to develop a division algorithm in $k\{x_1, \ldots, x_n\}$. In the non-differential case the division algorithm for multivariate polynomials lets us write a polynomial $f \in k[x_1, \ldots, x_n]$ on the form

$$f = g_1 f_1 + \ldots + g_s f_s + r,$$

where $f_1, \ldots, f_s, r \in k[x_1, \ldots, x_n]$ and r is not divisible by any of $\mathrm{lt}(f_1), \ldots, \mathrm{lt}(f_s)$. We use the notation

$$f = \langle f_1, \ldots, f_s \rangle + r \qquad \text{or} \qquad f - r \equiv 0 \mod \langle f_1, \ldots, f_s \rangle$$

using the modulo operation.

In the differential case the problem becomes a bit more tricky. A corresponding representation could be

$$f = [f_1, \ldots, f_s] + r \qquad \text{or} \qquad f - r \equiv 0 \mod [f_1, \ldots, f_s], \tag{3.4}$$

where r should be a remainder in some sense. We notice that we also have to allow differentiation in the division process. The following example shows how a division algorithm in $k\{x_1, \ldots, x_n\}$ can be constructed.

Example 3.12 *Let the ranking be $x < \dot{x} < \ddot{x} < \ldots < y < \dot{y} < \ddot{y} < \ldots$ and consider the differential polynomials*

$$f = \dot{x}(y^{(3)})^2 + y\ddot{y}, \qquad g = x\ddot{y}^2.$$

We observe that f is not reduced w.r.t. g and we try to find a representation of f in the form $f = [g] + r$.
Following the non-differential case we should somehow use g and its derivatives to eliminate the leader of f. If we differentiate g we get a polynomial $\dot{g} = 2x\ddot{y}y^{(3)} + \dot{x}\ddot{y}^2$ which has the same leader as f. Now, consider f and g as polynomials in their common leader, i.e., $\mathrm{ld}(f) = \mathrm{ld}(g) = y^{(3)}$:

$$f = a(y^{(3)})^2 + b, \qquad \dot{g} = S_g y^{(3)} + c,$$

where $a = \dot{x}$, $b = y\ddot{y}$, $S_g = 2x\ddot{y}$ and $c = \dot{x}\ddot{y}^2$ are expressions in variables ranked lower than $y^{(3)}$.
We use the division algorithm for polynomials in one variable to determine a remainder on division of f by $\dot{g}$. This gives us

$$f = \left(\frac{a}{S_g}y^{(3)} - \frac{ac}{S_g^2}\right)\dot{g} + \frac{ac^2}{S_g^2} + b \quad \Rightarrow \quad S_g^2 f = q_1\dot{g} + r_1 \tag{3.5}$$

where q_1 and r_1 are polynomials. We observe that according to the division algorithm for polynomials in one variable r_1 will be a polynomial of lower degree *in* $y^{(3)}$ *than* $\dot{g}$*, i.e.,* r_1 *is reduced w.r.t.* $\dot{g}$*. Carrying out the calculations we get* $r_1 = ac^2 + bS_g^2 = \dot{x}^3\ddot{y}^4 + 4x^2y\ddot{y}^3$ *which has the leader* $\ddot{y}$*, i.e., the same leader as* g*. We observe that* r_1 *is not reduced w.r.t.* g*. Applying the division algorithm again to* $r_1 = d\ddot{y}^4 + e\ddot{y}^3$ *and* $g = I_g\ddot{y}^2$ *considered as polynomials in their common leader we get*

$$r_1 = (\frac{d}{I_g}\ddot{y}^2 + \frac{e}{I_g}\ddot{y})g + 0 \quad \Rightarrow \quad I_g r_1 = q_2 g + r_2,\ r_2 = 0, \tag{3.6}$$

where r_2 *is of lower degree in* $\ddot{y}$ *than* g*, i.e.,* r_2 *is reduced w.r.t.* g*.*
Combining (3.5) and (3.6) we get the expression

$$I_g S_g^2 f = \alpha_1 g + \alpha_2 \dot{g} + r, \tag{3.7}$$

where $r = 0$ *in our case. We see that this is almost an expression of the proposed form (3.4).*

The procedure to clear denominators above can be avoided if f is premultiplied by a positive power of the "leading coefficient" of g and $\dot{g}$ considered as polynomials in their leaders. In this case S_g^2 and I_g, respectively. This way of avoiding fractional coefficients is sometimes called pseudo-division, see [64] for an extensive treatment of this algorithm.

If we start with two differential polynomials f and g, where f is not reduced w.r.t. g following the same calculations as in the example we will always get an expression

$$I_g^\sigma S_g^\nu f = \alpha_1 g + \ldots + \alpha_s g^{(s)} + r,$$

where r is reduced w.r.t. g. We notice that $\alpha_1 g + \ldots + \alpha_s g^{(s)} \in [g]$, which motivates the notation

$$I_g^\sigma S_g^\nu f = [g] + r.$$

The expressions I_g and S_g turns out to be of such importance that we give them special names.

Definition 3.12 *The initial* I_g *of a differential polynomial* g *is the coefficient of the highest power of the leader.*

Example 3.13 *For the ranking (3.2)*

$$g_1 = x\ddot{y}^2 \text{ has the leader } \ddot{y},\ \text{hence } I_{g_1} = x$$

$$g_2 = \dot{x}\ddot{y}^3 + \dot{y}\ddot{y}^3 + x\ddot{y}^2 \text{ has the leader } \ddot{y},\ \text{hence } I_{g_2} = \dot{x} + \dot{y}.$$

Definition 3.13 *The separant* S_g *of a differential polynomial* g *is the partial derivative of* g *w.r.t. the leader.*

Example 3.14 *For the ranking (3.2)*

$$g_1 = x\ddot{y}^2 \textit{ has the leader } \ddot{y}, \textit{ hence } S_{g_1} = \frac{\partial g_1}{\partial \ddot{y}} = 2x\ddot{y}$$

$$g_2 = \dot{x}\ddot{y}^3 + \dot{y}\ddot{y}^3 + x\ddot{y}^2 \textit{ has the leader } \ddot{y}, \textit{ hence } S_{g_2} = \frac{\partial g_1}{\partial \ddot{y}} = 3\dot{x}\ddot{y}^2 + 3\dot{y}\ddot{y}^2 + 2x\ddot{y}.$$

Notice that the initial and separant are the "leading" coefficients for g and $g^{(i)}$ considered as polynomials in their leaders.

3.3.3 Characteristic Sets

The division algorithm in $k\{x_1, \ldots, x_n\}$ can be used to construct an algorithm for calculating a useful description, called a *characteristic set*, of a possibly infinite set of differential polynomials. The details of the complete algorithm for constructing a characteristic set is rather involved and we will not reproduce it here. The interested reader may consult [50, 72] or [36] for a full algorithmic treatment. To define characteristic sets we need some auxiliary concepts.

Definition 3.14 *A set of differential polynomials, all of which are reduced with respect to each other, is called an autoreduced set.*

Example 3.15 *Let the ranking be given according to (3.2). The polynomials*

$$x\dot{x} + \ddot{x}^2, \quad \ddot{x} + x\ddot{y}, \quad y^2\dot{y} + z$$

form an autoreduced set.

Next example shows an interesting fact about state space equations of dynamical systems.

Example 3.16 *Let $f_1, \ldots, f_n, h \in k[x_1, \ldots, x_n, u]$ and consider*

$$\begin{aligned} \dot{x}_1 - f_1(x_1, \ldots, x_n, u) &= 0 \\ &\vdots \\ \dot{x}_n - f_n(x_1, \ldots, x_n, u) &= 0 \\ y - h(x_1, \ldots, x_n, u) &= 0 \end{aligned}$$

which is the state space equations for some system. These polynomials forms an autoreduced set under the ranking

$$u < \dot{u} < \ldots < x_1 < \ldots < x_n < y < \dot{x}_1 < \ldots < \dot{x}_n < \dot{y} < \ldots$$

Notice that the equations have the leaders $\dot{x}_1, \ldots, \dot{x}_n, y$, respectively.

Theorem 3.4 *In an autoreduced set the polynomials must have different leading variables.*

Proof *Suppose that two differential polynomials in an autoreduced set have leaders which are derivatives of the same variable. Then we have two cases:*

(i) One of the leaders is a higher derivative of the leading variable than the other. Then the polynomial whose leader is of higher order is not reduced w.r.t. the other one.

(ii) The leaders are of the same order. Then one of them is of equal or higher power in the leader than the other. Hence, this polynomial is not reduced w.r.t. the other one.

Both cases contradicts the fact that the polynomials belong to an autoreduced set, since they exclude mutual reducedness. □

There is a way to generalize the division algorithm in $k\{x_1, \dots, x_n\}$ to the case when we want to calculate the remainder of a differential polynomial w.r.t. an autoreduced set of differential polynomials, (cf. Gröbner bases). Given an autoreduced set $\mathbf{A}$ and a differential polynomial f then one can show that f can be written:

$$\prod_{A_j \in \mathbf{A}} S_{A_j}^{\sigma_j} I_{A_j}^{\nu_j} f = [\,\mathbf{A}\,] + r, \tag{3.8}$$

where r is called the remainder of f with respect to the autoreduced set $\mathbf{A}$. The remainder r is reduced with respect to the elements of $\mathbf{A}$ and is lower ranked than f.

We now introduce a common notation, which allow us to write (3.8) in a more convenient way. Let $H_{\mathbf{A}} = \prod_{A_j \in \mathbf{A}} S_{A_j} I_{A_j}$, where S_{A_j} and I_{A_j} are the separant and initial of A_j, respectively.

Definition 3.15 *Let* $[\,\mathbf{A}\,] : H_{\mathbf{A}}^{\infty} = \left\{ f \mid H_{\mathbf{A}}^{s} f \in [\,\mathbf{A}\,] \text{ for some } s \in \mathbb{N} \right\}$.

Example 3.17 *Consider the polynomials* f *and* g *in Example 3.12 and let* $\mathbf{A} = \{g\}$. *Since* $\mathbf{A}$ *only consists of one element it is autoreduced. In Example 3.12 we deduced the expression*

$$I_g S_g^2 f = \alpha_1 g + \alpha_2 \dot{g} \in [\,\mathbf{A}\,] \quad \Rightarrow \quad I_g^2 S_g^2 f \in [\,\mathbf{A}\,].$$

Using Definition 3.15 we can now write this as

$$f \in [\,\mathbf{A}\,] : H_{\mathbf{A}}^{\infty} = [\,g\,] : H_g^{\infty}.$$

Continuing the generalization of ranking, beginning with variables and polynomials we now define a ranking of autoreduced sets.

Definition 3.16 *Let* $\mathbf{A} = \{A_1, \ldots, A_r\}$ *and* $\mathbf{B} = \{B_1, \ldots, B_s\}$ *be two autoreduced sets of differential polynomials, where the polynomials are ordered in increasing rank. We say that* $\mathbf{A}$ *is higher ranked than* $\mathbf{B}$ *if*

(i) there is a $j \le \min(r, s)$ *such that for all* $i < j$ A_i *and* B_i *are equally ranked and* A_j *is higher ranked than* B_j *or*

(ii) if $s > r$ *and for all* $i \le r$ A_i *and* B_i *are equally ranked.*

Example 3.18 *Let the ranking be given by (3.2) and consider the autoreduced sets*

$$\mathbf{A} = \{ x\ddot{x}^2 + \dot{x}, x\dot{y} + y, \dot{z} - x \},$$
$$\mathbf{B} = \{ \ddot{x}^2 + 17, y\dot{y} - \dot{x}, z - \dot{x} \}.$$

$\mathbf{A}$ *is ranked higher than* $\mathbf{B}$ *since the two first polynomials in each set are equally ranked but the last polynomial in* $\mathbf{A}$ *is higher ranked than the last one in* $\mathbf{B}$*. Let*

$$\mathbf{C} = \{ \ddot{x}^2 + x, \dot{y} - \ddot{x}y \},$$

which is an autoreduced set. Then, $\mathbf{C}$ *is higher ranked than both* $\mathbf{A}$ *and* $\mathbf{B}$ *since the polynomials in* $\mathbf{C}$ *are equally ranked as the two first polynomials in* $\mathbf{A}$ *and* $\mathbf{B}$ *but* $\mathbf{C}$ *has fewer elements.*

As for variables and differential polynomials there is a theorem concerning finiteness of descending sequences.

Theorem 3.5 *A sequence of autoreduced sets, each one ranked strictly lower than the preceding one, can only have a finite length.*

Proof *See [36].* □

Compare the above theorem with the ascending chain condition (2.1) and Definition 2.12. Although differential ideals do not satisfy the ascending chain condition, chains of autoreduced sets have a similar saturating property.

We next define the concept of *characteristic sets*, which is a finite description of a possibly infinite set of differential polynomials, e.g., an ideal.

Definition 3.17 *Let* Σ *be a set of differential polynomials, not necessarily finite. If* $\mathbf{A} \subset \Sigma$ *is an autoreduced set, such that no lower ranked autoreduced set can be formed in* Σ*, then* $\mathbf{A}$ *is called a* characteristic set *of* Σ*.*

Theorem 3.6 *Every set of differential polynomials has a characteristic set.*

Proof *Follows from Theorem 3.5.* □

This theorem tells us that every differential ideal, which is an infinite set of differential polynomials has a characteristic set.

Ritt's algorithm

There is an algorithm due to Ritt,see [36, 72], for finding the characteristic set of a prime differential ideal $[\,\mathbf{\Phi}\,]$ generated by a finite set of differential polynomials $\mathbf{\Phi}$. In short the algorithm consists of extracting an characteristic set $\mathbf{A}$ from a finite set of differential polynomials $\mathbf{\Phi}$. Then one computes the remainders of the polynomials in $\mathbf{\Phi}$ w.r.t. $\mathbf{A}$. The set $\mathbf{\Phi}$ is now extended with the nonzero remainders and one picks a new characteristic set $\mathbf{A}$ from $\mathbf{\Phi}$. The above steps are repeated and the process continues until all elements in $\mathbf{\Phi}$ produces zero remainders on division by $\mathbf{A}$, (cf. the calculation of remainders of S-polynomials in the Gröbner basis algorithm). The algorithm terminates according to Theorem 3.5.

It can be shown that if $\Sigma = [\,\mathbf{\Phi}\,]$ is a differential ideal, not necessarily prime and $\mathbf{A}$ is the set of differential polynomials obtained by running Ritt's algorithm then

$$[\,\mathbf{A}\,] \subset \Sigma \subset [\,\mathbf{A}\,] : \mathsf{H}_{\mathbf{A}}^{\infty}. \tag{3.9}$$

We will examine this expression a bit closer in Chapter 6. An implementation of Ritt's algorithm is described in [35].

4

Real Algebra and Cylindrical Algebraic Decomposition

The formulation of problems in terms of real numbers have an advantage over the often implicit assumption that the variables are complex. We can use not only equations to describe objects but also inequalities to form so called *real polynomial systems*, i.e., systems of polynomial equations and inequalities. The set of solutions to such systems has a much more descriptive power than the solution sets of systems of polynomial equations. Another advantage is the possibility of visualizing objects described in this way. It is easier to imagine or draw pictures of shapes and forms when all coordinates are real numbers. However, the price we have to pay seems to be more complex algorithms.

Real algebra, roughly speaking, is the study of “real objects” such as real rings, real fields and real varieties. The constructive parts of real algebra consists of, e.g., computation of bounds on the real zeros of a polynomial, sign determination of a polynomial evaluated over a real zero of another polynomial, and computation of the number of real distinct zeros of a polynomial in an interval.

The main objective of this chapter is to serve as a an introduction to two constructive methods in real algebra; so called *cylindrical algebraic decomposition* (CAD) and *quantifier elimination* (QE). Given a set of multivariate polynomials the CAD algorithm decomposes $\mathbb{R}^n$ into components over which the polynomials have constant signs. This decomposition can then be utilized to decide if any real polynomial system determined by the given polynomials has a solution. If this is the case the algorithm also provides a point in each connected solution set of the system. Briefly the QE method can be said to be an algorithm for eliminating variables, quantified by $\exists$ (there exists) and $\forall$ (for all), from formulas consisting of polynomial equations and inequalities combined by Boolean operators, e.g., $\wedge$

(and), $\vee$ (or), and $\rightarrow$ (implies). The objective is to find an equivalent formula without quantifiers in the unquantified or free variables only.

The chapter is organized as follows. In Section 4.1 some basic definitions and concepts of real algebra is reviewed. Section 4.2 presents some of the terminology needed in connection with CAD such as semialgebraic sets and decompositions. A detailed presentation of CAD relies on a number of technical tools such as resultants, subresultants, Sturm chains, projection operators, real algebraic numbers etc., which are treated in Section 4.3. In Section 4.4 we present the CAD algorithm and give a number of examples illustrating the different steps of the algorithm. Finally, an introduction to QE and its connections to CAD is given in Section 4.5.

4.1 Real Closed Fields

One of the underlying basic structures in real algebra is so called *real closed fields.* This is a specialization of the field concept such that an ordering of the field elements can be introduced, i.e., we can always compare field elements w.r.t. the ordering and use terms like greater/less than and between to describe how field elements are related. Formally we have the following axiomatic definition.

Definition 4.1 *An ordered field is a field,* k *together with a subset* $P \subset k$*, the set of positive elements, such that*

(i) $0 \notin P$.

(ii) If $a \in k$*, then either* $a \in P$, $a = 0$ *or* $a \in -P$.

(iii) If $a, b \in P$*, then* $a + b \in P$ *and* $ab \in P$*, i.e.,* P *is closed under addition and multiplication.*

Hence, an ordered field is a field that can be decomposed into three disjoint subsets: the positive elements, zero and the negative elements. The negative elements are the additive inverses of the positive elements. Given an ordered field we can introduce an ordering, $>$ by defining

$$a > b \quad \text{if} \quad a - b \in P,$$

which can be shown to satisfy the properties usually associated with the inequality sign and the real numbers such as

$$a > b \;\Rightarrow\; a + c > b + c \quad \text{and} \quad a > b, c > 0 \;\Rightarrow\; a\,c > b\,c.$$

By $a \geq b$ we mean $a > b$ or $a = b$. Using the concept of an ordering we can define closed and open intervals, respectively:

$$[a, b] \triangleq \{\, x \in k \,|\, a \leq x \leq b \,\}, \quad \text{and} \quad (a, b) \triangleq \{\, x \in k \,|\, a < x < b \,\}.$$

The absolute value $|a|$ of a field element a is defined in the obvious way and by construction $|a| \in P$.

Definition 4.2 *An field* k *is called formally real if* -1 *cannot be written as a sum of squares, i.e.,* $\sum_{i=1}^{n} a_i^2 \neq -1$ *for any* a_i.

A motivation for this definition is given by the following observations. In any ordered field k we have $1 > 0 > -1$, since if $-1 > 0$, i.e., $-1 \in P$ then $(-1)^2 \in P$ and the positive elements is closed under addition. Hence, $0 = (-1) + (-1)^2 > 0$, which is impossible. Furthermore,

$$a \neq 0 \;\Rightarrow\; a^2 = (-a)^2 = |a|^2 > 0 > -1.$$

Hence $\sqrt{(-1)} \notin k$ and $a_i \neq 0 \;\Rightarrow\; \sum_{i=1}^{n} a_i^2 > 0 > -1$. The second observation follows from

$$\sum_{i=1}^{n} a_i^2 = \sum_{i=1}^{n} |a_i^2| \geq \left| \sum_{i=1}^{n} a_i^2 \right| > 0$$

and $\sum_{i=1}^{n} a_i^2 = 0$ if and only if $a_i = 0, i = 1, \ldots, n$. Hence, every ordered field is formally real.

Example 4.1 *The rational numbers* $\mathbb{Q}$ *is a formally real field. The complex numbers* $\mathbb{C}$ *is not a formally real field, since* $i^2 = -1$. *The set of rational functions in one variable over* $\mathbb{Q}$, $\mathbb{Q}(t)$ *is a formally real field.*

To get an algebraically more complete structure the following concept can be introduced.

Definition 4.3 *An ordered field* k *is called real closed if*

(i) Every positive element of k *has a square root in* k.

(ii) Every polynomial $f(x) \in k[x]$ *of odd degree has a root in* k.

Example 4.2 *The real numbers* $\mathbb{R}$ *is a real closed field. The rational numbers* $\mathbb{Q}$ *is not a real closed field since* $\sqrt{2} \notin \mathbb{Q}$.
A real algebraic number, α *is a real number that satisfies a polynomial equation* $f(\alpha) = 0$, *where* $f(x) \in \mathbb{Q}[x]$. *The set of real algebraic numbers is a real closed field.*

The real algebraic numbers are important in constructive algorithms in real algebra for two reasons. They obey an exact representation on computers and are the natural ground field in many computations. Much more about algebraic numbers can be found in [17]. Real algebraic numbers are used in the CAD algorithm.

This introduction to some basic concepts in real algebra is only a scratch on the surface of this mathematical branch. Further topics in real algebra concerns quadratic forms, ideals, Artin-Schreier theory etc. The interested reader may consult [7, 21, 48, 51, 64].

4.2 Semialgebraic Sets and Decompositions

The notions of *semialgebraic* sets and *decompositions* play a key role when studying algorithms in real algebra and geometry. Here we will introduce the reader to semialgebraic sets and some other basic concepts needed to understand the CAD method. The definitions essentially come from [4]. A comprehensive treatment of CAD is given in [64][1] and some briefer presentations can be found in [24, 41].

The ground field is assumed to be the real numbers $\mathbb{R}$ in the rest of this section, which allows us to draw pictures.

Definition 4.4 *A subset of* $\mathbb{R}^n$ *is semialgebraic if it can be constructed by finitely many applications of union, intersection and complementation operations on sets of the form*

$$\{\, x \in \mathbb{R}^n \mid f(x) \geq 0 \,\},$$

where $f \in \mathbb{R}[x_1, \ldots, x_n]$.

We use the Boolean operators $\wedge$ (and) and $\vee$ (or) to combine inequalities of the above form to represent semialgebraic sets.

Example 4.3 *Using Boolean operators we have*

$$\{\, x \in \mathbb{R}^n \mid f(x) \geq 0 \,\} \cup \{\, x \in \mathbb{R}^n \mid g(x) \geq 0 \,\} = \{\, x \in \mathbb{R}^n \mid f(x) \geq 0 \vee g(x) \geq 0 \,\}$$

An interesting property of semialgebraic sets is that they are closed under projection, i.e., the projection of a semialgebraic set to some lower dimensional space is again a semialgebraic set. In general, this is not true for varieties or algebraic sets, i.e., sets defined by a system of polynomial equations (consider the unit circle). We also observe that the set of points which satisfies a system of equalities and inequalities is a semialgebraic set.

Example 4.4 *An example of a semialgebraic set* $\mathcal{S} \in \mathbb{R}^2$*,*

$$\mathcal{S} = \{ x \in \mathbb{R}^2 \mid \left[x_1^2 + x_2^2 - 1 \leq 0 \wedge x_1^2 - x_2 = 0 \right] \vee \\ \left[(x_1 - 1)^2 + (x_2 - 1)^2 - 1 \leq 0 \wedge (x_1 - 2)^2 + (x_2 - 2)^2 - 1 \leq 0 \right] \}.$$

The projection of the set onto the x_1*-axis*

$$\mathcal{S}_{x_1} = \{\, x \in \mathbb{R}^2 \mid x_1^2 + x_1^4 - 1 \leq 0 \vee 1 \leq x_1 \leq 2 \,\},$$

is again a semialgebraic set, see Figure 4.1.

We now introduce some terminology for describing how algebraic curves and surfaces partition $\mathbb{R}^n$.

[1] Unfortunately there are some errors in the description of the algorithm, but a correct version can be found in the errata

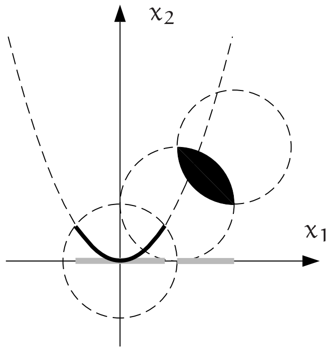


Figure 4.1 A semialgebraic set (black region) and its projection onto the x_1-axis (gray region). The dashed lines correspond to the zero sets of the defining polynomials.

Definition 4.5

(i) *A region,* R *is a connected subset of* $\mathbb{R}^n$.

(ii) *The set* $Z(R) = R \times \mathbb{R} = \{ (\alpha, x) \mid \alpha \in R,\ x \in \mathbb{R} \}$ *is called a cylinder over* R.

(iii) *Let* f, f_1, f_2 *be continuous, real-valued functions on* R.
A f*-section of* $Z(R)$ *is the set* $\{ (\alpha, f(\alpha)) \mid \alpha \in R \}$ *and a* (f_1, f_2)*-sector of* $Z(R)$ *is the set* $\{ (\alpha, \beta) \mid \alpha \in R,\ f_1(\alpha) < \beta < f_2(\alpha) \}$.

(iv) *Let* $X \subset \mathbb{R}^n$. *A decomposition of* X *is a finite collection of disjoint regions (or components) whose union is* X, *i.e.,* $X = \bigcup_{i=1}^{k} X_i,\ \ X_i \cap X_j = \emptyset,\ \ i \neq j$.

(v) *A stack over* R *is a decomposition which consists of* f_i*-sections and* (f_i, f_{i+1})*-sectors, where* $f_0(x) < \ldots < f_{k+1}(x)$ *for all* $x \in R$ *and* $f_0 = -\infty,\ f_{k+1} = +\infty$.

Notice that an algebraic equation in n variables implicitly defines a set of real-valued, piecewise continuous functions over $\mathbb{R}^{n-1}$.

Observe the strict inequalities in Definition 4.5. Geometrically this means that the graphs of different functions do not intersect each other over R. The above concepts are visualized in Figure 4.2.

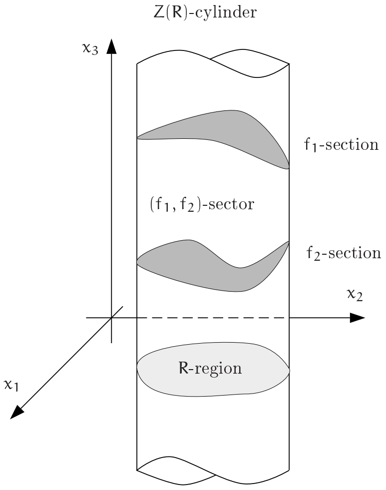


Figure 4.2 *A geometrical interpretation of the definitions of region, cylinder, sections, sectors and stack. The stack consists of three sectors and two sections in this particular case.*

Definition 4.6 *A decomposition $\mathcal{D}$ of $\mathbb{R}^n$ is cylindrical if*

$n = 1$ *$\mathcal{D}$ is a partition of $\mathbb{R}^1$ into a finite set of numbers, and the finite and infinite open intervals bounded by these numbers.*

$n > 1$ *$\mathcal{D}' = F_1 \cup \ldots \cup F_m$ is a cylindrical decomposition of $\mathbb{R}^{n-1}$ and over each F_i there is a stack which is a subset of $\mathcal{D}$.*

From the definition of a cylindrical decomposition it is clear that any cylindrical decomposition of $\mathbb{R}^n$ induces a cylindrical decomposition of $\mathbb{R}^{n-1}$ etc. down to $\mathbb{R}^1$.

The property that a polynomial has constant sign over a region appears to be important.

Definition 4.7 *Let $X \subset \mathbb{R}^n$ and $f \in \mathbb{R}[x_1, \ldots, x_n]$. Then f is invariant on X if one of $f(x) > 0$, $f(x) = 0$ or $f(x) < 0$ holds for all $x \in X$. The set $\mathcal{F} = \{f_1, \ldots, f_r\} \in \mathbb{R}[x_1, \ldots, x_n]$ of polynomials is invariant on X if each f_i is invariant on X.*

We also say that X is $\mathcal{F}$-invariant if $\mathcal{F}$ is invariant on X.

Example 4.5 *Let $\mathcal{F} = \{x_1^2 + x_2^2 - 1,\ (x_1 - 1)^2 + x_2^2 - 1\}$. Then the set $\mathcal{I}$ is $\mathcal{F}$-invariant but $\mathcal{J}$ is not, see Figure 4.3.*

Definition 4.8 *A decomposition is algebraic if each of its components is a semi-algebraic set.*

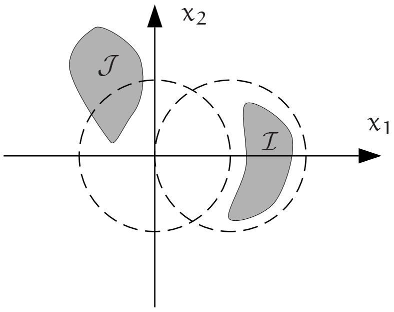


Figure 4.3 *The zero set of $\mathcal{F}$ in Example 4.5 (dashed lines), the set $\mathcal{J}$ and the $\mathcal{F}$-invariant set $\mathcal{I}$.*

Example 4.6 *Let* $f(x) = (x_1 - 2)^2 + (x_2 - 2)^2 - 1$. *Then*

$$\{x \mid f(x) > 0\} \cup \{x \mid f(x) = 0\} \cup \{x \mid f(x) < 0\}$$

is an algebraic decomposition of $\mathbb{R}^2$.

Notice that when the decomposition is defined by a set of polynomials it is algebraic since all boundaries are zero sets of the defining polynomials.

Definition 4.9 *A Cylindrical Algebraic Decomposition (CAD) of* $\mathbb{R}^n$ *is a decomposition which is both cylindrical and algebraic. The components of a CAD are called cells.*

Example 4.7 *Let* $\mathcal{F} = \{(x_1 - 2)^2 + (x_2 - 2)^2 - 1,\ (x_1 - 3)^2 + (x_2 - 2)^2 - 1\}$. *The CAD consists of the distinct black "dots", "arcs" and "patches of white space" in Figure 4.4 whose union is* $\mathbb{R}^2$. *The induced CAD of* $\mathbb{R}^1$ *consists of the gray dots and the intermediate intervals on the* x_1*-axis.*
One of the cells may be semialgebraically characterized as

$$\begin{aligned}\{x \in \mathbb{R}^2 \mid\ & (x_1 - 2)^2 + (x_2 - 2)^2 - 1 > 0\ \wedge \\ & (x_1 - 3)^2 + (x_2 - 2)^2 - 1 < 0\ \wedge \\ & x_1 < 3\ \wedge\ x_2 < 2\}.\end{aligned}$$

Which one? ;-)

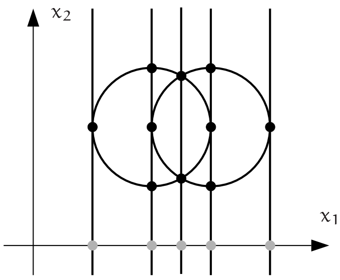


Figure 4.4 *A CAD of $\mathbb{R}^2$ and the induced CAD of $\mathbb{R}^1$.*

4.3 Tools

In this section we present some constructive tools which are used in different steps of the CAD-algorithm. These are polynomial remainder sequences, subresultants, principal subresultant coefficients, Sturm chains, projection operators, and representations of algebraic numbers. These are the main tools for other constructive methods in real algebra and real algebraic geometry, see [7, 9, 48, 64]. Most of the concepts in this chapter are also treated in [64]. Readers interested in the CAD algorithm but not the underlying details may at this stage proceed to Section 4.4.

4.3.1 Polynomial Remainder Sequences

An important step in the CAD algorithm is to find projections of "significant points" of higher dimensional real zero sets such as vertical tangents and self-crossings. These projections are strongly related to how the multiplicity of the zeros of a polynomial depends on its coefficients. This dependency can be captured by studying common factors of multivariate polynomials.

We start with the univariate case and then generalize the ideas to multivariate polynomials. Given two polynomials f, $g \in k[x]$ we can ask how many common zeros f and g have and how many distinct zeros f has? These two questions can be answered knowing the *greatest common divisor* (gcd) of two polynomials.

Lemma 4.1 *Let f, $g \in k[x]$. Then the number of common zeros of f and g is equal to the degree of $\gcd(f, g)$. Furthermore, the number of distinct zeros of f is equal to the difference between the degree of f and the degree of $\gcd(f, f')$.*

Proof *The first assertion follows from the definition of* gcd *and the fact that the number of zeros (counting multiplicity) of a univariate polynomial is equal to its degree.*

The second assertion follows from the observation that the multiplicity of a zero drops by one when differentiating a polynomial. Hence, all multiple zeros can be removed by dividing f *by* $\gcd(f, f')$ *and the lemma follows.* □

How do we compute the gcd of two univariate polynomials? This is a classical question and the answer is provided by the Euclidean algorithm for polynomials. By repeated polynomial division we end up with the gcd of the two original polynomials. The process may be described as follows:

$$\begin{aligned}
f_1 &= q_1 f_2 + f_3, &\qquad& \deg(f_3) < \deg(f_2)\\
f_2 &= q_2 f_3 + f_4, && \deg(f_4) < \deg(f_3)\\
&\vdots\\
f_{p-2} &= q_{p-2} f_{p-1} + f_p, && \deg(f_p) < \deg(f_{p-1})\\
f_{p-1} &= q_{p-1} f_p + 0,
\end{aligned}$$

where $\deg(f_2) \leq \deg(f_1)$ and $f_p = \gcd(f_1, f_2)$, see [49] or [23] for a detailed treatment. The algorithm terminates since the degree decreases in each step. The sequence $(f_1, f_2, \ldots, f_p)$ is called a *polynomial remainder sequence* (PRS).

Example 4.8 *Let* $f_1 = (1+x)(3+x)(1+x+x^2)$ *and* $f_2 = (1+x)(2+x)^2$*. The PRS of* f_1, f_2 *in expanded form is*

$$\begin{aligned}
f_1(x) &= 9 + 24x + 31x^2 + 23x^3 + 8x^4 + x^5\\
f_2(x) &= 4 + 8x + 5x^2 + x^3\\
f_3(x) &= 9 + 12x + 3x^2\\
f_4(x) &= 1 + x,
\end{aligned}$$

where $f_4(x) = 1 + x = \gcd(f_1, f_2)$.

The above ideas can be extended to multivariate polynomials. Let $f, g \in k[x_1, \ldots, x_{n-1}][x_n]$, i.e., f and g is considered as polynomials in x_n with polynomial coefficients in the $x_1, \ldots, x_{n-1}$. Hence the coefficients of f and g is no longer fixed numbers but depends on over which point $\alpha = (\alpha_1, \ldots, \alpha_{n-1}) \in \mathbb{R}^{n-1}$ they are evaluated. This implies that both the number of zeros and their location varies for different $\alpha \in \mathbb{R}^{n-1}$.

Example 4.9 *Let* $f = (x_1^2 + x_2^2 - 1)x_3^2 + (x_2 - 1)x_3 + x_2^2$ *and consider* f *as a polynomial in* x_3*. Then the degree of* f *is 2 everywhere except on the cylinder* $x_1^2 + x_2^2 - 1 = 0$*. On the cylinder the degree is 1 except on the line* $(x_1, x_2) = (0, 1)$ *where it is zero.*

For some $\alpha \in \mathbb{R}^{n-1}$ there might be common factors of f and g but not for others, i.e., the gcd(f, g) and hence its degree depends on α. To compute a gcd of two polynomials we use the Euclidean algorithm with so called *pseudo-division* in each step since the polynomial coefficients no longer belongs to a field. Here is a brief exposition of pseudo-division.

Lemma 4.2 *Let $f,\ g \in k[x_1, \ldots, x_{n-1}][x_n]$, be written in the form*

$$f = f_p x_n^p + \ldots f_0 \quad \textit{and} \quad g = g_m x_n^m + \ldots g_0,$$

where $m \leq p$, i.e., f_i, g_j are polynomials in $k[x_1, \ldots, x_{n-1}]$. Then

$$g_m^s f = qg + r,$$

where $q, r \in k\,[x_1, \ldots, x_{n-1}][x_n]$ are unique, $s = p - m + 1$ and the degree of r w.r.t. x_n is less than m.

Proof *A proof can be found in [64].* □

The remainder r of the pseudo-division is called the pseudo-remainder of f and g and is denoted prem(f, g).

If we allow denominators pseudo-division can be seen as ordinary polynomial division for polynomials in x_n with coefficients from the rational function field $k(x_1, \ldots, x_{n-1})$, followed by clearing denominators. The only term which needs to be inverted in the division is the leading coefficient, g_m of g. Hence we get the equation $g_m^s f = qg + r$. Compare with the division algorithm in $k\{x_1, \ldots, x_n\}$ treated in Section 3.3.

In general, a factorization of a non-invertible ring element into irreducible elements does not need to be unique[2]. A ring whose non-invertible elements can be uniquely factorized is henceforth referred to as a *unique factorization domain* (UFD) [64]. The polynomial rings we have seen so far is UFDs.

Definition 4.10 *Let R be a UFD. Two polynomials in $R[x]$ are similar if there exist $a, b \in R$ such that $a\,f(x) = b\,g(x)$. Notation: $f(x) \sim g(x)$.*

Using the notation of similarity we can now generalize the concept of *polynomial remainder sequences.*

Definition 4.11 *Let $R = k[x_1, \ldots, x_{n-1}]$ and $f_1, f_2 \in R[x_n]$ with $\deg(f_1) \geq \deg(f_2)$, where deg denotes the degree w.r.t. x_n. The sequence $f_1, f_2, \ldots, f_k$ is a PRS for f_1 and f_2 if:*

(i) For all $i = 3, \ldots, k$ $f_i \sim \mathrm{prem}(f_{i-2}, f_{i-1})$.

(ii) The sequence terminates with $\mathrm{prem}(f_{k-1}, f_k) = 0$.

[2] The element 9 in $\mathbb{Z}[\sqrt{-5}]$ can be factorized according to $(2 + i\sqrt{5})(2 - i\sqrt{5}) = 9 = 3 \cdot 3$.

Since we only claim similarity there are infinitely many PRSs to every pair $f_1, f_2 \in k[x_1, \ldots, x_{n-1}][x_n]$ of polynomials. Two such sequences are:

(i) *Euclidean* Polynomial Remainder Sequence (EPRS):

$$\begin{aligned} f_i &= \mathrm{prem}(f_{i-2}, f_{i-1}) \neq 0, \\ \mathrm{prem}(f_{k-1}, f_k) &= 0. \end{aligned}$$

(ii) *Primitive* Polynomial Remainder Sequence (PPRS):

$$\begin{aligned} f_i &= \mathrm{pp}(\mathrm{prem}(f_{i-2}, f_{i-1})) \neq 0, \\ \mathrm{prem}(f_{k-1}, f_k) &= 0. \end{aligned}$$

where pp stands for *primitive part*, i.e. the remaining part of the polynomial when we have removed all common factors of the coefficients.

We observe that a PRS is unique up to similarity since pseudo-division is unique. Furthermore,

$$\gcd(f_1, f_2) \sim \ldots \sim \gcd(f_{k-1}, f_k) \sim f_k,$$

i.e., PRS essentially computes the gcd of two polynomials up to similarity.

From a computational point of view both EPRS and PPRS suffer from complexity problems [49]. EPRS has an exponential coefficient growth and the PPRS algorithm involves calculations of gcd of the coefficients, which in our case again are polynomials. However, this computational complexity is not inherent in the problem. The *Subresultant Polynomial Remainder Sequence* (SPRS) offers a trade-off between coefficient growth and the cost of gcd calculations of the coefficients. We will now take a closer look at subresultants and SPRS.

4.3.2 Subresultants

To extend the Euclidean algorithm for computing the gcd of multivariate polynomials the SPRS seems to be the most attractive alternative. The basic idea of the construction of this PRS is that the coefficients of all polynomials of a PRS can be computed as minors of a special matrix. In this subsection we present one way of computing the SPRS, see [49, 56, 64].

The process of pseudo-division can be organized as row operations on a matrix containing the coefficients of the involved polynomials.

Example 4.10 *Pseudo-division applied to* $f = x^3 + 2x^2 + 3x + 1$ *and* $g = 2x^2 + x + 2$ *gives*

$$2^2 f = (2x+3)g + \boxed{5x - 2}.$$

One way of organizing the calculations is

$$M = \begin{bmatrix} 1 & 2 & 3 & 1 \\ 2 & 1 & 2 & 0 \\ 0 & 2 & 1 & 2 \end{bmatrix} \sim \begin{bmatrix} 2 & 1 & 2 & 0 \\ 0 & 2 & 1 & 2 \\ 0 & 0 & \boxed{5 \quad -2} \end{bmatrix} = \tilde{M}$$

where $\tilde{M}$ is obtained from M by row operations. Observe the correspondence between the pseudo-remainder coefficients and the last row of $\tilde{M}$. In fact, multiplying the first row of M with 2^2, $\tilde{M}$ can be obtained by just subtracting integer multiples of the other rows from the first row. Finally, permuting the rows gives the triangular form. According to the triangular structure of $\tilde{M}$ the pseudo-remainder coefficients may be found, modulo a common factor, by determinant calculations on a matrix consisting of the first two columns of $\tilde{M}$ augmented by either column three or four.

Since M and $\tilde{M}$ only differs by row operations their minors are strongly related. Hence the coefficients of a polynomial similar to the pseudo-remainder may be calculated as minors of the original matrix M. In fact, by computing minors of matrices whose elements are the coefficients of the polynomials f and g we can construct a whole PRS of f and g. To see this we first need a couple of definitions [56, 64]. Throughout the rest of this subsection we let R denote a UFD.

Definition 4.12 *Let $f_i = \sum_{j=0}^{n_i} f_{ij} x^j \in R[x]$, $i = 1, \ldots, k$. The matrix associated with $f_1, \ldots, f_k$ is*

$$\mathrm{mat}(f_1, \ldots, f_k) = \left[\begin{array}{c} f_{i,l-j} \end{array}\right],$$

where $l = 1 + \max_{1 \le i \le k}(n_i)$.

In other words the matrix associated with $f_1, \ldots, f_k$ is a matrix where each row contains the coefficients of one of the polynomials and each column corresponds to decreasing powers of x from left to right.

Definition 4.13 *Let $M \in R^{k \times l}, k \le l$. The determinant polynomial of M is*

$$\mathrm{detpol}(M) = \det(M^{(k)}) x^{l-k} + \ldots + \det(M^{(l)}),$$

where $M^{(j)} = \left[\begin{array}{cccc} M_{\cdot,1} & \ldots & M_{\cdot,k-1} & M_{\cdot,j} \end{array}\right]$.

Example 4.11 *In Example 4.10 we have $M = \mathrm{mat}(f, x\,g, g)$, $\mathrm{detpol}(M) = 5x - 2$ and $\mathrm{detpol}(\tilde{M}) = 2^2(5x - 2)$.*

Definition 4.14 *Let $f, g \in R[x]$ and $\deg(f) = m$, $\deg(g) = n$, $m \ge n$.*

(i) The k^{th} subresultant of f and g is

$$\mathrm{subres}_k(f, g) = \mathrm{detpol}(M_k),$$

where $M_k = \mathrm{mat}(x^{n-k-1} f, \ldots, f, x^{m-k-1} g, \ldots, g)$.

(ii) The subresultant chain of f *and* g *is* $\{S_j\}_{j=0}^{n+1}$, *where*

$$\begin{aligned} S_{n+1} &= f, \\ S_n &= g, \\ S_{n-1} &= \text{subres}_{n-1}(f, g), \\ &\vdots \\ S_0 &= \text{subres}_0(f, g). \end{aligned}$$

Observe that M_0 is the so called Sylvester matrix of f and g and $\text{subres}_0(f, g) = \text{detpol}(M_0) = \det(M_0)$ is the resultant of f and g [23]. Furthermore, the M_k matrices are obtained by deleting rows and columns of M_0.

The importance of the definition of subresultants and subresultant chains lies in the fact that all PRS of f and g is embedded in the subresultant chain of f and g up to similarity. We illustrate the definitions with a numerical example.

Example 4.12 *Let* $f = x^5 - 2x^4 + 3x^3 - 4x^2 + 5x - 6$ *and* $g = 3x^3 + 5x^2 + 7x + 9$. *Then*

$$M_0 = \begin{bmatrix} 1 & -2 & 3 & -4 & 5 & -6 & 0 & 0 \\ 0 & 1 & -2 & 3 & -4 & 5 & -6 & 0 \\ 0 & 0 & 1 & -2 & 3 & -4 & 5 & -6 \\ 3 & 5 & 7 & 9 & 0 & 0 & 0 & 0 \\ 0 & 3 & 5 & 7 & 9 & 0 & 0 & 0 \\ 0 & 0 & 3 & 5 & 7 & 9 & 0 & 0 \\ 0 & 0 & 0 & 3 & 5 & 7 & 9 & 0 \\ 0 & 0 & 0 & 0 & 3 & 5 & 7 & 9 \end{bmatrix}$$

$$M_1 = \left[\begin{array}{ccccc|cc} 1 & -2 & 3 & -4 & 5 & -6 & 0 \\ 0 & 1 & -2 & 3 & -4 & 5 & -6 \\ 3 & 5 & 7 & 9 & 0 & 0 & 0 \\ 0 & 3 & 5 & 7 & 9 & 0 & 0 \\ 0 & 0 & 3 & 5 & 7 & 9 & 0 \\ 0 & 0 & 0 & 3 & 5 & 7 & 9 \end{array}\right], \quad M_2 = \left[\begin{array}{ccc|ccc} 1 & -2 & 3 & -4 & 5 & -6 \\ 3 & 5 & 7 & 9 & 0 & 0 \\ 0 & 3 & 5 & 7 & 9 & 0 \\ 0 & 0 & 3 & 5 & 7 & 9 \end{array}\right].$$

Observe how M_1 *and* M_2 *are obtained by deleting rows and columns of* M_0. *The* $M_k^{(l)}$ *matrices are formed by the left part of the partitioned matrices and one column from the right part. Now, the subresultants are the determinant polynomials of* M_0, M_1 *and* M_2 , *i.e.,*

$$\begin{aligned} S_4 &= x^5 - 2x^4 + 3x^3 - 4x^2 + 5x - 6 \\ S_3 &= 3x^3 + 5x^2 + 7x + 9 \\ S_2 &= \det(M_2^{(4)})x^2 + \det(M_2^{(5)})x + \det(M_2^{(6)}) = 263x^2 - 5x + 711 \\ S_1 &= \det(M_1^{(6)})x + \det(M_1^{(7)}) = -2598x - 11967 \\ S_0 &= \det(M_0) = 616149 = 3^2 \cdot 223 \cdot 307 \end{aligned}$$

Compare the subresultant chain with the EPRS and PPRS for f *and* g*:*

$$
\begin{aligned}
f_1 &= x^5 - 2x^4 + 3x^3 - 4x^2 + 5x - 6 & \tilde{f}_1 &= x^5 - 2x^4 + 3x^3 - 4x^2 + 5x - 6\\
f_2 &= 3x^3 + 5x^2 + 7x + 9 & \tilde{f}_2 &= 3x^3 + 5x^2 + 7x + 9\\
f_3 &= -263x^2 + 5x - 711 & \tilde{f}_3 &= -263x^2 + 5x - 711\\
f_4 &= -70146x - 323109 & \tilde{f}_4 &= -866x - 3989\\
f_5 &= -3^8 \cdot 223 \cdot 263^2 \cdot 307 & \tilde{f}_5 &= -1
\end{aligned}
$$

where $f_4 = 81\,\tilde{f}_4 = 3\,S_1$. *Notice the quick growth of the coefficients in the EPRS compared with the modest growth in the subresultant chain (the difference is more obvious for longer chains). In this example there is a one-one correspondence between the PRS and the subresultant chain. This is due to the fact that the degree drops by one for each pseudo-division. If the degree drops with more than one, some of the subresultants becomes similar.*

What is the connection between a polynomial in a PRS and the determinant polynomial of M_k? By row operations on M_k a number of pseudo-divisions may be carried out consecutively. The process is described by the following example which is a generalization of Example 4.10.

Example 4.13 *Let* $f = a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ *and* $g = b_3x^3 + b_2x^2 + b_1x + b_0$. *The corresponding* M_1 *matrix becomes*

$$
M_1 = \begin{bmatrix}
a_5 & a_4 & a_3 & a_2 & a_1 & a_0 & \\
 & a_5 & a_4 & a_3 & a_2 & a_1 & a_0\\
b_3 & b_2 & b_1 & b_0 & & & \\
 & b_3 & b_2 & b_1 & b_0 & & \\
 & & b_3 & b_2 & b_1 & b_0 & \\
 & & & b_3 & b_2 & b_1 & b_0
\end{bmatrix}
\overset{(1)}{\sim}
\begin{bmatrix}
b_3 & b_2 & b_1 & b_0 & & & \\
 & b_3 & b_2 & b_1 & b_0 & & \\
 & & b_3 & b_2 & b_1 & b_0 & \\
 & & & b_3 & b_2 & b_1 & b_0\\
 & & & \tilde{a}_2 & \tilde{a}_1 & \tilde{a}_0 & \\
 & & & & \tilde{a}_2 & \tilde{a}_1 & \tilde{a}_0
\end{bmatrix}
\overset{(2)}{\sim}
$$

$$
\begin{bmatrix}
b_3 & b_2 & b_1 & b_0 & & & \\
 & b_3 & b_2 & b_1 & b_0 & & \\
 & & b_3 & b_2 & b_1 & b_0 & \\
 & & & \tilde{a}_2 & \tilde{a}_1 & \tilde{a}_0 & \\
 & & & & \tilde{a}_2 & \tilde{a}_1 & \tilde{a}_0\\
 & & & & & \tilde{b}_1 & \tilde{b}_0
\end{bmatrix}
= \tilde{M}_1
$$

Row operations in detail:

(1) *Multiply the first row with* b_3^3 *and subtract multiplicities of row* $3, 4$ *and* 5 *to eliminate* a_5, a_4 *and* a_3. *A similar treatment of the second row followed by interchanging rows gives the second matrix. The resulting matrix elements* $\tilde{a}_i$ *are the coefficients of* $\operatorname{prem}(f, g)$.

(2) *The same process as in (1) repeated on row* $4, 5$ *and* 6. *The matrix elements* $\tilde{b}_i$ *are the coefficients of* $\operatorname{prem}(g, \operatorname{prem}(f, g))$.

As pointed out earlier the only operations needed to get from M_k to the last triangular matrix, $\tilde{M}_k$ are row operations. It is then clear that the minors $\det(M_k^{(l)})$ and the corresponding minors of the triangular matrix only differ by a common factor.

Our original motivation for calculating PRSs was to determine the gcd of two polynomials and especially its degree, since the degree was strongly related to the number of common zeros. Hence, we are interested in the coefficient of the highest power of a gcd.

Definition 4.15 *Let* $f,\ g \in R[x]$ *and* $\deg(f) = m$, $\deg(g) = n$, $m \geq n$. *The* k^{th} *principal subresultant coefficient of* f *and* g *is*

$$\mathrm{psc}_k(f, g) = \det(M_k^{(k)}), \quad 0 \leq k \leq n.$$

In other words $\mathrm{psc}_k(f, g)$ is the coefficient of x^k in $\mathrm{subres}_k(f, g)$. The following lemma, which is not hard to believe in knowing the connection between a PRS and the corresponding subresultant chain, tells us that knowing the psc chain of two polynomials we know the degree of their gcd.

Lemma 4.3 *Let* $f,\ g \in R[x]$ *where* $\deg(f) = m$, $\deg(g) = n$ *and* $0 < i \leq \min(m, n)$. *Then* f *and* g *have a common factor of degree* i *if and only if*

$$\mathrm{psc}_j(f, g) = 0, j = 0, \ldots, i-1 \quad \textit{and} \quad \mathrm{psc}_i(f, g) \neq 0.$$

Proof *See Corollary 7.7.9. in [64].* □

In this subsection we have worked with polynomials in $R[x]$ where R is an UFD. For simplicity the examples were done for polynomials over the integers but any UFD will do, e.g., multivariate polynomials. In the following we use $R = \mathbb{R}[x_1, \ldots, x_{n-1}]$, i.e., polynomials in $\mathbb{R}[x_1, \ldots, x_{n-1}][x_n]$.

4.3.3 The Projection Operator

The key idea in the so called projection phase of the CAD algorithm is to find regions over which a given set of polynomials have a constant number of real roots. This is formalized by the concept of delineability.

Let $f \in \mathbb{R}[x_1, \ldots, x_{n-1}][x_n]$ be a real polynomial in n-variables:

$$f(x_1, \ldots, x_{n-1}, x_n) = f^d(x_1, \ldots, x_{n-1})x_n^d + \cdots + f^0(x_1, \ldots, x_{n-1})$$

and $\xi = (\xi_1, \ldots, \xi_{n-1}) \in \mathbb{R}^{n-1}$. Then we write $f_\xi(x_n) = f(\xi_1, \ldots, \xi_{n-1}, x_n)$ for the univariate polynomial obtained by substituting ξ for the first $n-1$ variables.

We also introduce the *reductum*, $\hat{f}^k$ of a polynomial. Let

$$\hat{f}^k(x_1, \ldots, x_{n-1}, x_n) = f^k(x_1, \ldots, x_{n-1})x_n^k + \cdots + f^0(x_1, \ldots, x_{n-1}),$$

where $0 \leq k \leq d$. Thus, if $f^d(\xi) = \cdots = f^{k+1}(\xi) = 0$ and $f^k(\xi) \neq 0$, then

$$f_\xi(x_n) = \hat{f}^k(\xi, x_n) = f^k(\xi)x_n^k + \cdots + f^0(\xi).$$

Definition 4.16 *Let $\mathcal{F} = \{f_1, f_2, \dots, f_r\} \subset \mathbb{R}[x_1, \dots, x_{n-1}][x_n]$ be a set of multivariate real polynomials and $C \subset \mathbb{R}^{n-1}$ a region. We say that $\mathcal{F}$ is delineable on C, if it satisfies the following invariant properties:*

(i) For every $1 \le i \le r$, the total number of complex roots of $f_{i,\xi}(x_n)$ (counting multiplicity) remains invariant as ξ varies over C.

(ii) For every $1 \le i \le r$, the number of distinct complex roots of $f_{i,\xi}(x_n)$ remains invariant as ξ varies over C.

(iii) For every $1 \le i < j \le r$, the total number of common complex roots of $f_{i,\xi}(x_n)$ and $f_{j,\xi}(x_n)$ (counting multiplicity) remains invariant as ξ varies over C.

Observe that the set of polynomials defining the CAD in Figure 4.4 is delineable over each cell of the decomposition of $\mathbb{R}^1$.

Consider one of the polynomials $f_i \in \mathcal{F}$. Since it has real coefficients its complex roots have to occur in conjugate pairs. Let ξ and ξ' be two points in $C \subset \mathbb{R}^{n-1}$. The only way for a pair of complex conjugated roots of $f_{i,\xi}$ to become real when ξ varies to ξ' is to coalesce into a real double root, i.e., the number of distinct roots of f_i drops. Hence a transition from a non-real root to a real root is impossible over C. Similar arguments holds for the reversed problem.

Lemma 4.4 *Let $\mathcal{F} \subset \mathbb{R}[x_1, \dots, x_{n-1}][x_n]$ be a set of polynomials and let $\mathcal{F}$ be delineable over $C \subset \mathbb{R}^{n-1}$. Then the total number of distinct real roots of $\mathcal{F}$ is invariant over C.*

Proof *See Lemma 8.6.3. in [64].* □

We are now able to formulate the main theorem of this subsection.

Theorem 4.1 *Let $\mathcal{F} \subset \mathbb{R}[x_1, \dots, x_{n-1}][x_n]$ be a set of polynomials and let $C \subset \mathbb{R}^{n-1}$ be a connected maximal $\mathcal{F}$-delineable set. Then C is semialgebraic.*

Proof *A complete proof may be found in [18] or in the errata of [64]. Here we only sketch the ideas. The idea is to show that the three invariant properties of Definition 4.16 have semialgebraic characterizations. Let $\mathrm{psc}_i^{x_n}$ denote the i^{th} principal resultant coefficient w.r.t. x_n and D_{x_n} denote the formal derivative operator w.r.t. x_n. Let $\mathcal{F} = \{f_1, \dots, f_r\}$.*

(i) Total number of complex roots of $f_{i,\xi}$ remains invariant over C. This corresponds to the claim that

$$(\forall 1 \le i \le r)(\exists 0 \le k_i \le d_i)$$
$$\Big[(\forall k > k_i) \left[f_i^k(x_1, \dots, x_{n-1}) = 0 \right] \wedge f_i^{k_i}(x_1, \dots, x_{n-1}) \ne 0 \Big]$$

holds for all $\xi \in C$.

(ii) The number of distinct complex roots of $f_{i,\xi}$ *remains invariant over* C*, which corresponds to*

$$(\forall 1 \le i \le r)(\exists 0 < k_i \le d_i)(\exists 0 \le l_i \le k_i - 1)$$
$$\Big[(\forall k > k_i)\big[f_i^k(x_1, \ldots, x_{n-1}) = 0\big] \wedge f_i^{k_i}(x_1, \ldots, x_{n-1}) \neq 0 \wedge$$
$$(\forall l < l_i)\Big[\mathrm{psc}_l^{x_n}(\hat{f}_i^{k_i}(x_1, \ldots, x_n), D_{x_n}\hat{f}_i^{k_i}(x_1, \ldots, x_n)) = 0\Big] \wedge$$
$$\mathrm{psc}_{l_i}^{x_n}(\hat{f}_i^{k_i}(x_1, \ldots, x_n), D_{x_n}\hat{f}_i^{k_i}(x_1, \ldots, x_n)) \neq 0\Big].$$

holds for all $\xi \in C$.

(iii) The total number of common complex roots of $f_{i,\xi}$ *and* $f_{j,\xi}$ *(counting multiplicity) remains invariant over* C*, which is characterized by*

$$(\forall 1 \le i < j \le r)(\exists 0 < k_i \le d_i)(\exists 0 < k_j \le d_j)(\exists 0 \le m_{i,j} \le \min(d_i, d_j))$$
$$\Big[(\forall k > k_i)\big[f_i^k(x_1, \ldots, x_{n-1}) = 0\big] \wedge f_i^{k_i}(x_1, \ldots, x_{n-1}) \neq 0 \wedge$$
$$(\forall k > k_j)\big[f_j^k(x_1, \ldots, x_{n-1}) = 0\big] \wedge f_j^{k_j}(x_1, \ldots, x_{n-1}) \neq 0 \wedge$$
$$(\forall m < m_{i,j})\big[\mathrm{psc}_m^{x_n}(\hat{f}_i^{k_i}(x_1, \ldots, x_n), \hat{f}_j^{k_j}(x_1, \ldots, x_n)) = 0\big] \wedge$$
$$\big[\mathrm{psc}_{m_{i,j}}^{x_n}(\hat{f}_i^{k_i}(x_1, \ldots, x_n), \hat{f}_j^{k_j}(x_1, \ldots, x_n)) \neq 0\big]\Big],$$

holds for all $\xi \in C$.

□

In summary, given a set of polynomials $\mathcal{F} \subset \mathbb{R}[x_1, \ldots, x_{n-1}][x_n]$ we can compute another set of $(n-1)$-variate polynomials $\mathrm{proj}(\mathcal{F}) \subset \mathbb{R}[x_1, \ldots, x_{n-1}]$, which characterizes the maximal connected $\mathcal{F}$-delineable subsets of $\mathbb{R}^{n-1}$. Hence, we have a projection operator.

Definition 4.17 *Let* $\mathcal{F} = \{f_1, f_2, \ldots, f_r\}$*, where* $f_i \in \mathbb{R}[x_1, \ldots, x_{n-1}][x_n]$*. The projection operator is defined as*

$$\mathrm{proj}(\mathcal{F}) \triangleq \mathrm{proj}_1(\mathcal{F}) \cup \mathrm{proj}_2(\mathcal{F}) \cup \mathrm{proj}_3(\mathcal{F})$$

where

$$\mathrm{proj}_1 \triangleq \big\{ f_i^k(x_1, \ldots, x_{n-1}) \mid 1 \le i \le r,\ 0 \le k \le d_i \big\},$$

$$\mathrm{proj}_2 \triangleq \Big\{ \mathrm{psc}_l^{x_n}\big(\hat{f}_i^k(x_1, \ldots, x_n), D_{x_n}\hat{f}_i^k(x_1, \ldots, x_n)\big) \mid 1 \le i \le r,\ 0 \le l < k \le d_i - 1 \Big\},$$

$$\mathrm{proj}_3 \triangleq \Big\{ \mathrm{psc}_m^{x_n}\big(\hat{f}_i^{k_i}(x_1, \ldots, x_n), \hat{f}_j^{k_j}(x_1, \ldots, x_n)\big) \mid 1 \le i < j \le r,\ 0 \le m \le k_i \le d_i,\ 0 \le m \le k_j \le d_j \Big\}.$$

Suppose that the set $\mathrm{proj}(\mathcal{F})$ is invariant over $C \subset \mathbb{R}^{n-1}$. The invariance of the set $\mathrm{proj}_1(\mathcal{F})$ implies that the degree w.r.t. x_n of the polynomials is constant over C and hence the number of roots of each polynomial is constant over C (condition (i) in Definition 4.16).

The invariance of $\mathrm{proj}_2(\mathcal{F})$ implies that the gcd of each polynomial and its derivative has constant degree according to Lemma 4.3. Together with the invariance of $\mathrm{proj}_1(\mathcal{F})$ the number of distinct zeros of each polynomial in $\mathcal{F}$ is constant, cf. Lemma 4.1 (condition (ii) in Definition 4.16).

The invariance of $\mathrm{proj}_3(\mathcal{F})$ together with the invariance of $\mathrm{proj}_1(\mathcal{F})$ implies that the number of common zeros of each pair of polynomials in $\mathcal{F}$ is constant, cf. Lemma 4.3 (condition (iii) in Definition 4.16).

According to Lemma 4.4 the set $\mathrm{proj}(\mathcal{F})$ characterizes the sets over which there are a constant number of real zeros of the polynomials in $\mathcal{F}$.

The projection operator is technically the most involved part of the CAD algorithm. However, we repeat that the basic idea of the operator is to project the "significant points" of an n-dimensional zero set of a number of polynomials to an $n-1$-dimensional zero set of another set of polynomials.

4.3.4 Sturm Chains

The number of distinct real zeros of a univariate polynomial is important in both the so called base and extension phase of the CAD algorithm. In these phases we need to isolate the real zeros of univariate polynomials. Sturm chains [30, 64] are the appropriate tool to count the number of real zeros of a polynomial in an interval. Knowing the number of real roots it is easy to compute isolating intervals for them. In this subsection we present Sturm chains and comment on how to represent and make arithmetic operations with algebraic numbers.

Definition 4.18 *Let $f_1, \ldots, f_r \in \mathbb{R}[x]$ have the following properties in the interval (a, b):*

(i) $f_k(x) = 0 \;\Rightarrow\; f_{k-1}(x)f_{k+1}(x) < 0, \;\; a < x < b$

(ii) $f_r(x) \neq 0, \;\; a < x < b$

Then $f_1, \ldots, f_r$ is called a Sturm chain in the interval (a, b). If each polynomial of the Sturm chain is multiplied by the same arbitrary polynomial $d(x)$ the chain obtained is called a generalized Sturm chain.

Example 4.14 *Let $f_1 \in \mathbb{R}[x]$. A Sturm chain whose first two elements are f_1 and f_1' gives information about the number of real zeros of f_1 in (a, b). Let $f_2 = f_1'$ and* *consider the following PRS:*

$$\begin{aligned} f_1 &= q_1 f_2 - f_3 \\ f_2 &= q_2 f_3 - f_4 \\ &\vdots \\ f_{r-2} &= q_{r-2} f_{r-1} - f_r \\ f_{r-1} &= q_{r-1} f_r + 0, \end{aligned}$$

where $f_r = \gcd(f_1, f_2)$*. This sequence is a Sturm chain in* (a, b) *by construction if* $f_r \neq 0$ *in* (a, b) *or a generalized Sturm chain in* $(-\infty, \infty)$*, since* $f_1, \ldots, f_r$ *may be seen as a Sturm chain multiplied by the* gcd *of all* f_i*. Observe the reversed sign of each remainder, which guarantees property (i) in Definition 4.18.*

Definition 4.19 *Let* $\{a_i\}_{i=1}^r$ *be a finite sequence of real numbers. Then let* $\operatorname{var}(a_1, \ldots, a_r)$ *denote the number of sign variations in the sequence.*

Example 4.15

$$\operatorname{var}(-1, 0, -2, 0, 0, 4, 3, -1) = 2 \quad \text{and} \quad \operatorname{var}(3, 5, 0, -1, 2, 0, -2, 1) = 4.$$

Theorem 4.2 *Let* $f_1, \ldots, f_r$ *be the chain obtained in Example 4.14 and* $V(x) = \operatorname{var}(f_1(x), \ldots, f_r(x))$*. Then the number of distinct real zeros of* f_1 *in the interval*[3] (a, b) *is*

$$V(a) - V(b).$$

Proof *See [24, 64].* □

Example 4.16 *Let* $f_1 = (x^2 + x + 1)(x + 1)(x - 1)(x - 3)$*. Then* $f_2 = f_1' = 5x^4 - 8x^3 - 9x^2 - 2x + 2$ *and the Sturm chain from Example 4.14 of* f_1 *and* f_2 *becomes:*

$$\begin{aligned} f_1 &= x^5 - 2x^4 - 3x^3 - x^2 + 2x + 3 \\ f_2 &= 5x^4 - 8x^3 - 9x^2 - 2x + 2 \\ f_3 &= \tfrac{46}{25} x^3 + \tfrac{33}{25} x^2 - \tfrac{36}{25} x - \tfrac{79}{25} \\ f_4 &= -\tfrac{6825}{2116} x^2 + \tfrac{2625}{1058} x + \tfrac{37875}{2116} \\ f_5 &= -\tfrac{321632}{29575} x - \tfrac{50784}{4225} \\ f_6 &= -\tfrac{2142075}{190969} \end{aligned}$$

[3] Here a may be $-\infty$ and b may be ∞.

According to Theorem 4.2 we can compute the number of distinct real zeros of f_1 *in* $(-2, 2)$. *We have*

$$V(-2) = \text{var}(-,+,-,+,+,-) = 4, \qquad V(2) = \text{var}(-,-,+,+,-,-) = 2$$

where only the signs of the evaluated Sturm chain is presented. Hence the number of real zeros of f_1 *in* $(-2, 2)$ *is 2.*
The total number of distinct real zeros of f_1 *is 3 since*

$$V(-\infty) = \text{var}(-,+,-,-,+,-) = 4, \qquad V(\infty) = \text{var}(+,+,+,-,-,-) = 1.$$

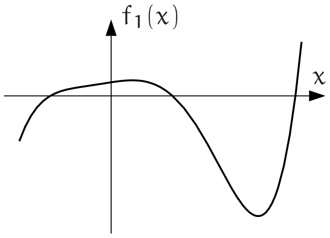


Figure 4.5 *A plot of* $f_1(x)$.

We now have a method for counting the number of distinct real zeros of a univariate polynomial in an interval. Together with a bound on the modulus of the zeros it is easy to state an algorithm which separates or isolates the real zeros of a polynomial to any desired accuracy, e.g., using some bisection method, see [24] and [64].

Real Algebraic Numbers

In the so called extension phase of the CAD-algorithm multivariate polynomials are evaluated over algebraic numbers, i.e., zeros of univariate polynomials over $\mathbb{Z}$. Hence, we need both representations and algorithms for arithmetic operations on these numbers.

A real algebraic number, α, may be represented in a number of different ways. Perhaps the most intuitive way is by a polynomial and an isolating interval. A common representation is an irreducible polynomial, which has α as a zero:

$$\alpha \; : \; [p_\alpha(x), \, I_\alpha],$$

where $p_\alpha \in \mathbb{Z}[x]$, $I_\alpha = (a, b) \in \mathbb{Q}^2$ and I_α contains no other real zero of p_α than α.

The main tool for arithmetic operations with algebraic numbers is resultants. We will not go into details here but there are algorithms for, e.g., addition, multiplication, inverse, and sign evaluation of algebraic numbers.

More about representation and computation in connection with algebraic numbers can be found in [17, 22, 24, 64]. There is also software developed for these calculations. For example SACLIB [12], which is a library of C routines offering algebraic number arithmetic.

4.4 Cylindrical Algebraic Decomposition

In this section we will describe the CAD algorithm and consider some examples of its use. The input of the algorithm is a set, $\mathcal{F}$ of n-variate polynomials and the output is a representation of a decomposition of $\mathbb{R}^n$ into regions over which the polynomials of $\mathcal{F}$ are invariant. The representation also includes a sample point of each region that can be used to determine the actual signs of the polynomials.

4.4.1 The Algorithm

The algorithm can be divided into three phases: projection, base and extension. The projection phase consists of a number of steps, each in which new sets of polynomials is constructed. The zero sets of the resulting polynomials of each step is the projection of "significant" points of the zero set of the preceding polynomials, i.e., self-crossings, isolated points, vertical tangent points etc. In each step the number of variables is decreased by one and hence the projection phase consists of $n-1$ steps.

The base phase consists of isolation of real roots, $\alpha_i \in \mathbb{R}^1$ of the monovariate polynomials, which are the outputs from the projection phase. Each root and one point in the each interval between two roots are chosen as sample points of a decomposition of $\mathbb{R}^1$.

The purpose of the extension phase is to construct sample points of all cells of the CAD of $\mathbb{R}^n$. The extension phase consists of $n-1$ steps. In the first step a sample point, $(\alpha_i, \beta_j) \in \mathbb{R}^2$ of each cell of the stack over the cells of the base phase is constructed. In the following steps the above procedure is repeated until we have sample points in all cells of the CAD of $\mathbb{R}^n$.

Observe that to determine if a real polynomial system has a solution, it is enough to determine the signs of $\mathcal{F}$ in a sample point of each cell since $\mathcal{F}$ is invariant in each cell by construction. The algorithm can be summarized as:

Input: $\mathcal{F} = \{f_1, \dots, f_r\} \subset \mathbb{R}[x_1, \dots, x_n]$.

Output: An $\mathcal{F}$-invariant CAD of $\mathbb{R}^n$.

- Projection:

$$\mathcal{F}_0 = \mathcal{F} \subset \mathbb{R}[x_1, \dots, x_n],\ \mathcal{F}_1 \subset \mathbb{R}[x_1, \dots, x_{n-1}], \dots, \mathcal{F}_{n-1} \subset \mathbb{R}[x_1].$$

- Base: Root isolation and sample point construction for a CAD of $\mathbb{R}^1$.
- Extension: Sample point construction for CADs of $\mathbb{R}^k$, $k = 2, \dots, n$.

We following subsections give a detailed description of each phase.

Projection Phase

In the projection phase the proj-operator is applied recursively $n-1$ times. Let $\mathcal{F} = \{f_1, \ldots, f_r\}$, $\mathrm{proj}^0(\mathcal{F}) = \mathcal{F}$ and $\mathrm{proj}^i(\mathcal{F}) = \mathrm{proj}(\mathrm{proj}^{i-1}(\mathcal{F}))$ for $1 \le i \le n-1$. Then

$$\begin{aligned} \mathcal{F} &\subset \mathbb{R}[x_1, \ldots, x_n] \\ \mathrm{proj}(\mathcal{F}) &\subset \mathbb{R}[x_1, \ldots, x_{n-1}] \\ \mathrm{proj}^2(\mathcal{F}) &\subset \mathbb{R}[x_1, \ldots, x_{n-2}] \\ &\vdots \\ \mathrm{proj}^{n-1}(\mathcal{F}) &\subset \mathbb{R}[x_1], \end{aligned}$$

where the corresponding zero sets are the successive projections of the "significant" points of the original zero set.

Example 4.17 *Consider the* 3D*-sphere of radius 1 centered at* $(2,2,2)$ *which is given by the set of real zeros of*

$$f = (x_1 - 2)^2 + (x_2 - 2)^2 + (x_3 - 2)^3 - 1.$$

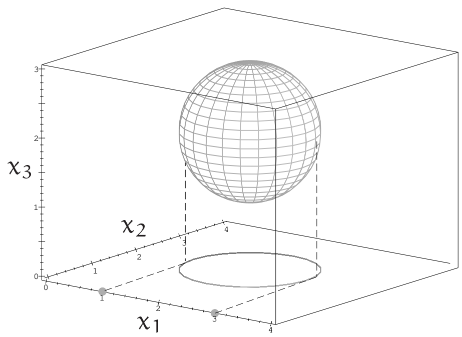


Figure 4.6 *The set of real zeros of* f *and the projections of its "significant" points.*

Now, the projection polynomials are

$$\begin{aligned}
\operatorname{proj}(f) &= \{\, 4((x_1-2)^2+(x_2-2)^2-1),\ (x_1-2)^2+(x_2-2)^2+3 \,\} \\
\operatorname{proj}^2(f) &= \{\, 4(x_1-1)(x_1-3)(x_1^2-4x_1+19), \\
&\qquad 16(x_1^4-8x_1^3+30x_1^2-56+113), \\
&\qquad 4(x_1^2-4x_1+7), \\
&\qquad x_1^2-4x_1+11, \\
&\qquad 256(x_1-1)(x_1-3) \,\}
\end{aligned}$$

The only real zeros of the polynomials in $\operatorname{proj}(f)$ *correspond to the circle in the* x_1x_2*-plane in Figure 4.6 and the only real zeros of the polynomials in* $\operatorname{proj}^2(f)$ *is* 1 *and* 3*, i.e., the gray dots on the* x_1*-axis in Figure 4.6.*

Base Phase

The zeros of the monovariate polynomials in $\operatorname{proj}^{n-1}(\mathcal{F})$ define a sign invariant decomposition of $\mathbb{R}^1$. Enumerate the real zeros of the polynomials in $\operatorname{proj}^{n-1}(\mathcal{F})$ according to

$$-\infty < \xi_1 < \xi_2 < \cdots < \xi_s < +\infty.$$

The $\operatorname{proj}^{n-1}(\mathcal{F})$-invariant decomposition of $\mathbb{R}^1$ consists of these zeros and the intermediate open intervals. The purpose of the base phase is to isolate the above zeros and find sample points for each component in the decomposition, i.e., the zeros and a sample point in each interval. For an open interval we may choose a rational sample point but for a zero we must store an exact representation of an algebraic number, see Subsection 4.3.4. The base phase may thus be described briefly as:

Input: A set of monovariate polynomials, $\operatorname{proj}^{n-1}(\mathcal{F})$.

Output: Sample points of each component of the decomposition of $\mathbb{R}^1$.

- Real root isolation: $\xi_1 \in (u_1, v_1], \ldots, \xi_s \in (u_s, v_s]$, where $u_i, v_i \in \mathbb{Q}$.
- Choice of sample points: $\alpha_1 = u_1$, $\alpha_2 = \xi_1$, $\alpha_3 = u_2, \ldots, \alpha_{2s+1} = v_s + 1$.

Example 4.18 *Given the* $\operatorname{proj}^2(f)$ *set from the projection phase of Example 4.17 we isolate the real roots of its polynomials. The number of real roots of each polynomial may be calculated using Theorem 4.2. This gives the real roots* 1 *and* 3 *with isolating intervals* $(\frac{1}{2}, 2]$ *and* $(2, \frac{7}{2}]$*, respectively. The* x_1*-axis is decomposed into 5 cells with the following sample points:*

$$\tfrac{1}{2},\ 1,\ 2,\ 3,\ \tfrac{7}{2}.$$

Extension Phase

In the extension phase we "lift" a sign invariant decomposition, $\mathcal{D}_{i-1}$ of $\mathbb{R}^{i-1}$ to a sign invariant decomposition, $\mathcal{D}_i$ of $\mathbb{R}^i$ using the technique of the base phase repetitively.

Consider the "lift" from $\mathbb{R}^1$ to $\mathbb{R}^2$. According to how $\mathrm{proj}^{n-2}(\mathcal{F})$ is constructed it is delineable over each cell of $\mathcal{D}_1$. We will construct the sample points of those cells of $\mathcal{D}_2$ which belong to the stack over a cell $C \subset \mathcal{D}_1$. Evaluate the polynomials in $\mathrm{proj}^{n-2}(\mathcal{F})$ over the sample point, α of C. We then get a set of monovariate polynomials in x_2 corresponding to the values of $\mathrm{proj}^{n-2}(\mathcal{F})$ on the "vertical" line $x_1 = \alpha$.

These monovariate polynomials are then treated in the same way as the polynomials in the base phase, i.e., root isolation and choice of sample points. Hence the "lift" from $\mathbb{R}^1$ to $\mathbb{R}^2$ corresponds to the construction of the second component of the sample points of $\mathcal{D}_n$.

Having sample points for all cells of $\mathcal{D}_2$ the above process may be repeated to construct sample points in the cells of $\mathcal{D}_3, \ldots, \mathcal{D}_n$.

A geometrical picture of the steps in the extension phase consists of so to speak raising vertical lines over each sample point of the lower dimensional decomposition and calculate the intersections between these lines with the zero set of the next higher dimensional set of polynomials.

The result of the extension phase is a list of cells (indexed in some way) and their sample points. Hence, the decomposition may be represented as a tree structure where the first level of nodes under the root corresponds to the cells of $\mathbb{R}^1$, the second level of nodes represents the cells of $\mathbb{R}^2$, i.e., the stacks over the cells of $\mathbb{R}^1$ etc., see Figure 4.8. The leaves represents the cells of the CAD of $\mathbb{R}^n$. In each node or leaf a sample point of the corresponding cell is stored. To each level of the tree there are a number of projection polynomials $\mathrm{proj}^k(\mathcal{F})$ whose signs when evaluated over a sample point defines a cell.

Notice that the projection polynomials $\mathrm{proj}^k(\mathcal{F}), k = 0, \ldots, n-1$ together with the sample points describe the set of solutions of any system of polynomial equations and inequalities defined by the original polynomials. The signs of the projection polynomials gives us information about the restrictions on the solution set imposed by the original system on smaller and smaller sets of variables.

Example 4.19 *We now extend the CAD of $\mathbb{R}^1$ from Example 4.18 to a CAD of $\mathbb{R}^2$ and $\mathbb{R}^3$. The base phase gives a decomposition of the x_1-axis into five cells and sample points for each cell, see the top left plot in Figure 4.7.*
Specializing the polynomials of $\mathrm{proj}(f)$ *over each sample point gives five univariate polynomials in x_2. For each specialized polynomial we use the same algorithms as in the base phase to construct sample points of the cells of $\mathbb{R}^2$. We may choose the*

following sample points

$$
\begin{aligned}
&(\tfrac{1}{2}, 2),\\
&(1,1),\ (1,2),\ (1,3),\\
&(2,\tfrac{1}{2}),\ (2,1),\ (2,2),\ (2,3),\ (2,\tfrac{7}{2}),\\
&(3,1),\ (3,2),\ (3,3),\\
&(\tfrac{7}{2}, 2),
\end{aligned}
$$

see the top right plot in Figure 4.7.
In the same way we specialize f *over each sample point of* $\mathbb{R}^2$ *and construct the 25 sample points of the corresponding CAD of* $\mathbb{R}^3$. *In Figure 4.8 the structure of a tree representation of the CAD is given.*

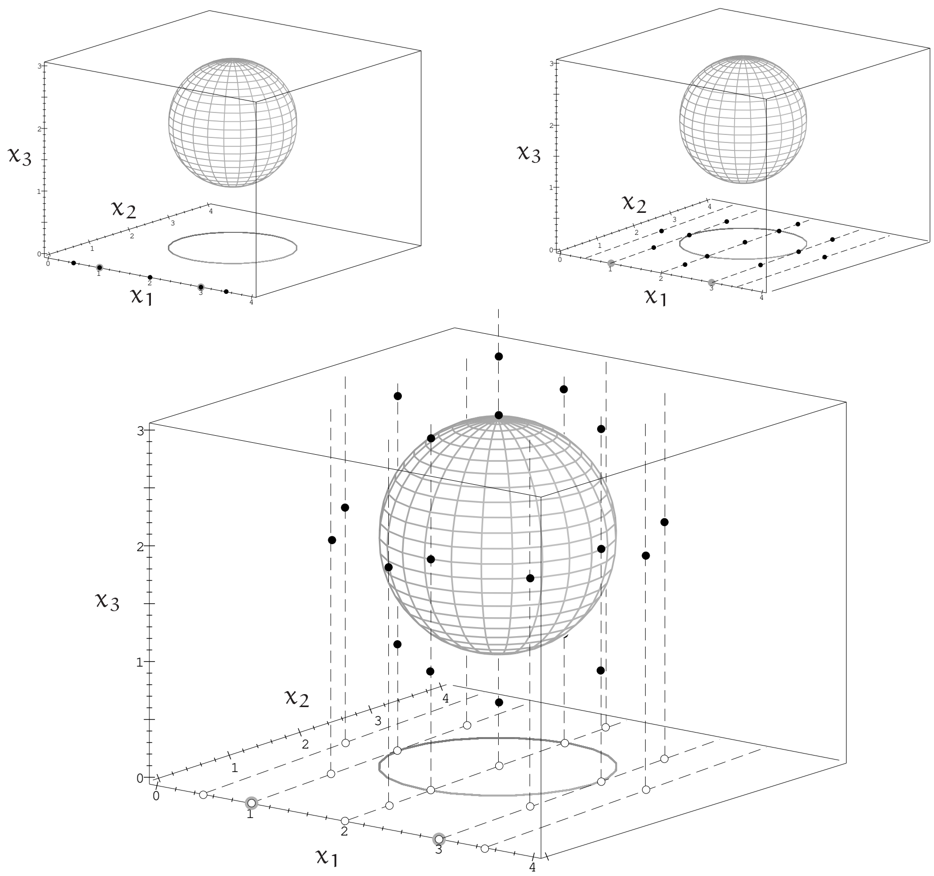


Figure 4.7 *The sample points of each step of the extension phase.*

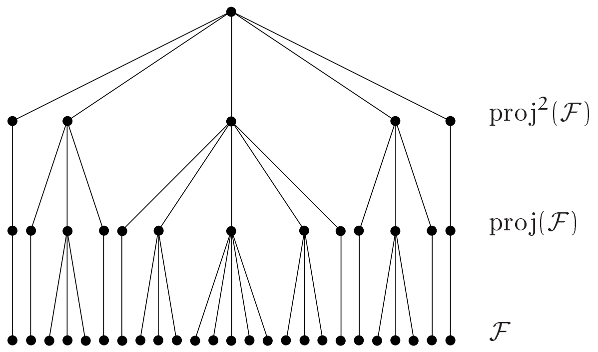


Figure 4.8 A tree representation of the CAD.

Observe that even for this trivial example the number of cells become quite large. In general the number of cells of a CAD grows very quickly as the number of variables increases.

4.4.2 An Example

The following example show how the CAD-algorithm can be used to find solutions to a system of polynomial inequalities. Once a sample point of each sign invariant cell of the decomposition has been computed we only have to evaluate the original polynomials over these sample points to determine if there are any solutions.

Example 4.20 *Given the following polynomials*

$$\begin{aligned} f_1 &= x_2^2 - 2x_1x_2 + x_1^4 \\ f_2 &= (2431x_1 - 3301)x_2 - 2431x_1 + 2685, \end{aligned} \tag{4.1}$$

determine a corresponding CAD of $\mathbb{R}^2$.

- *Projection:*

$$\mathrm{proj}_1(f_1) = \left\{ -2\,x_1, 1, {x_1}^4 \right\}$$

$$\mathrm{proj}_1(f_2) = \{ -2431\,x_1 + 2685, 2431\,x_1 - 3301 \}$$

$$\mathrm{proj}_2(f_1) = \left\{ 4\,{x_1}^2\,(x_1 - 1)\,(x_1 + 1) \right\}$$

$$\mathrm{proj}_2(f_2) = \{\}$$

$$\begin{aligned} \mathrm{proj}_3(\{f_1, f_2\}) = \Big\{ & -x_1 \left(-4862\,x_1 + 5370 + 2431\,{x_1}^4 - 3301\,{x_1}^3\right), \\ & (17\,x_1 - 15)\,(13\,x_1 - 5)\,\big(26741\,{x_1}^4 - 38742\,{x_1}^3 - \\ & 8854\,{x_1}^2 - 51552\,x_1 + 96123\big) \Big\} \end{aligned}$$

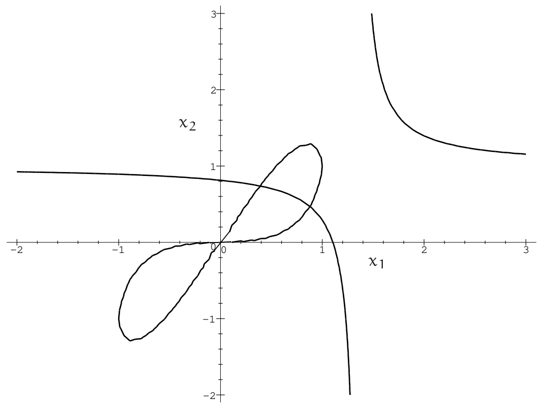


Figure 4.9 *A plot of the real zero set of the polynomials in (4.1).*

- *Base: The real roots of* $\operatorname{proj}(\{f_1, f_2\})$ *are*

$$-1,\ 0,\ \tfrac{5}{13},\ \tfrac{15}{17},\ \alpha \approx 0.93208,\ 1,\ \tfrac{2685}{2431},\ \tfrac{3301}{2431},\ \beta \approx 1.59982$$

where α *and* β *are the real zeros of* $2431x_1^4 - 3301x_1^3 - 4862x_1 + 5370$. *It turns out that we only need five of these roots to determine a CAD of* $\mathbb{R}^2$. *These are*

$$-1,\ 0,\ \tfrac{5}{13},\ \tfrac{15}{17},\ 1,\ \tfrac{3301}{2431}.$$

We also need sample points from each interval between the above roots, e.g.,

$$-2,\ -\tfrac{1}{2},\ \tfrac{1}{4},\ \tfrac{1}{2},\ \tfrac{9}{10},\ \tfrac{5}{4},\ 2.$$

Hence the base phase produces 13 sample points.

- *Extension: We compute the sample points for the stack over cell 7* ($x_1 = \frac{1}{2}$) *of the decomposition of* $\mathbb{R}^1$ *to illustrate the procedure. We have*

$$\begin{aligned} f_1(\tfrac{1}{2}, x_2) &= {x_2}^2 - x_2 + \tfrac{1}{16} \\ f_2(\tfrac{1}{2}, x_2) &= -\tfrac{4171}{2}x_2 + \tfrac{2939}{2}, \end{aligned}$$

with real roots

$$\tfrac{1}{2} \pm \tfrac{1}{4}\sqrt{3} \qquad \text{and} \qquad \tfrac{2939}{4171}$$

respectively. Together with four points in the intermediate intervals we get seven sample points, see Table 4.1.

The information in one row of Table 4.1 is typically what is stored in a leaf of a tree representation of a CAD, i.e., the sample point and the signs of the polynomials. The whole CAD of $\mathbb{R}^3$ *consists of 63 cells, see Figure 4.10. Given the sign of* f_1 *and* f_2 *in all cells we can "solve" any real polynomial system defined by* f_1 *and* f_2.

Sample point (x_1, x_2)	sign(f_1)	sign(f_2)
$(\frac{1}{2}, 0)$	$+$	$+$
$(\frac{1}{2}, \frac{1}{2} - \frac{1}{4}\sqrt{3})$	0	$+$
$(\frac{1}{2}, \frac{1}{2})$	$-$	$+$
$(\frac{1}{2}, \frac{2939}{4171})$	$-$	0
$(\frac{1}{2}, \frac{3}{4})$	$-$	$-$
$(\frac{1}{2}, \frac{1}{2} + \frac{1}{4}\sqrt{3})$	0	$-$
$(\frac{1}{2}, 2)$	$+$	$-$

Table 4.1 *Sample points and signs of f_1 and f_2 over cell 7.*

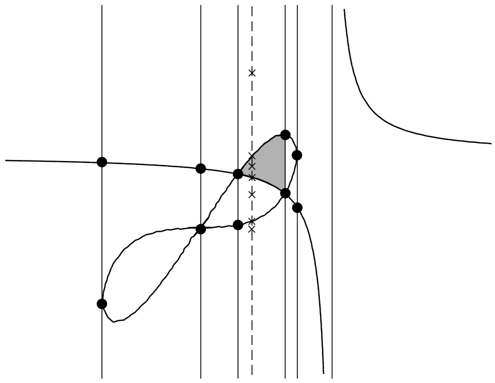


Figure 4.10 *The CAD, where the dashed line and the $\times$ corresponds to $x_1 = \frac{1}{2}$ and the sample points over cell 7 of the decomposition of $\mathbb{R}^1$. The gray region corresponds to row five in Table 4.1 where $f_1 < 0,\ f_2 < 0$.*

The point $x_1 = \frac{1}{2}, x_2 = \frac{3}{4}$ is one solution to $f_1 < 0, f_2 < 0$.

In the above example we were able to avoid evaluating polynomials over irrational algebraic numbers. This is one of the most time consuming parts in an implementation of the CAD algorithm. However, computations with algebraic numbers can be avoided if we only are interested in *open* solution sets of systems of polynomial strict inequalities [77].

Observe that given a CAD generated by a set of real polynomials, it is possible to solve any real polynomial system defined by these polynomials. A CAD can also be utilized in connection with quantifier elimination, which is the main tool in Part III of the thesis. This is the reason for the detailed treatment of CAD in this chapter.

4.5 Quantifier Elimination

Many problems can be formulated as formulas involving polynomial equations, inequalities, quantifiers ($\exists, \forall$) and Boolean operators (e.g., $\vee, \wedge$). Formally these expressions constitute a sentence in the so called first-order theory of real closed fields. The problem of *quantifier elimination* consists of finding an equivalent expression without any quantified variables. In the late forties Tarski [79] gave a general algorithm for quantifier elimination in expressions of the abovementioned type, but this algorithm had a very high complexity and is impractical for most problems. In the mid seventies Collins [18] presented a method that decreased the algorithmic complexity and since then the algorithmic development has proceeded. Recently, more effective algorithms have made it possible to solve non-trivial quantifier elimination problems on workstations [19]. Introductions to quantifier elimination are given in, e.g., [24, 64]. For an extensive bibliography on the subject, see [3]. The aim of this section is to give a brief introduction to QE and its connection to CAD.

Formulas

By a quantifier free *Boolean* formula we mean a formula consisting of polynomial equations ($f_i(x) = 0$) and inequalities ($f_j(x) < 0$) combined using the Boolean operators $\wedge$ (and), $\vee$ (or), $\neg$ (not) and $\rightarrow$ (implies).

Definition 4.20 *A formula in the first-order theory of real closed fields is an expression in the variables* $x = (x_1, \ldots, x_n)$ *of the following type*

$$(Q_1 x_1 \ldots Q_s x_s)\mathcal{F}(f_1(x), \ldots, f_r(x)),$$

where Q_i *is one of the quantifiers* $\forall$ *(for all) or* $\exists$ *(there exists) and* $\mathcal{F}(f_1(x), \ldots, f_r(x))$ *is a quantifier free Boolean formula.*

The term *sentence* is often used synonymously for formula. Note that there is always an implicit assumption that all variables are considered to be real.

A quantifier free Boolean formula $\mathcal{F}(f_1(x), \ldots, f_r(x))$ defines a semialgebraic set, cf. Definition 4.4. Even though the quantifiers give expressive power they do not enlarge the class of sets defined by quantifier free Boolean formulas, i.e., formulas including quantifiers still define semialgebraic sets. This implies that given a formula including quantifiers there is always a logically equivalent quantifier free Boolean formula, a result first shown by Tarski [79].

Example 4.21 *The following expression is a formula in the first-order theory of real closed fields:*

$$\exists y \, \forall x \left[x^2 + y^2 > 1 \ \wedge \ (x-1)^2 + (y-1)^2 > 1 \right],$$

which can be read as:

> *There exists a real number* y *such that for all real numbers* x *we have that* $x^2 + y^2 > 1$ *and* $(x-1)^2 + (y-1)^2 > 1$.

This particular formula is true since both inequalities are satisfied for all real x *if* $y = 3$*, for instance.*

The above problem is called a *decision problem* since all variables are bounded by quantifiers and the problem consists of deciding if the formula is true or false. If there are some *free* or unquantified variables as well one has a general *quantifier elimination* problem. As mentioned above, it is always possible to eliminate quantified variables.

Example 4.22 *The following is a general quantifier elimination problem.*

$$\forall x \left[x > 0 \ \rightarrow \ x^2 + ax + b > 0 \right],$$

which can be read as:

> *For all real numbers* x*, if* $x > 0$ *then* $x^2 + ax + b > 0$.

Performing quantifier elimination we obtain the following equivalent expression

$$\left[4b - a^2 > 0 \ \vee \ [\, a \geq 0 \ \wedge \ b \geq 0] \right],$$

which are the conditions on the coefficients of a second order polynomial so that it only takes positive values for $x > 0$.

The Boolean operator $\rightarrow$ (implies) is used to get more structured formulas but can be expressed using other Boolean operators. Let A and B be two formulas. Then

$$A \ \rightarrow \ B \quad \Leftrightarrow \quad \neg A \ \vee \ B.$$

Many properties of semialgebraic sets can be exploited by eliminating quantifiers in formulas, e.g., set inclusion, projection and set adjacency.

Example 4.23 *Consider the semialgebraic sets*

$$\mathcal{A} = \left\{ (x, y) \in \mathbb{R}^2 \,|\, x^2 + y^2 < 1 \right\} \quad \textit{and} \quad \mathcal{B} = \left\{ (x, y) \in \mathbb{R}^2 \,|\, y > x^4 - 2 \right\}.$$

Is $\mathcal{A} \subset \mathcal{B}$? What is the projection of $\mathcal{B}$ onto the y-axis? The corresponding formulas and their equivalent quantifier free counterparts are

$$\forall x \, \forall y \, \left[x^2 + y^2 < 1 \; \rightarrow \; y > x^4 - 2 \right] \quad \Leftrightarrow \quad \text{TRUE},$$
$$\exists x \, \left[y > x^4 - 2 \right] \quad \Leftrightarrow \quad [y > -2].$$

Hence, $\mathcal{A} \subset \mathcal{B}$ and the projection of $\mathcal{B}$ onto the y-axis is $\left\{ y \in \mathbb{R} \,|\, y > -2 \right\}$.

The quantified variables are implicitly assumed to range over all real numbers. Often we need to express that some of the bounded variables are constrained to a semialgebraic set, e.g., an interval. Then the following equivalences turns out to be useful

$$\exists (u \in \mathcal{U}) \left[\mathcal{X}(x, u) \right] \quad \Leftrightarrow \quad \exists u \left[\mathcal{X}(x, u) \, \wedge \, \mathcal{U}(u) \right], \tag{4.2}$$

$$\forall (u \in \mathcal{U}) \left[\mathcal{X}(x, u) \right] \quad \Leftrightarrow \quad \forall u \left[\mathcal{U}(u) \, \rightarrow \, \mathcal{X}(x, u) \right]. \tag{4.3}$$

More about Boolean formulas and logic can be found in, e.g., [80].

Relations between QE and CAD

For a decision problem the truth value of a formula can be decided once a CAD defined by the involved polynomials has been computed. The quantifier free Boolean part of the original formula $\mathcal{F}(f_1(x), \ldots, f_r(x))$ is evaluated in each sample point and depending on how a variable is quantified $\mathcal{F}(f_1(x), \ldots, f_r(x))$ has to be true for some or all sample points.

For the quantifier elimination problem we evaluate $\mathcal{F}(f_1(x), \ldots, f_r(x))$ over the sample points. The cells corresponding to the sample points for which this formula is true can be characterized by the sign of the polynomials from the projection phase of the CAD algorithm. The solution formula can then be constructed by combining such partial formulas, see [43].

Example 4.24 *Consider the polynomials in Example 4.20 and their corresponding CAD shown in Figure 4.10. Suppose that we have the following decision problem*

$$\exists x_1 \, \exists x_2 \, [f_1 < 0 \, \wedge \, f_2 < 0]. \tag{4.4}$$

This particular CAD consists of 63 cells and by construction we know that f_1 and f_2 are invariant over each cell. Since x_1 and x_2 are bounded by existential quantifiers we only have to evaluate them in each sample point and check if for some point the inequalities in (4.4) are satisfied. In this particular case we know that the formula is true according row five in Table 4.1.
Now, suppose we have the formula

$$\forall x_2 \, [f_1 > 0 \, \wedge \, f_2 < 0], \tag{4.5}$$

which is a quantifier elimination problem.
Consider the cells of the induced decomposition of the x_1*-axis in Figure 4.10. The formula is true for those cells over which there is a stack with* $f_1 > 0$ *and* $f_2 < 0$ *for all its sections and sectors. Hence, we have to find the cells with this property (by sign evaluation in sample points) and describe them in terms of their defining polynomials, i.e., the polynomials from the projection phase. In Figure 4.10 it is easy to see that the only stack in which* $f_1 > 0$ *and* $f_2 < 0$ *is the rightmost vertical line. This stack is described by the zero of the projection polynomial* $2431x_1 - 3301$*. Hence, a quantifier free equivalent formula to (4.5) is*

$$2431x_1 - 3301 = 0.$$

To perform quantifier elimination in the non-trivial examples of this thesis we have used the program QEPCAD, developed by Hoon Hong et al. at RISC in Austria, see [19]. In Part III we use QE and demonstrate its applicability to problems in control theory.

Part II

Issues in Differential Algebraic Systems Theory

5

Introduction

In this part of the thesis a few problems relating to differential algebraic systems theory are addressed. We will point out some problems that arise when treating systems of polynomial differential equations using the language of differential algebra. In particular we will see what happens when we search for the input-output equations, which are possible to derive from the original system. Most of the problems are related to computational difficulties, e.g., the existence of a finite set of differential equations describing the system.

Differential algebra made its way into systems theory via the discoveries by Michel Fliess in the mid eighties and has now grown quite extensive. A very good survey is given in [27]. Several authors have studied how the algorithms of differential algebra, mostly elimination theory, can be used in systems theory. Let us here mention [25, 33, 34, 63, 67]. The results in this part of the thesis can be found in condensed form in [29].

5.1 Representations of Dynamical Systems

Consider the class of single-input, single-output (SISO) systems which can be represented by the following classical state space description:

$$\begin{aligned} \dot{x}_1 &= f_1(x, u) \\ &\vdots \\ \dot{x}_n &= f_n(x, u) \\ y &= h(x, u), \end{aligned} \tag{5.1}$$

where $x = (x_1, \ldots, x_n)$ and $f_1, \ldots, f_n, h \in k[x, u]$, in other words, we consider a set of polynomial differential equations. This represents a rather large class of systems and it can be shown that systems where the nonlinearities are not originally polynomial may be rewritten on this polynomial form if the nonlinearities themselves are solutions to algebraic differential equations. For more details on this we refer the reader to [55, 74].

One way of considering the state space equations is as generators of a differential ideal. This is the motivation for the following definition.

Definition 5.1 *To a given state space description (5.1) of a system we associate a differential ideal*

$$\Sigma = [\dot{x}_1 - f_1(x, u), \ldots, \dot{x}_n - f_n(x, u), y - h(x, u)], \tag{5.2}$$

called the state space differential ideal of system (5.1).

We now turn our interest to which differential equations in only y and u that belong to Σ, i.e., which input-output equations we can derive from the state space description (5.1). Mathematically speaking this means that we look for $\Sigma^c = \Sigma \cap k\{u, y\}$, where superscript c stands for *contraction* [75], a mathematical operation we will not discuss here. The intersection of two differential ideals can be shown to be a differential ideal [50]. Thus the set Σ^c of all *input-output polynomials* or *input-output relations*, derivable from system (5.1), is a differential ideal.

Definition 5.2 *The differential ideal $\Sigma^c = \Sigma \cap k\{u, y\}$ is called the input-output ideal of system (5.1).*

The input-output equations of a system are important in problems concerning, e.g., identifiability, observability [32, 37] and when one tries to decide if two systems of state equations are equivalent. To simplify notation we often let subscripts indicate differentiation w.r.t. time for input and output variables, i.e., $y_0 = y, y_1 = \dot{y}, \ldots, y_j = y^{(j)}$.

Example 5.1 *Consider the state space equations*

$$\begin{aligned} \dot{x}_1 - x_2 x_1 - u &= 0 \\ \dot{x}_2 &= 0 \\ y - x_1 &= 0. \end{aligned} \tag{5.3}$$

If we differentiate the last equation and substitute into the first one we get

$$\dot{y} - x_2 y = u. \tag{5.4}$$

This is a first order linear differential equation with the constant coefficient $-x_2$, since $\dot{x}_2 = 0$. If we differentiate y twice

$$y = x_1, \quad \dot{y} = \dot{x}_1 = x_1 x_2 + u, \quad \ddot{y} = \dot{x}_1 x_2 + x_1 \dot{x}_2 + \dot{u} = x_1 x_2^2 + x_2 u + \dot{u}$$

and eliminate x_1 *and* x_2 *from these equations we get*

$$y\ddot{y} - \dot{y}(\dot{y} - u) - y\dot{u} = 0$$

or with the subscript notation

$$y_0 y_2 - y_1(y_1 - u_0) - y_0 u_1 = 0.$$

The left hand side differential polynomial p *is an example of an element in* Σ^c*, which we found by elimination. Let* A, B *and* C *denote the left hand side differential polynomials of system (5.3). Then* p *can be rewritten as*

$$p = y\ddot{y} - \dot{y}^2 + \dot{y}u - y\dot{u} = -\dot{y}A + y\dot{A} + x_1 yB + x_2\dot{y}C - (x_2 y + \dot{y})\dot{C} + y\ddot{C} \in [A, B, C] = \Sigma$$

which again shows that $p \in \Sigma$.

A physical interpretation of the state equations in the above example could be a model of a DC-motor where the input is an applied voltage and the output is the angular velocity of the motor shaft. Since $\dot{x}_2 = 0$, we can consider x_2 as a fixed parameter. Equation (5.4) can be recognized as a first order system with time constant $-\frac{1}{x_2}$. This corresponds to the transient behavior of the angular velocity modeled as a first order system. More general, the equations (5.3) can be seen as a state space description of all systems, which is described by a linear, first order differential equation with constant coefficients.

In Example 5.1 we found a differential polynomial of differential order two in $k\{u, y\}$. This is the differential polynomial in $k\{u, y\}$ of lowest order that can be derived from system (5.3). It can be shown [28] that starting from the state space representation (5.1) we can always find an element in Σ^c of the same (or possibly lower) differential order than the number of state variables. In the next section we will see how this can be accomplished in the general case.

The aim of this part of the thesis is to study the input-output ideal Σ^c. Is $\Sigma^c = [p]$, where p is the differential polynomial of lowest order in Σ^c? If the answer to this question is negative, can we then compute a finite set of generators for Σ^c at all? In the non-differential case all ideals are finitely generated according to Theorem 2.2. Theorem 3.1 tells us that differential ideals do not have this nice property in general and a valid question is: Does Σ^c always have a finite set of generators? One may anticipate this since Σ is always finitely generated by construction. We will use both non-differential algorithmic tools as Gröbner bases and differential algebraic techniques such as characteristic sets from Part I to answer these questions and to reveal some of the structure of the input-output ideal, Σ^c.

5.2 The Lowest Order Input-Output Relation

The computation of the input-output relation of lowest order for a given state space description, as in Example 5.1, can be generalized by the use of Gröbner bases [28]. We will briefly describe this method.

Let us first consider the case of two state variables, $x = (x_1, x_2)$:

$$\begin{aligned} \dot{x}_1 &= f_1(x,u) \\ \dot{x}_2 &= f_2(x,u) \\ y &= h(x,u). \end{aligned}$$

Differentiating y gives

$$\dot{y} = \dot{x}_1 \frac{\partial h}{\partial x_1} + \dot{x}_2 \frac{\partial h}{\partial x_2} + \dot{u}\frac{\partial h}{\partial u}.$$

By use of the state equations the state derivatives can be eliminated giving

$$\dot{y} = f_1 \frac{\partial h}{\partial x_1} + f_2 \frac{\partial h}{\partial x_2} + \dot{u}\frac{\partial h}{\partial u}.$$

We notice that $\dot{y}$ is a function of x_1, x_2, u and $\dot{u}$. With the subscript notation for the output y we have $y_1 = f_1 \frac{\partial h}{\partial x_1} + f_2 \frac{\partial h}{\partial x_2} + \dot{u}\frac{\partial h}{\partial u}$. We now differentiate $\dot{y} = y_1(x, y, u, \dot{u})$ and repeat the above process

$$\ddot{y} = \dot{x}_1 \frac{\partial y_1}{\partial x_1} + \dot{x}_2 \frac{\partial y_1}{\partial x_2} + \dot{u}\frac{\partial y_1}{\partial u} + \ddot{u}\frac{\partial y_1}{\partial \dot{u}} = f_1 \frac{\partial y_1}{\partial x_1} + f_2 \frac{\partial y_1}{\partial x_2} + \dot{u}\frac{\partial y_1}{\partial u} + \ddot{u}\frac{\partial y_1}{\partial \dot{u}}.$$

This is again an expression in x_1, x_2, u and the derivatives of u. We observe that each derivation can be considered as applying the operator

$$f_1 \frac{\partial}{\partial x_1} + f_2 \frac{\partial}{\partial x_2} + \sum_{i=1}^{N} u^{(i)} \frac{\partial}{\partial u^{(i-1)}}$$

to the former derivative of y, where N is taken big enough.

The generalization of this operator to the case with n state variables is immediate.

Definition 5.3 *The extended Lie-derivative operator is defined by*

$$L_f = \sum_{i=1}^{n} f_i \frac{\partial}{\partial x_i} + \sum_{i=1}^{\infty} u^{(i)} \frac{\partial}{\partial u^{(i-1)}}.$$

The key observation now is that all derivatives of y can be obtained by repeated use of the Lie-derivative operator and that the result is a function of the state variables x, the input u, and the derivatives of u.

After applying the extended Lie-derivative operator n times we have $n+1$ polynomials

$$y_0 = y = h(x,u), \quad y_1 = \dot{y} = L_f h(x,u), \quad \ldots \quad , y_n = y^{(n)} = L_f^n h(x,u).$$

Using Gröbner bases it is always possible to eliminate the n variables $x_1, \ldots, x_n$ from the above $n+1$ equations giving a polynomial in y, u and its derivatives, i.e., a differential equation which describes the input-output behavior of the system.

Definition 5.4 *Let* $h_j = L_f^j h(x, \bar{u})$, *where* $x = (x_1, \dots, x_n)$ *and* $\bar{u} = (u_0, \dots, u_N)$, *i.e.,* u *and its* N *first derivatives. We define the ideal*

$$\mathcal{O}_j \triangleq \langle\, y_0 - h_0(x, \bar{u}), \dots, y_j - h_j(x, \bar{u})\,\rangle \subset k\,[y_0, \dots, y_j, x, \bar{u}].$$

Theorem 5.1 $\mathcal{O}_n \cap k\,[y_0, \dots, y_n, \bar{u}]$ *contains at least one input-output relation for (5.1).*

Proof *See [28].* □

To find this input-output equation we choose an ordering with $y_0, \dots, y_n$ and $\bar{u}$ lower than $x_1, \dots, x_n$ and compute a lex Gröbner basis for $\mathcal{O}_n$. We then get at least one input-output equation in the Gröbner basis according to Theorem 2.7 and 5.1.

To find the input-output equation of lowest degree the above procedure is carried out repeatedly for $\mathcal{O}_j$, $j = 1, 2, \dots$ terminating when the Gröbner basis contains an input-output equation. It can be shown that when the process stops we get one polynomial, which is unique [28].

Note that the input-output equation is just a dependency relation over $k\langle u\rangle$ of $h, L_f h, \dots, L_f^n h$.

Example 5.2 *We look at the same system as in Example 5.1. Then*

$$h_0 = x_1, \quad h_1 = x_1 x_2 + u_0, \quad h_2 = x_1 x_2^2 + x_2 u_0 + u_1$$

$$\mathcal{O}_2 = \langle\, y_0 - x_1, y_1 - x_1 x_2 - u_0, y_2 - x_1 x_2^2 - x_2 u_0 - u_1\,\rangle$$

Let $x_2 > x_1 > y_2 > y_1 > y_0 > u_1 > u_0$ *and calculate a Gröbner basis w.r.t. this ordering, which can be made in, e.g.,* MAPLE *with the command* `gbasis`.

$$\mathrm{GB}(\mathcal{O}_2) = \{-y_2 - x_2 y_1 + u_1, -y_1 + x_2 y_0 + u_0, -y_0 + x_1, y_0 y_2 - y_1^2 + y_1 u_0 - y_0 u_1\}.$$

Among the elements in the Gröbner basis we recognize the last one as $p = y_0 y_2 - y_1^2 + y_1 u_0 - y_0 u_1$.

6

Input-Output Descriptions and Differential Algebra

We have mentioned that differential rings may be considered as non-differential rings in infinitely many variables. The problem is that most of the methods and theorems we have presented so far, e.g., Gröbner bases, are limited to the case of finitely many variables. The exception is the method of characteristic sets.

This chapter is organized as follows. Section 6.1 and 6.2 display some properties of the system in Example 5.1, which are probably rather surprising and counter-intuitive to the non-expert. Section 6.3 and 6.4 explain how most of the problems that arose can be resolved. Section 6.5 describes some problems that are still open and summarizes the results in the chapter.

6.1 The Simplest Description Fails

Having a method to compute the input-output polynomial of lowest order, denoted p, from a classical state space description we can pose the question: Is $\Sigma^c = [\,p\,]$? If the answer is negative, then how many generators do we need? Is Σ^c even finitely generated? We start our investigation by considering a particular system.

Example 6.1 *Consider the same system as in Example 5.1 but without an input*

$$\dot{x}_1 = x_1 x_2, \quad \dot{x}_2 = 0, \quad y = x_1. \tag{6.1}$$

The equations correspond to the following state space differential ideal

$$\Sigma = [\,\dot{x}_1 - x_1 x_2, \dot{x}_2, y - x_1\,]$$

and the (input-)output differential equation of lowest order in Σ *is*

$$y\ddot{y} - \dot{y}^2 = 0.$$

Using subscripts to denote derivation we let $p = y_0 y_2 - y_1^2$.

The differential polynomial $p = y_0 y_2 - y_1^2$ has a special property that we will exploit. The following definitions can be found in [50].

Definition 6.1 *Let* $M = y_{\mu_1}^{\nu_1} y_{\mu_2}^{\nu_2} \ldots y_{\mu_m}^{\nu_m} \in k\{y\}$ *be a differential monomial. Then,* M *is said to have weight* $\sum_{i=1}^{m} \mu_i \nu_i$. *Notation:* $\mathrm{wt}(M)$.

Definition 6.2 *A differential polynomial* $p \in k\{y\}$ *is said to be isobaric of weight* w *if all its monomials have the same weight* w. *We then use the notation* $\mathrm{wt}(p) = w$.

Example 6.2 *The differential polynomials*

$$g_1 = y_0 y_1 y_3 - y_2^2, \quad g_2 = y_1 y_2 y_3 - y_3^2 + y_1^6, \quad g_3 = y_0 y_2 - y_1^2$$

are all isobaric and

$$\mathrm{wt}(g_1) = 4, \quad \mathrm{wt}(g_2) = 6, \quad \mathrm{wt}(g_3) = 2.$$

Lemma 6.1 *The derivative* $\dot{p}$ *of an isobaric polynomial* p *is isobaric and* $\mathrm{wt}(\dot{p}) = \mathrm{wt}(p) + 1$

Proof *Consider an arbitrary differential monomial* $M = y_{\mu_1}^{\nu_1} y_{\mu_2}^{\nu_2} \ldots y_{\mu_m}^{\nu_m}$ *of weight* $\mathrm{wt}(M) = \sum_{i=1}^{m} \mu_i \nu_i$. *Use the rule for derivation of a product and differentiate* M:

$$\begin{aligned}\dot{M} = (\nu_1 y_{\mu_1}^{\nu_1 - 1} y_{\mu_1+1}) y_{\mu_2}^{\nu_2} \ldots y_{\mu_m}^{\nu_m} &+ \\ y_{\mu_1}^{\nu_1} (\nu_2 y_{\mu_2}^{\nu_2 - 1} y_{\mu_2+1}) \ldots y_{\mu_m}^{\nu_m} &+ \\ \ldots + y_{\mu_1}^{\nu_1} y_{\mu_2}^{\nu_2} \ldots (\nu_m y_{\mu_m}^{\nu_m - 1} y_{\mu_m+1})&.\end{aligned}$$

We see that $\dot{M}$ *is an isobaric polynomial and*

$$\mathrm{wt}(\dot{M}) = \sum_{i=1}^{m} \mu_i \nu_i + 1 = \mathrm{wt}(M) + 1$$

If a polynomial p *is isobaric of weight* $\mathrm{wt}(p)$ *it consists of monomials of equal weight. The derivative of such a polynomial is a sum of derivatives of the monomials, all being isobaric polynomials of weight* $\mathrm{wt}(p) + 1$ *and the lemma follows.* □

Lemma 6.2 *Let* f *and* $g \in k\{y\}$ *be isobaric polynomials, then* $f\,g$ *is isobaric with weight* $\mathrm{wt}(f) + \mathrm{wt}(g)$.

Proof *Consider two monomials*

$$M_1 = y_{\mu_{11}}^{\nu_{11}} y_{\mu_{12}}^{\nu_{12}} \dots y_{\mu_{1m}}^{\nu_{1m}} \quad \textit{and} \quad M_2 = y_{\mu_{21}}^{\nu_{21}} y_{\mu_{22}}^{\nu_{22}} \dots y_{\mu_{2n}}^{\nu_{2n}}.$$

The product $M_1M_2 = y_{\mu_{11}}^{\nu_{11}} y_{\mu_{12}}^{\nu_{12}} \dots y_{\mu_{1m}}^{\nu_{1m}} y_{\mu_{21}}^{\nu_{21}} y_{\mu_{22}}^{\nu_{22}} \dots y_{\mu_{2n}}^{\nu_{2n}}$ *of these is again a monomial. The weight of this polynomial is*

$$\mathrm{wt}(M_1M_2) = \sum_{i=1}^{m} \mu_{1i}\nu_{1i} + \sum_{i=1}^{n} \mu_{2i}\nu_{2i} = \mathrm{wt}(M_1) + \mathrm{wt}(M_2).$$

The product of two isobaric polynomials f *and* g *is a sum of monomial products where each monomial product consists of one monomial from* f *with weight* $\mathrm{wt}(f)$ *and one from* g *with weight* $\mathrm{wt}(g)$, *hence every monomial product has weight* $\mathrm{wt}(f) + \mathrm{wt}(g)$ *and thus the total product is isobaric with weight* $\mathrm{wt}(f) + \mathrm{wt}(g)$. □

The input-output polynomial of lowest degree, $y_0y_2 - y_1^2$ from Example 6.1 is isobaric. Consider a differential ideal $[\,p\,]$, where p is an isobaric polynomial. The isobaric property of the generator of this principal differential ideal can help us solve the membership problem for such differential ideals, which is shown in the following theorem.

We introduce the notation

$$P_j \triangleq \langle p, \dot{p}, \dots, p^{(j)} \rangle \subset k[y_0, \dots, y_{\mathrm{wt}(p)+j}].$$

Theorem 6.1 *Let* $[\,p\,] \subset k\{y\}$, *where* p *is isobaric with weight* $\mathrm{wt}(p)$ *and* $f \in k\{y\}$. *If the monomials of* f *have weight less than or equal to* $w \geq \mathrm{wt}(p)$ *then*

$$f \in [\,p\,] \quad \Leftrightarrow \quad f \in P_{w-\mathrm{wt}(p)}.$$

If $w < \mathrm{wt}(p)$ *then* $f \notin [\,p\,]$.

Proof *Clearly* $f \in P_{w-\mathrm{wt}(p)}$ *implies* $f \in [\,p\,]$, *since* $P_{w-\mathrm{wt}(p)} \subset [\,p\,]$. *Going the other direction we know*

$$f \in [\,p\,] \quad \Rightarrow \quad f = \sum_{i=0}^{N} \alpha_i p^{(i)},$$

where N *is big enough but finite (*f *is a polynomial) and* $\alpha_i \in k\{y\}$. *Considering each term in this sum we notice that* $\alpha_i p^{(i)}$ *consists of monomials of weight equal to or greater than* $\mathrm{wt}(p^{(i)})$. *From this it immediately follows that if* $f \in [\,p\,]$ *it cannot contain monomials of weight less than* $\mathrm{wt}(p)$. *Furthermore, the monomials in* f *of weight* $\mathrm{wt}(p)$ *must be equal to* $\alpha_{00}p$, *where* $\mathrm{wt}(\alpha_{00}) = 0$, *since all the other terms in the sum defining* f *consist of monomials of weight greater than* $\mathrm{wt}(p)$. *In the same way we conclude that the monomials in* f *of weight* $\mathrm{wt}(p) + 1$ *must be written in the form* $\alpha_{10}p + \alpha_{11}\dot{p}, \quad \mathrm{wt}(\alpha_{10}) = 1, \mathrm{wt}(\alpha_{11}) = 0$. *Continuing in this way we see that the monomials of weight* w *in* f *have to be written in the form*

$$\alpha_{w-\mathrm{wt}(p),0}\,p + \alpha_{w-\mathrm{wt}(p),1}\,\dot{p} + \dots + \alpha_{w-\mathrm{wt}(p),w-\mathrm{wt}(p)}\,p^{(w-\mathrm{wt}(p))},$$

where

$$\mathrm{wt}(\alpha_{w-\mathrm{wt}(p),i}) = w - \mathrm{wt}(p) - i, \quad i = 1, \ldots, w - \mathrm{wt}(p).$$

Higher derivatives of p *only contribute with monomials of weight bigger than* w. *This means that if* $f \in [p]$ *and the monomials of* f *have weight less than or equal to* w *then* $f \in P_{w-\mathrm{wt}(p)}$. □

As already mentioned this theorem solves the membership problem in the case of differential principal ideals with isobaric generator. We will now use this to show the main result of this section: the differential ideal $[y_0y_2 - y_1^2]$, constructed from the system in Example 6.1, is not prime. We carry out the calculations in terms of an example.

Example 6.3

A differential ideal, A *is not prime if there exists some* $f \notin A$ *and* $g \notin A$ *but* $fg \in A$. *We claim that* $f = y_1$ *and* $g = y_1y_3 - y_2^2$ *are such elements for* $[y_0y_2 - y_1^2]$.
The differential polynomial $f = y_1 \notin [y_0y_2 - y_1^2]$ *since any member of* $[y_0y_2 - y_1^2]$ *has monomials of at least weight 2.*
To decide if the differential polynomial $g = y_1y_3 - y_2^2 \in [y_0y_2 - y_1^2]$ *we use Theorem 6.1. Thus, we have to check if* $g \in P_2$, *i.e., if*

$$y_1y_3 - y_2^2 \in \langle y_0y_2 - y_1^2, y_0y_3 - y_1y_2, y_0y_4 - y_2^2 \rangle.$$

Similarly, the question if $fg \in [y_0y_2 - y_1^2]$ *can be answered by checking if* $f\,g \in P_3$ *since the weight of the monomials in* fg *are 5 and* $\mathrm{wt}(y_0y_2 - y_1^2) = 2$.
The membership problem for non-differential ideals was treated in Section 2.3 of Part I. According to Theorem 2.5 we only have to compute the remainder of a polynomial w.r.t. a Gröbner basis of an ideal to decide membership. A MAPLE *session for doing this looks like follows:*

```
> with(grobner):
> vars:=[seq(y[i],i=0..5)];
           vars := [y[0], y[1], y[2], y[3], y[4], y[5]]
> P3:=[ y[0]*y[2]-y[1]^2, y[0]*y[3]-y[1]*y[2],
       y[0]*y[4]-y[2]^2, y[0]*y[5]+y[1]*y[4]-2*y[2]*y[3] ]:
> GB:=gbasis(P3,vars):
> f:=y[1]:
> g:=y[1]*y[3]-y[2]^2:
> normalf(f,GB,vars);
                              y[1]
> normalf(g,GB,vars);
                      y[1] y[3] - y[0] y[4]
> normalf(f*g,GB,vars);
                                0
```

Since the remainder is non-zero for f *and* g *we conclude that neither of these polynomials lies in* $[y_0y_2 - y_1^2]$. *On the other hand* $fg \in [y_0y_2 - y_1^2]$ *since*

the remainder in this case becomes zero. In the above MAPLE *session we have checked for membership in* P_3 *for all polynomials, without loss of generality since* $P_1 \subset P_2 \subset P_3$.
We have found an element $fg \in [\,p\,]$ *but* $f \notin [\,p\,]$ *and* $g \notin [\,p\,]$, *which implies that* $[\,p\,]$ *is not prime!*

Searching for generators of the input-output differential ideal $\Sigma^c \subset \Sigma$, the input-output polynomial of lowest degree p is a natural candidate. Above we have given an example of a system whose input-output polynomial p generates a differential ideal $[\,p\,]$, which is not prime.

The non-primeness of a differential ideal describing the input-output properties of a system implies that there are input-output relations not in the ideal but whose product belongs to the ideal. Hence, there may be non-unique solutions to the corresponding differential equations, since a solution only has to satisfy one of the relations. The "uniqueness of solutions" is often a desired property for models of physical systems.

With the approach in Section 6.3 we will be able to show that Σ and Σ^c, derived from classical state space descriptions, always are prime differential ideals. Hence, this property of Σ^c together with the observation in the above example shows there are systems for which the input-output ideal Σ^c cannot be generated by the input-output polynomial p of lowest degree only, i.e.,

$$[\,p\,] \subsetneq \Sigma^c$$

in general and we have to include more generators. Can we compute a finite set of generators for Σ^c at all?

6.2 Is the Input-Output Ideal Finitely Generated?

In this section we will show the somewhat surprising result that there are systems described by the finitely generated differential ideal Σ that have an input-output ideal Σ^c which is not (differentially) finitely generated. This means that we need infinitely many elements in the usual description of Σ^c as a differential ideal. In other words this particular differential ideal corresponds to infinitely many differential equations, which all are needed to describe the input-output behavior of the system.

As in the previous section we consider the system in Example 6.1

$$\dot{x}_1 = x_1 x_2, \quad \dot{x}_2 = 0, \quad y_0 = x_1$$

and its differential algebraic representation

$$\Sigma = [\,\dot{x}_1 - x_1 x_2, \dot{x}_2, y_0 - x_1\,].$$

In the rest of this section Σ^c denotes the input-output ideal for this specific system. If we differentiate the last generator w.r.t. time and substitute $\dot{x}_1$ and $\dot{x}_2$ by the first two equations or equivalently use the extended Lie-derivative operator we get $y_1 = x_1x_2$. Repeating this a number of times results in the expression $y_s = x_1x_2^s$. This expression can now be used to determine if a polynomial $f \in k\{y\}$ is a member of Σ^c. Since the elements in Σ^c is formed by elimination in the differential ideal Σ, which includes the elements $y_s - x_1x_2^s$ the substitution $y_s \mapsto x_1x_2^s$ in an element of Σ^c must yield zero.

We will now study how an arbitrary element of Σ^c can be constructed.

Definition 6.3 *A monomial* $M \in k[y_0, y_1, \ldots]$ *is said to be of type* (i, j)*, where* $i = \mathrm{tdeg}(M)$ *is the total degree and* $j = \mathrm{wt}(M)$ *is the weight of* M.
Notation (non-standard): $\mathrm{type}(M)$.

By the notation $\mathrm{tdeg}(M)$ we here mean the sum of the powers of all variables in M.

Example 6.4 *Let* $M = y_0y_3^2y_4$ *then*

$$\mathrm{tdeg}(M) = 4, \quad \mathrm{wt}(M) = 10 \quad \Leftrightarrow \quad \mathrm{type}(M) = (4, 10).$$

Lemma 6.3 *Let* $M \in k[y_0, y_1, \ldots]$ *be a monomial of* $\mathrm{type}(M) = (i, j)$*. Then* $M \mapsto x_1^i x_2^j$ *under the substitution rule* $y_s \mapsto x_1x_2^s$.

Proof *The monomial* M *can be written as* $M = y_{\mu_1}^{\nu_1} y_{\mu_2}^{\nu_2} \ldots y_{\mu_m}^{\nu_m}$ *and* $\mathrm{type}(M) = (i, j) = (\sum_{i=1}^m \nu_i, \sum_{i=1}^m \nu_i\mu_i)$*. We do the substitution* $y_s \mapsto x_1x_2^s$ *in* M *and get*

$$M \mapsto (x_1x_2^{\mu_1})^{\nu_1} \ldots (x_1x_2^{\mu_m})^{\nu_m} = x_1^{\sum_{i=1}^m \nu_i} x_2^{\sum_{i=1}^m \nu_i\mu_i} = x_1^i x_2^j,$$

which proves the lemma. □

The lemma tells us that monomials of the same type in $k[y_0, y_1, \ldots]$ will be reduced to the same monomial in $k[x_1, x_2]$ under the substitution $y_s \mapsto x_1x_2^s$.

Now let $f \in \Sigma^c$. A consequence of the discussion just before Definition 6.3 and of Lemma 6.3 is that the coefficients of monomials of same type in f have to sum up to zero.

Example 6.5 *Consider the following differential polynomial*

$$f = y_0y_2 - y_1^2 + y_0^2y_6 - \frac{1}{2}y_1y_2y_3 - \frac{1}{2}y_1^2y_4,$$

where the first two monomials are of type (2,2) and the three last monomials are of type (3,6). The substitution $y_s \mapsto x_1x_2^s$ *gives*

$$f = (1-1)x_1^2x_2^2 + (1 - \frac{1}{2} - \frac{1}{2})x_1^3x_2^6 = 0.$$

Hence, $f \in \Sigma^c$.

We now generalize these ideas. Every polynomial in $k[y_0, y_1, \ldots]$ can be divided into homogeneous polynomials, i.e., polynomials whose monomials are of the same total degree. Furthermore, every polynomial can be divided into isobaric polynomials. Combining these two observations implies that we can always obtain the following decomposition of a polynomial $P \in k[y_0, y_1, \ldots]$

$$P = \sum_{i=0}^{N} P_i = \sum_{i=0}^{N} \sum_{j=0}^{L_i} P_i^j,$$

where P_i^j is a homogeneous and isobaric polynomial, i.e., all its monomials are of type (i, j). N is the largest total degree of any monomial in P and L_i is the highest weight of any monomial in P_i. An explicit expression for P_i^j is

$$P_i^j = \sum_{s=1}^{S_{ij}} \alpha_s^{ij} M_s^{ij},$$

where M_s^{ij} are monomials of type (i, j), S_{ij} are the number of monomial in P_i^j and $\alpha_s^{ij} \in k$.

From the above discussion we know that $P \in \Sigma^c$ if and only if the coefficients of the monomials of P that are of the same type sums up to zero, i.e.,

$$\sum_{s=1}^{S_{ij}} \alpha_s^{ij} = 0, \quad \forall i, j.$$

Now concentrate on the pieces P_i^j in which we can divide P if $P \in \Sigma^c$. From which monomials can then P_i^j be constructed? We consider a number of different cases.

type$(M_s^{ij}) = (1, j)$ Then $M_s^{ij} = y_j$ and hence there is only one monomial of each type. Thus, the coefficient of the monomial of each type have to be zero if $P \in \Sigma^c$, i.e., P contains no monomials of type $(1, j)$.

type$(M_s^{ij}) = (2, 1)$ Then $M_s^{ij} = y_0 y_1$ and again this monomial cannot appear in a $P \in \Sigma^c$.

type$(M_s^{ij}) = (2, j)$ If $j \geq 2$ then we have the monomials $y_0 y_j, y_1 y_{j-1}, \ldots, y_{j/2}^2$ if j is even and $y_1 y_j, y_1 y_{j-1}, \ldots, y_{\frac{j-1}{2}} y_{\frac{j+1}{2}}$ if j is odd.

We consider the case when j is even. Then any $P_2^j \in \Sigma^c$ can be written

$$\begin{aligned} P_2^j = \sum_{s=0}^{j/2} \alpha_s y_s y_{j-s} &= \alpha_0 y_0 y_j + \alpha_1 y_1 y_{j-1} + \ldots + \alpha_{j/2} y_{j/2}^2 \\ &= -\alpha_1 (y_0 y_j - y_1 y_{j-1}) - \alpha_2 (y_0 y_j - y_2 y_{j-2}) - \ldots - \alpha_{j/2} (y_0 y_j - y_{j/2}^2) + \sum_{s=0}^{j/2} \alpha_s y_0 y_j \\ &= -\alpha_1 (y_0 y_j - y_1 y_{j-1}) - \alpha_2 (y_0 y_j - y_2 y_{j-2}) - \ldots - \alpha_{j/2} (y_0 y_j - y_{j/2}^2), \end{aligned}$$

where the last equality follows from the assumption $P_i^j \in \Sigma^c$, since the α_s then have to sum up to zero.

Hence we observe that P_2^j can be expressed as a linear combination of the isobaric polynomials $q_s^{2j} = y_0 y_j - y_s y_{j-s}$, i.e.,

$$P_2^j = \sum_{s=0}^{j/2} -\alpha_s q_s^{2j}, \quad q_s^{2j} = y_0 y_j - y_s y_{j-s}.$$

In the same way we construct isobaric polynomials $q_s^{3j}, q_s^{4j}, \ldots$ such that P_i^j can be written as a linear combination of q_s^{ij}. These polynomials are given by

$$q_s^{ij} = y_0^{i-1} y_j - y_{l_1} y_{l_2} \ldots y_{l_i}, \quad \sum_{r=1}^{i} l_r = j,$$

where y_{l_r} are written in ascending order.

Example 6.6 *Here we list* q_s^{35}.

$$y_0y_0y_5 - y_0y_1y_4,\ y_0y_0y_5 - y_0y_2y_3,\ y_0y_0y_5 - y_1y_1y_3,\ y_0y_0y_5 - y_1y_2y_2$$

The part of a polynomial, $P \in \Sigma^c$ *that consists of monomials of type* $(3, 5)$, *i.e.,* P_3^5 *can be written as a linear combination of these polynomials.*

We summarize the above discussion in the following theorem, which gives us a useful representation of the elements in Σ^c.

Theorem 6.2 *Let* P_i^j *be a linear combination of the isobaric polynomials*

$$q_s^{ij} = y_0^{i-1} y_j - y_{l_1} y_{l_2} \ldots y_{l_i}.$$

where the index s *is an enumeration of the index sets* $\{l_1, \ldots, l_i\}$ *satisfying* $\sum_{r=1}^{i} l_r = j$, $0 \le l_r \le j$. *Then*

$$P \in \Sigma^c \quad \Leftrightarrow \quad P = \sum_{i=2}^{N} \sum_{j=2}^{L_i} P_i^j,$$

where N *and* L_i *are defined as before.*

Note that the polynomials q_s^{ij} can be considered as a basis of a finite dimensional vector space with coefficients α_s^{ij} as coordinates.

In Chapter 3 we mentioned that differential ideals are not always finitely generated. Though, there are some ideals that are generated by an infinite set of polynomials but still can be generated by a finite number of polynomials.

Lemma 6.4 *Consider the following set of polynomials*

$$\mathcal{S} = \Big\{ f \in k\{y\} \,|\, f = \sum_{i=1}^{N} \alpha_i f_i,\ \alpha_i \in k,\ f_i \in k\{y\} \Big\}.$$

Then,

$$[\mathcal{S}] = [f_1, \ldots, f_N].$$

Proof *An element* h *in* $[\mathcal{S}]$ *can be written as*

$$h = \sum \beta_f^0 f + \sum \beta_f^1 f' + \ldots + \sum \beta_f^M f^{(M)}, \quad f \in \mathcal{S}, \quad \beta_f^j \in k\{y\},$$

where each sum only ranges over finitely many $f \in \mathcal{S}$ *since* h *has a finite number of terms. Using the representation of an element in* $\mathcal{S}$ *we can rewrite the above sum according to*

$$h = \sum_{i=1}^{N} \gamma_i^0 f_i + \sum_{i=1}^{N} \gamma_i^1 f_i' + \ldots + \sum_{i=1}^{N} \gamma_i^M f_i^{(M)},$$

i.e., by definition an element in $[f_1, \ldots, f_N]$*. Hence,* $[S] \subset [f_1, \ldots, f_N]$*.*

The other inclusion is trivial since $\{f_1, \ldots, f_N\} \subset [\mathcal{S}]$ *and the theorem is proved.*

□

The above lemma tells us that if we take the set of all linear combinations (over k) of a finite number of polynomials, then the differential ideal generated by this (infinite) set is equal to the ideal generated by the original finitely many polynomials.

We can now attack the question: Is Σ^c (differentially) finitely generated? The answer is negative despite the fact that Σ is generated by a finite number of polynomials (the state equations).

Theorem 6.3 *There are systems whose input-output ideal* Σ^c *is not (differentially) finitely generated.*

Proof *The proof consists of an example of a system whose input-output ideal is not finitely generated. We consider the following system and its corresponding input-output ideal*

$$\Sigma = [\dot{x}_1 - x_1 x_2,\ \dot{x}_2,\ y - x_1], \qquad \Sigma^c = \Sigma \cap k\{y\},$$

which is the same system as in Example 6.1.

We also introduce the notation

$$\Sigma^c_{\mathrm{wt} \le L} \triangleq \Big\{ P \in \Sigma^c \,|\, \textit{The monomials of } P \textit{ have weight} \le L \Big\}.$$

Observe that $\Sigma^c_{\mathrm{wt}\le L}$ *is not a differential ideal, since it is not closed under derivation.*

We will prove that Σ^c is not finitely generated by contradiction.

Suppose that Σ^c is finitely generated. This means that there are polynomials $g_1, \ldots, g_K \in \Sigma^c$ such that $\Sigma^c = [\, g_1, \ldots, g_K \,]$. Since K is finite there is a monomial in one of the generators which has higher or equal weight than all the other monomials of the generators. This implies that $\{g_1, \ldots, g_K\} \in \Sigma^c_{wt \le L}$ if L is chosen big enough. Fix such an L.

Now,

$$\{g_1, \ldots, g_K\} \subset \Sigma^c_{wt \le L} \quad \Rightarrow \quad [\, \Sigma^c_{wt \le L} \,] = \Sigma^c.$$

If we can show that there exists a polynomial in Σ^c that is not in $[\, \Sigma^c_{wt \le L} \,]$ we have the sought contradiction. To succeed with this task we utilize the special representation of the polynomials in Σ^c derived in Theorem 6.2.

We claim that

$$f_{L+1} = y_{\frac{L-1}{2}} y_{\frac{L+3}{2}} - y^2_{\frac{L+1}{2}} \in \Sigma^c \quad \textit{but} \quad f_{L+1} \notin [\, \Sigma^c_{wt \le L} \,],$$

where L is an odd number, i.e., f_{L+1} is the kind of polynomial we are searching for.

That $f_{L+1} \in \Sigma^c$ holds is easy to see since f_{L+1} maps to zero under the substitution $y_s \mapsto x_1 x_2^s$.

By Theorem 6.2 we know that an element in $\Sigma^c_{wt \le L}$ always can be written on the form $P = \sum_{r=1}^{M} \gamma_r \tilde{f}_r$ since

$$P = \sum_{i=2}^{N} \sum_{j=2}^{L_i} P_i^j = \sum_{i=2}^{N} \sum_{j=2}^{L_i} \sum_{s=1}^{S_{ij}} \alpha_s^{ij} q_s^{ij} = \sum_{r=1}^{M} \gamma_r \tilde{f}_r,$$

where γ_r and $\tilde{f}_r$ are new names for all the α_s^{ij} and q_s^{ij}, respectively.

According to Lemma 6.4 we have $[\, \Sigma^c_{wt \le L} \,] = [\, f_1, \ldots, f_M \,]$, i.e., $[\, \Sigma^c_{wt \le L} \,]$ is finitely generated by the isobaric polynomials $f_r = q_s^{ij}$ where $\mathrm{wt}(f_r) \le \max_i(L_i)$.

Hence,

$$\begin{aligned}[\, \Sigma^c_{wt \le L} \,] = [\, & y_0 y_2 - y_1^2,\ y_0 y_3 - y_1 y_2, \ldots, \\ & y_0 y_L - y_1 y_{L-1}, \ldots, y_0 y_L - y_{\frac{L-1}{2}} y_{\frac{L+1}{2}}, \\ & y_0^2 y_2 - y_0 y_1^2,\ y_0^2 y_3 - y_0 y_1 y_2,\ y_0 y_3 - y_1^3, \ldots \,],\end{aligned}$$

where the last dots denote a finite number of generators of weight less than or equal to L.

We observe that the total degree of a generator remains constant under derivation. This means that if $f_{L+1} \in [\, \Sigma^c_{wt \le L} \,]$ then it has to be constructed of generators of total degree two and their derivatives.

Since f_{L+1} is an isobaric polynomial of weight $L+1$ and all the generators are isobaric, f_{L+1} has to be constructed of generators of weight $L+1$. The only polynomials *which can be used to construct* f_{L+1} *are the first derivative of the generators of weight* L*, i.e.,*

$$\frac{d}{dt}(y_0 y_L - y_j y_{L-j}) = y_0 y_{L+1} + y_1 y_L - y_j y_{L-j+1} - y_{j+1} y_{L-j},$$

where $j = 1, 2, \ldots, \frac{L-1}{2}$ *and* L *is odd. The polynomial* f_{L+1} *has to be a linear combination over* k *of these derivatives since this is the only construction that keeps the weight and total degree constant. We now try to express* f_{L+1} *as a linear combination of these derivatives, i.e.,*

$$\begin{aligned} y_{\frac{L-1}{2}} y_{\frac{L+3}{2}} - y^2_{\frac{L+1}{2}} = \gamma_1 &(y_0 y_{L+1} + y_1 y_L - y_1 y_L - y_2 y_{L-1}) \\ &+ \gamma_2 (y_0 y_{L+1} + y_1 y_L - y_2 y_{L-1} - y_3 y_{L-2}) \\ &\quad + \gamma_3 (y_0 y_{L+1} + y_1 y_L - y_3 y_{L-2} - y_4 y_{L-3}) + \ldots \\ &\qquad + \gamma_{\frac{L-1}{2}} (y_0 y_{L+1} + y_1 y_L - y_{\frac{L-1}{2}} y_{\frac{L+3}{2}} - y^2_{\frac{L+1}{2}}) \end{aligned}$$

We now introduce the notation

$$z_0 = y_0 y_{L+1} + y_1 y_L,\ z_1 = y_1 y_L,\ z_2 = y_2 y_{L-1}, \ldots, z^2_{\frac{L+1}{2}} = q y^2_{\frac{L+1}{2}}$$

to simplify solving for γ_i *in the above expression. These new variables are linearly independent over* k*. Consider the resulting system*

$$\begin{aligned} z_{\frac{L-1}{2}} - z_{\frac{L+1}{2}} = \gamma_1 &(z_0 - z_1 - z_2) \\ &+ \gamma_2 (z_0 - z_2 - z_3) + \ldots \\ &\quad + \gamma_{\frac{L-1}{2}} (z_0 - z_{\frac{L-1}{2}} - z_{\frac{L+1}{2}}) \end{aligned}$$

Since z_1 *appears only in the first parenthesis and not on the left hand side of the expression,* $\gamma_1 = 0$*. If* $\gamma_1 = 0$*, then* z_2 *appears only in the second parenthesis hence* $\gamma_2 = 0$*. Continuing in this way we get* $\gamma_1 = \ldots = \gamma_{\frac{L-3}{2}} = 0$ *and* $\gamma_{\frac{L-1}{2}}$ *can not be determined to get a valid equality.*

Hence, $f_{L+1} \notin [\Sigma^c_{wt \le L}]$ *and we have a contradiction. This proves the theorem.* □

The conclusions to be drawn from the results in this section is that the representation of the input-output ideal is a non-trivial question. Since some systems do not have a finite set of generators for their input-output ideals the input-output behavior cannot explicitely be represented on this form in, e.g., a computer program.

Compare with the situation in commutative algebra where the ideals behave much nicer. Then all ideals are finitely generated, even those that are constructed by intersection, see Theorem 2.7.

6.3 A Non-Differential Approach

To be able to apply some non-differential algebraic results, such as Gröbner bases to state space differential ideals in differential rings, we have to develop some preliminary results concerning rings of infinitely many variables.

As pointed out in Chapter 3 a differential ring can be considered as a ring in infinitely many variables. In order to use non-differential tools it is useful to consider the subset of a differential ideal which consists of all elements up to a specific differential order. This subset can be seen as an ideal in a non-differential ring and will be called a *truncation* of the differential ideal. To compute truncations we introduce the concept of graph ideals.

Graph Ideals

By a *graph* of a mapping one usually means the set of points defined by the expression $y = f(x),\ x \in \mathbb{R}$ or equivalently $\{ (x, y) \mid y - f(x) = 0,\ x \in \mathbb{R} \}$. If we allow more variables and equations we can generalize the meaning of a graph to be the point set $\{ (x, z_1, \ldots, z_j) \mid z_1 - f_1(x) = 0, \ldots, z_j - f_j(x) = 0,\ x \in \mathbb{R}^n \}$. In Chapter 2 we saw that there is a correspondence between a set of polynomial equations and the ideal generated by the left hand sides of these equations. This motivates the name of the kind of ideal we define now.

Definition 6.4 *A graph ideal is an ideal* I *of the form*

$$I = \langle z_1 - f_1, \ldots, z_s - f_s \rangle,$$

where z_i *are variables and* $f_i \in k[x_1, \ldots, x_n]$.

The aim now is to study relations of dependency between $f_1, \ldots, f_s$, which are supposed to be known polynomials. By dependency we mean *algebraic dependency*. The polynomials $f_1, \ldots, f_m \in k[x_1, \ldots, x_n]$ are said to be *algebraically dependent* if there is a non-zero polynomial $P \in k[w_1, \ldots, w_m]$ such that $P(f_1, \ldots, f_m) = 0$.

We recall Theorem 2.7, which tells us how to eliminate the x_i. If we compute a Gröbner basis of the graph ideal w.r.t. the ordering

$$x_1 > \ldots > x_n > z_1 > \ldots > z_s,$$

then we immediately get a generating set for the ideal $I \cap k[z_1, \ldots, z_s]$. From the discussion of ideals we know that this ideal is the subset of all polynomials in the variables $z_1, \ldots, z_s$ only, which can be "formed" by the polynomials that generate I.

Example 6.7 *Consider the graph ideal*

$$I = \langle z_1 - x_1 + x_2, z_2 - (x_1 - x_2)^2 + x_1 - x_2 \rangle,$$

where $f_1 = x_1 - x_2$ *and* $f_2 = x_1^2 - 2x_1x_2 + x_2^2 - x_1 + x_2$. *If we calculate a Gröbner basis for* I *w.r.t. the ordering* $x_2 > x_1 > z_2 > z_1$ *we get*

$$\mathrm{GB}(I) = \{z_1 - x_1 + x_2, z_2 - z_1^2 + z_1\}.$$

According to Theorem 2.7, $\{z_2 - z_1^2 + z_1\}$ *is a Gröbner basis of* $I \cap \mathbb{Q}[z_1, z_2]$, *that is* $I \cap \mathbb{Q}[z_1, z_2] = \langle z_2 - z_1^2 + z_1 \rangle$. *Hence, we have the dependency relation*

$$f_2 - f_1^2 + f_1 = 0$$

between f_1 *and* f_2.

We will now consider chains of graph ideals and will therefore need a notation for rings of variable size. Let

$$R_j \triangleq k[x_1, \ldots, x_n, z_1, \ldots, z_j],$$

(where n is fixed) and

$$G_j \triangleq \{z_1 - f_1, \ldots, z_j - f_j\},$$

where as before z_i are variables and $f_1, \ldots, f_j \in k[x_1, \ldots, x_n]$.

Definition 6.5 *Let* $I_{j,N}$ *be the graph ideal generated by* G_j *in* R_N, *i.e.,*

$$I_{j,N} = \langle z_1 - f_1, \ldots, z_j - f_j \rangle = \Big\{ f \in R_N \,|\, f = \sum_{\substack{\alpha \in R_N \\ p \in G_j}} \alpha p \Big\}.$$

The ideals $I_{j,N}$, $1 \le j \le N$ form a strictly ascending chain of ideals

$$I_{1,N} \subset I_{2,N} \subset \ldots \subset I_{N,N},$$

since a new variable is introduced in each new ideal of the chain.

Truncations of Graph Ideals

The concept of graph ideals is very useful when considering differential ideals like (5.2), i.e., differential ideals generated by the state space equations. These differential ideals can be considered as graph ideals of a ring in infinitely many variables. First we present a preliminary lemma.

Lemma 6.5 *Let* $p \in R_N$ *and* $f_1, \ldots, f_j \in k[x_1, \ldots, x_n]$, *then the substitution* $z_i \mapsto f_i, 1 \le i \le j$ *maps* $p \mapsto 0$ *if and only if* $p \in I_{j,N}$, *i.e.,*

$$z_i \mapsto f_i \;\Rightarrow\; p \mapsto 0 \qquad \Leftrightarrow \qquad p \in I_{j,N}.$$

Proof *Consider a polynomial* $h \in k[x]$ *in one variable. Using the binomial theorem it is easy to see that*

$$h(x + c) = h(c) + g(x, c)\,x = h(c) + q(x),$$

for some polynomial $g \in k[x]$ *or* $q \in \langle x \rangle$*. This is easily generalized to the multivariate case, i.e., if* $h \in k[w_1, \ldots, w_m]$ *then*

$$h(w + c) = h(c) + \sum_{i=1}^{m} g_i(w, c)\, w_i,$$

where $h(c)$ *do not contain any* w_i*. Hence*

$$h(w + c) \in \langle w_1, \ldots, w_m \rangle \quad \Leftrightarrow \quad h(c) = 0.$$

Now, take a polynomial $p \in R_N$ *and observe that the identity* $z_i = (z_i - f_i) + f_i$ *is an expression of the above form. Suppressing the* $x_1, \ldots, x_n$ *dependence we can rewrite* p *as follows:*

$$\begin{aligned} p(z_1, \ldots, z_N) &= p((z_1 - f_1) + f_1, \ldots, (z_j - f_j) + f_j, z_{j+1}, \ldots, z_N) \\ &= p(f_1, \ldots, f_j, z_{j+1}, \ldots, z_N) + q(z_1, \ldots, z_N), \end{aligned}$$

where $q \in \langle z_1 - f_1, \ldots, z_j - f_j \rangle \subset R_N$*. As above we have*

$$p(z_1, \ldots, z_N) \in \langle z_1 - f_1, \ldots, z_j - f_j \rangle \quad \Leftrightarrow \quad p(f_1, \ldots, f_j, z_{j+1}, \ldots, z_N) = 0$$

and the lemma follows. □

The following lemma applied to graph ideals in infinitely many variables projects the problems onto problems in a finite number of variables. Hence tools from commutative algebra as Gröbner bases can be utilized.

Lemma 6.6 *Let* $I_{j,N}$ *be the graph ideal defined in Definition 6.5, then*

(i) $I_{j,N}$ *is a prime ideal.*

(ii) $I_{j,N} \cap k[x_1, \ldots, x_n] = \{0\}$.

(iii) For all j, s, N *where* $j \leq s \leq N$ *we have that*

$$I_{N,N} \cap k[z_1, \ldots, z_j] = I_{s,N} \cap k[z_1, \ldots, z_j].$$

Proof *We start by showing that* $I_{j,N}$ *is a prime ideal.*

Let $p \in R_N$ *and* $\tilde{p}$ *denote that the substitution* $z_i \mapsto f_i$, $1 \leq i \leq j$ *has been performed. According to Lemma 6.5 we have*

$$p \in I_{j,N} \quad \Leftrightarrow \quad \tilde{p} = 0.$$

Now, suppose that $f\,g \in I_{j,N}$. *Then this is equivalent to*

$$\tilde{f}\,\tilde{g} = 0.$$

Both $\tilde{f}$ *and* $\tilde{g}$ *are polynomials in the ring* $k[x_1, \dots, x_n, z_{j+1}, \dots, z_N]$, *since* $f_i \in k[x_1, \dots, x_n]$. *The product of two polynomials cannot be zero unless one of them is zero. This forces at least one of* $\tilde{f}$ *or* $\tilde{g}$ *to be zero, i.e., at least one of* f *or* g *belongs to* $I_{j,N}$ *according to Lemma 6.5. Hence, we have that*

$$f\,g \in I_{j,N} \quad \Rightarrow \quad f \in I_{j,N} \ \vee \ g \in I_{j,N},$$

which is the definition of that $I_{j,N}$ *is prime and we have shown (i).*

Suppose that $p \in I_{j,N}$. *We know that this is equivalent to* $\tilde{p} = 0$. *On the other hand if* $p \in k[x_1, \dots, x_n]$ *then* $p = \tilde{p}$. *Hence if* $p \in I_{j,N} \cap k[x_1, \dots, x_n]$ *then* $p = 0$ *and (ii) follows.*

To show (iii) we observe that $I_{s,N} \subset I_{N,N}$, *since* $I_{k,N}$, $1 \le k \le N$ *is an ascending chain of ideals. Hence*

$$I_{s,N} \cap k[z_1, \dots, z_j] \subset I_{N,N} \cap k[z_1, \dots, z_j].$$

Now suppose $p \in I_{N,N}$. *Then* $p(x_1, \dots, x_n, f_1, \dots, f_N) = 0$ *according to Lemma 6.5. If in addition* $p \in k[z_1, \dots, z_j]$ *then*

$$p(x_1, \dots, x_n, f_1, \dots, f_j, z_{j+1}, \dots, z_N) = p(x_1, \dots, x_n, f_1, \dots, f_N)$$

since p *is free of* $z_{j+1}, \dots, z_N$. *Hence* $p(x_1, \dots, x_n, f_1, \dots, f_j, z_{j+1}, \dots, z_N) = 0$, *i.e.,* $p \in I_{j,N} \subset I_{s,n}$. *This shows that*

$$I_{s,N} \cap k[z_1, \dots, z_j] \supseteq I_{N,N} \cap k[z_1, \dots, z_j]$$

and the lemma is proved.

□

In words (ii) means that there is no polynomial in any graph ideal in the variables $x_1, \dots, x_n$ only. Number (iii) is even more useful. It tells us that if we are interested in relations (i.e., polynomial equations) between $z_1, \dots, z_j$ or equivalently between $f_1, \dots, f_j$ we do not have to care about $z_{s+1}, \dots, z_N$ or equivalently $f_{s+1}, \dots, f_N$, where $s \ge j$.

The Relation Between Σ^c and $\mathcal{O}_j$

How much information does one get about the truncated differential ideal $\Sigma^c \cap k[y_0, \dots, y_j, \bar{u}]$ by considering truncations of the graph ideal $\mathcal{O}_j$ which was introduced in Section 5.2?

Theorem 6.4 $\Sigma^c \cap k[y_0, \dots, y_j] = \mathcal{O}_j \cap k[y_0, \dots, y_j]$

Proof *By changing our point of view we know that we can consider a differential ring as a ring of infinitely many variables, i.e.,*

$$k\{x_1, \dots, x_n, y\} \to k[x_1, \dots, x_n, y, \dot{x}_1, \dots, \dot{x}_n, \dot{y}, \dots],$$

where the superscripts together with the subscripts are now nothing more than labels for the variables. A finitely generated differential ideal as $[A, B, C]$ *then becomes* $\langle A, \dot{A}, \dots, B, \dot{B}, \dots, C, \dot{C}, \dots \rangle$, *i.e., an ideal generated by an infinite number of generators.*

Consider the state space differential ideal $\Sigma = [\dot{x}_1 - f_1, \dots, \dot{x}_n - f_n, y - h]$. *If we now rename the variables according to*

$$\begin{aligned} z_0 &= y_0, \dots, z_j = y_j, \\ z_{j+1} &= \dot{x}_1, \dots, z_{j+n} = \dot{x}_n, \qquad z_{j+1+n} = y_{j+1}, \\ z_{j+2+n} &= \ddot{x}_1, \dots, z_{j+1+2n} = \ddot{x}_n, \quad z_{j+2+2n} = y_{j+2}, \dots \end{aligned}$$

we get the ideal

$$\langle z_0 - h_0, \dots, z_j - h_j, z_{j+1} - f_1, \dots, z_{j+n} - f_n, \dots \rangle$$

This is a graph ideal in infinitely many z_i *and* $x_1, \dots, x_n$. *Furthermore* h_i *is defined as in Definition 5.4. Since the proofs of Lemma 6.5 and 6.6 are completely independent of* N, *they hold even for* $N = \infty$.

With the non-differential description of Σ *it now follows from (iii) of Lemma 6.6 that*

$$\langle z_0 - h_0, \dots, z_j - h_j, z_{j+1} - f_1, \dots \rangle \cap k[z_0, \dots, z_j] = \langle z_0 - h_0, \dots, z_j - h_j \rangle \cap k[z_0, \dots, z_j]$$

or

$$\Sigma \cap k[y_0, \dots, y_j] = \langle y_0 - h_0, \dots, y_j - h_j \rangle \cap k[y_0, \dots, y_j]$$

that is

$$\Sigma \cap k[y_0, \dots, y_j] = \mathcal{O}_j \cap k[y_0, \dots, y_j]$$

which completes the proof. □

Observe that the input u does not appear anywhere in the proof. The theorem holds equally well if we adjoin u and its derivatives to k, i.e., if we let $k\langle u \rangle$ be the field of coefficients instead of k, see Definition 3.5.

The above theorem tells us that all elements in Σ^c up to order j are the same as those elements in $\mathcal{O}_j$ which contain only $y_i, 0 \le i \le j$. The main advantage is that $\mathcal{O}_j$ is a non-differential ideal which can be treated with methods from commutative algebra presented in preceding chapters.

Observe the interpretation of Σ as a graph ideal in the above proof. This tells us that Σ is prime, a property we utilized to show that $\Sigma^c \neq [p]$ in Section 6.1.

6.4 Description of Σ^c with Characteristic Sets

In Section 5.2 we showed that if p is a polynomial of lowest order in Σ^c then in general $\Sigma^c \neq [\,p\,]$. An interesting question now is: Are there cases when we have equality and in that case which conditions do we then have to pose on p?

Consider a system described by state space equations of the form (5.1) but without an input, i.e., $u = 0$. These equations correspond to the differential ideal

$$\Sigma \;=\; [\dot{x}_1 - f_1(x), \,\ldots\,, \dot{x}_n - f_n(x),\, y - h(x)] \;\subset\; k\{x_1, \,\ldots\,, x_n, y\} \tag{6.2}$$

As before we want to know which differential equations the output y satisfies, i.e., we are interested in $\Sigma^c = \Sigma \cap k\{y\}$. We are especially interested in the differential equation of lowest order $p = 0$ that y satisfies.

The differential polynomial p can be found using the method in [28] reviewed in Section 5.2. The polynomial p has some interesting properties. Since $p \in \Sigma$ is the polynomial of lowest order in the single variable y in Σ it is also of lowest order in Σ^c. Now Σ^c is a set of differential polynomials in one variable and since p is a polynomial of lowest order it has to be a characteristic set of Σ^c (two differential polynomials in one variable cannot be reduced w.r.t. each other so the characteristic set has only one element).

Theorem 6.5 *If* $\mathbf{A}$ *is a characteristic set of a prime differential ideal* I *then*

$$I = [\,\mathbf{A}\,] : H_{\mathbf{A}}^{\infty}.$$

Proof *See [50].* □

From the preceding sections we know that Σ can be interpreted as a graph ideal and hence is a prime differential ideal. That Σ is prime implies that Σ^c is prime. Furthermore, we have a characteristic set $\{p\}$ of the prime differential ideal Σ^c and consequently the theorem gives

$$\Sigma^c \;=\; [\,p\,] : H_p^{\infty} \tag{6.3}$$

We notice that since $p \in \Sigma^c$ we have

$$[\,p\,] \subset \Sigma^c \tag{6.4}$$

and from the expressions (6.3) and (6.4) we get the following limits on Σ^c:

$$[\,p\,] \;\subset\; \Sigma^c \;=\; [\,p\,] : H_p^{\infty} \tag{6.5}$$

We will now examine the relationship between $[\,p\,]$ and the differential ideal $[\,p\,] : H_p^{\infty}$.

If $f \in [\,p\,]$ then $f = f\,H_p^0 \in [\,p\,]$ which shows that

$$[\,p\,] \;\subset\; [\,p\,] : H_p^{\infty} \tag{6.6}$$

Suppose that $[p]$ is prime and that $\{p\}$ is a characteristic set of $[p]$. Let $f \in [p] : H_p^\infty$, then $fH_p^r \in [p]$ for some r. Since $[p]$ is prime either f or H_p^r has to be in $[p]$. According to the definition of H_p it has lower order than p. Since $\{p\}$ is a characteristic set of $[p]$ there are no element of lower order than p in $[p]$, hence $H_p \notin [p]$. The product $H_p^2 = H_p H_p \notin [p]$, since $[p]$ is prime and none of the factors of H_p^2 are in $[p]$. By induction one realizes that $H_p^r \notin [p]$. Thus we have a product $fH_p^r \in [p]$, where the factor $H_p^r \notin [p]$. Since $[p]$ is prime then f has to be in $[p]$. Thus we have showed that if p is the characteristic set of the prime differential ideal $[p]$ then

$$[p] : H_p^\infty \subset [p] \tag{6.7}$$

Combining the inclusions (6.6) and (6.7) gives the equality

$$[p] = [p] : H_p^\infty, \tag{6.8}$$

where p is the characteristic set of the prime differential ideal $[p]$. This discussion is in fact a proof of Theorem 6.5 in the case when the characteristic set consists of only one element.

We now observe the fact that since $\{p\}$ is a characteristic set of Σ^c it is also a characteristic set of $[p]$, since $[p] \subset \Sigma^c$.

Notice that if $\Sigma^c = [p]$ then $[p]$ have to be prime since Σ^c is prime. The discussion above is summed up in the following theorem:

Theorem 6.6 *If $p \in k\{y\}$ is the polynomial of lowest order in Σ^c then*

$$[p] \textit{ is prime} \quad \Leftrightarrow \quad \Sigma^c = [p].$$

Despite this characterization of Σ^c the theorem can be of limited use, since it can be a very difficult task to decide if $[p]$ is prime or not. However, there is one case when it is easy, namely if the initial I_p and separant S_p of p belong to k. Then $H_p \in k$ and $[p] = [p] : H_p$, which is prime.

We also note that Σ^c can be finitely generated without being equal to $[p]$.

Example 6.8 *The differential ideal $[{y_1}^2 - y_0]$ has a simple interpretation in mechanics and it has also been discussed by Pommaret [70]. Differentiating the generator once we obtain the differential equation $2y_1y_2 - y_1 = 0$ which has two solutions $y_1 = 0$ and $y_2 = \frac{1}{2}$. The last solution corresponds to the following state equations*

$$\dot{x} = \tfrac{1}{2}, \quad y = x^2, \tag{6.9}$$

where x could be the velocity of an object and y its kinetic energy. One can show that the (input-)output ideal Σ^c of (6.9) is

$$[{y_1}^2 - y_0] : y_1^\infty = [{y_1}^2 - y_0, 2y_2 - 1].$$

6.5 Conclusions and Extensions

Conclusions

We have considered systems described by polynomial differential equations, i.e.,

$$\begin{aligned} \dot{x}_1 - f_1(x, u) &= 0 \\ &\vdots \\ \dot{x}_n - f_n(x, u) &= 0 \\ y - h(x, u) &= 0, \end{aligned} \tag{6.10}$$

where $x = x_1, \ldots, x_n$ are the state variables, u is the input, y is the output and $f_1, \ldots, f_n$ are polynomials. In the language of differential algebra we can formulate this system description as a differential ideal generated by the left hand sides of (6.10), i.e.,

$$\Sigma = [\, \dot{x}_1 - f_1, \ldots, \dot{x}_n - f_n, y - h \,].$$

We have studied the set $\Sigma^c = \Sigma \cap k\{u, y\}$, i.e., all input-output differential equations that the systems input and output variables have to satisfy. In particular we have considered the case of no input, that is $\Sigma \subset k\{x, y\}$ and $\Sigma^c \subset k\{y\}$, which corresponds to the transient behavior of the system.

There is a method for obtaining the unique input-output equation of lowest order of the system represented by (6.10). Denoting the corresponding differential polynomial by p the results can be summarized as follows:

(i) $\Sigma^c \neq [\, p \,]$ in general.

(ii) Σ^c might not even be generated by a finite number of differential polynomials.

(iii) $\Sigma^c = [\, p \,] : H_p^\infty$ in general and $\Sigma^c = [\, p \,]$ if and only if $[\, p \,]$ is prime.

(iv) Σ^c is always prime but $[\, p \,]$ does not have to be prime.

These results were mainly obtained by considering a parameterized version of a first order system, namely

$$\dot{x}_1 - x_1 x_2 = 0, \quad \dot{x}_2 = 0, \quad y - x_1 = 0 \tag{6.11}$$

so they are not anomalies from some non-physical system.

These results show that despite the fact that the structure of the original system of polynomial differential equations is captured in the finitely generated differential ideal Σ we do not always have a simple differential algebraic description of the input-output ideal Σ^c. This means that one has to be careful in the study of such ideals and that the ideal description problem for them is highly non-trivial. It is also an indication that it could be more convenient to consider a system as described by solutions (time-trajectories) rather than equations. This is one of the ideas in the behavioral framework of Willems, see e.g., [83]. There is also a close analogy with the duality between algebraic geometry and commutative algebra, i.e., zero manifolds vs. equations, as in the spirit of [63].

Extensions

The results were obtained by examining a system written in polynomial state space form without an input. It would be interesting to examine the possibility to construct an example with nonzero input, which has similar consequences as the one treated. The input-output differential ideal $\Sigma^c \in k\{u, y\}$ is then a differential ideal in two variables.

How “badly” can the ideal $[p]$ behave? Are there examples of systems, where $[p]$ is not even a radical differential ideal? If this is the case, then there are elements in the ideal with the property $f \notin [p]$ but $f^r \in [p]$ for some r. Translated into differential equations this means that $[p]$ only contains the differential equation $f^r = 0, r > 1$ and not the simpler description $f = 0$. A candidate of a non-radical ideal is $P = [y_1{}^2 - y_0]$. It can be shown that $y_3{}^2 \in P$. To show that P is not radical, one have to show that $y_3 \notin P$.

The differential ideal $[y_0 y_2 - y_1]$ considered as an ideal in infinitely many variables has some similarities with a graph ideal. Is it prime? If this is the case we have an example of a prime differential ideal generated by a single differential polynomial whose initial and separant are nontrivial.

To use the method of Gröbner bases in differential algebraic calculations we had to interpret the differential algebraic problem in algebraic terms. This interpretation is possible due to the fact that we found a restriction of the problem to a ring of finitely many variables. In the general case this is not trivial or even possible, since there are differential ideals, which is not (differentially) finitely generated as was pointed out in Chapter 3. An interesting subject is the concept of differential Gröbner bases. Some references on this topic are [26, 58, 59, 66].

Part III

Quantifier Elimination in Control Theory

7

Introduction

Symbolic computation, also known as computer algebra, has been recognized by an increasing number of people as a useful tool in science and engineering. For control and design of linear and especially non-linear systems there exist toolboxes, user developed packages and algorithms of various sophistication, implemented in computer algebra programs such as MAPLE and MATHEMATICA [14, 15, 84].

A basic ingredient of many algorithms is some kind of symbolic solver, i.e., routines that manipulate systems of equations (even differential ones) to get exact solutions or simplified expressions for some desired subset of the variables. Gröbner bases, resultants and Ritt's algorithm which was treated in Part I of the thesis facilitate a systematic approach to these problems.

A systematic treatment of real polynomial systems, i.e., systems of real polynomial equations and inequalities has not yet been implemented as a standard feature in any of the major computer algebra systems. The main reason for this is the algorithmic complexity of such solvers which is a subject of intensive research [5, 40, 71]. However, there are solvers under construction, e.g., in [77] an implementation in MATHEMATICA of a solver for systems of polynomial strict inequalities is described.

It turns out that neither Gröbner bases nor resultants alone are suited to be generalized to solve real polynomial systems. The main reason is that none of the methods keep track of the signs of the involved polynomials. Cylindrical algebraic decomposition and quantifier elimination are more general algorithms whose intermediate steps rely heavily on resultant computations and sign determination as we saw in Part I of the thesis.

In control theory, one of the first attempts to apply quantifier elimination techniques was made by Anderson et al. [2] in the mid seventies. However, the algorithmic techniques at that time were very complex and no appropriate computer software was available. Recently, a few papers treating control related problems have appeared [8, 78] and since the seventies there has been considerable progress in the development of more effective quantifier elimination algorithms starting with Collins [18]. For an extensive bibliography on quantifier elimination see [3] and more recent results [13, 42, 43, 61].

In the control community there is a growing interest to use inequalities in modeling and control of dynamical systems, see e.g., [82]. In optimal control it is very common to have inequality constraints both on the control and system variables, see [11, 53]. However, the existence of algorithms for symbolic computation with systems of polynomial equations and inequalities have still not yet been fully recognized. The author believes that there will be an increased interest for this kind of algorithms in the control community since many problems, especially in non-linear control theory, can be formulated as real polynomial systems combined with logical operations. In this part of the thesis we will formulate some control related problems in this framework to demonstrate the power of the combination of algebra and logic that quantifier elimination in the first-order theory of real closed fields constitutes.

This part of the thesis is organized as follows. In Chapter 8 a number of applications of quantifier elimination to linear systems are discussed. Problems regarding stabilization and feedback design are treated. In Chapter 9 some problems in nonlinear control are addressed such as a constructive method for computing stationary points of a system with amplitude constraints on the control. We also consider reachability questions from a constructive perspective. Finally we summarize the results and list some possible extensions in Chapter 10.

8

Applications to Linear Systems

To investigate the usefulness of quantifier elimination in linear control theory we have chosen two types of problems. Stabilization, where the objective is to choose parameters in a controller such that the closed-loop system becomes stable; and the more general problem of feedback design, where the objective is not only to stabilize the system but also to fulfill a number of specifications. The methods are combinations of a variant of the Routh criterion and Nyquist techniques together with quantifier elimination. We also use quantifier elimination to investigate stability of feedback systems including static nonlinearities.

We only treat SISO-systems but many of the ideas have straight forward generalizations to MIMO-systems. However, the computational complexity of these algorithms grows quickly with the number of variables, which make them less attractive for larger examples.

In Section 8.1 stabilization problems are treated. An algebraic approach to feedback design is given in Section 8.2 where we also use the Circle criterion to compute bounds on static nonlinearities ensuring closed-loop stability.

8.1 Stabilization

To investigate stability of linear systems, inequality criteria by Routh [73], Hurwitz [44] et al. have been extensively used. These stability criteria give conditions on the coefficients of the characteristic equation of a linear time invariant differential equation in terms of a number of inequalities.

A differential equation which results from modeling or design of a controller for a linear system often includes parameters (e.g., physical design constants and/or parameters of a controller) to be determined so that the system becomes stable. If the coefficients of the differential equation are rational functions of those parameters the Routh criterion is equivalent to a system of inequalities or possibly a union of such systems.

In robust control a common objective is to design a controller which stabilizes a system even though some parameters are only known to belong to certain intervals. These uncertainties impose additional constraints on the solution to the stabilization problem but can equally well be treated with the method presented in this section.

Stability Conditions

It is well known that stability of a linear time invariant system is determined by its poles. Stable systems have pole polynomials whose zeros all have strictly negative real parts and such polynomials have been given a special name.

Definition 8.1 *A polynomial is said to be Hurwitz if all its zeros have strictly negative real parts.*

The criterion by Routh states that a polynomial is Hurwitz if and only if its coefficients satisfy a system of polynomial inequalities. There are many equivalent criteria and here we choose to present the Lienard-Chipart criterion, which has the advantages over the Routh criterion that it involves only about half the number of determinantal inequalities and hence gives a system of inequalities of lower order.

Theorem 8.1 *Let* $f(s) = a_0 s^n + a_1 s^{n-1} + \ldots + a_{n-1}s + a_n$, $a_0 > 0$ *be a given polynomial with real coefficients. Define the Hurwitz determinant of order* $1 \leq i \leq n$ *as*

$$D_i = \begin{vmatrix} a_1 & a_3 & a_5 & \ldots & & \\ a_0 & a_2 & a_4 & \ldots & & \\ 0 & a_1 & a_3 & \ldots & & \\ 0 & a_0 & a_2 & a_4 & & \\ \vdots & \vdots & \vdots & \vdots & \ddots & \\ & & & & & a_i \end{vmatrix}, \qquad a_k = 0 \text{ for } k > n.$$

Then f *is a Hurwitz polynomial if and only if*

$$a_n > 0,\ a_{n-2} > 0,\ a_{n-4} > 0,\ \ldots; \quad D_2 > 0,\ D_4 > 0,\ D_6 > 0,\ \ldots$$

Proof *See Parks and Hahn [68] or Gantmacher [30].* □

To illustrate the use of quantifier elimination in stability analysis of linear systems we apply this theorem in some examples. In all examples the program QEPCAD [19] has been used to perform quantifier elimination.

Example 8.1 *Consider the unstable linear system*

$$G(s) = \frac{4}{s^2 - 2s + 2},$$

which has poles in $1 \pm i$. *We use a phase-lead compensator*

$$F(s) = N\frac{s + b}{s + Nb}$$

to stabilize the system, where $b > 0$ *and* $1 < N < 10$, *e.g., due to physical limitations in the implementation of the controller. The resulting closed-loop system becomes*

$$G_c(s) = \frac{GF}{1 + GF} = \frac{4N(s + b)}{s^3 + (Nb - 2)s^2 + (2 + 4N - 2Nb)s + 6Nb}.$$

The application of Theorem 8.1 to the pole polynomial of the closed-loop system gives the following conditions on N *and* b *which ensure stability*

$$6Nb > 0, \quad Nb - 2 > 0, \quad (Nb - 2)(2 + 4N - 2Nb) - 6Nb > 0. \tag{8.1}$$

The first inequality implies that N *and* b *have equal sign and a plot of the semi-algebraic set defined by the other two inequalities is shown in Figure 8.1.*

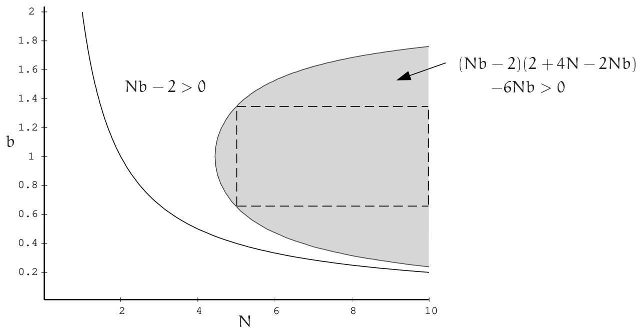


Figure 8.1 *Parameter values in the gray-shaded region given by the inequalities (8.1) correspond to a stable closed-loop system. The dashed rectangle illustrates the interval constraints from (8.2) and (8.3).*

The first question we pose is:

> *For which values of* $b > 0$, *exists a value of* $N \in (1, 10)$ *such that the closed-loop system is stable?*

The second half of this question can be formulated as

$$\exists N \left[1 < N < 10 \ \wedge \ b > 0 \ \wedge \ 6Nb > 0 \right.$$
$$\left. \wedge \ Nb - 2 > 0 \ \wedge \ (Nb-2)(2+4N-2Nb) - 6Nb > 0 \right],$$

where we have used equivalence (4.2) since the quantified variable N *is constrained to a semialgebraic set. Performing quantifier elimination gives*

$$\left[50b^2 - 100b + 21 < 0 \ \wedge \ b > 0 \right],$$

which approximately constrains b *to the interval* $(0.24, 1.76)$*. Compare with the projection onto the* b*-axes of the gray-shaded region in Figure 8.1.*
We can also find the projection of the gray-shaded region onto the N*-axis by quantifying* b *instead, which gives*

$$\left[N^2 - 4N - 2 > 0 \ \wedge \ N > 1 \ \wedge \ N - 10 < 0 \right].$$

Hence, N *is approximately constrained to the interval* $(4.45, 10.0)$*.*
Since the values of N *and* b *cannot be chosen independently the interval constraints only give necessary conditions for stability. To get a sufficient condition we can pose the question:*

> *For which values of* $b > 0$*, hold that for all* $N \in (5, 10)$ *the closed-loop system is stable?*

The second half of this question can be formulated as

$$\forall N \left[[5 < N < 10] \ \rightarrow \ [b > 0 \ \wedge \ 6Nb > 0 \right.$$
$$\left. \wedge \ Nb - 2 > 0 \ \wedge \ (Nb-2)(2+4N-2Nb) - 6Nb > 0] \right], \tag{8.2}$$

where we have used equivalence (4.3) since the quantified variable N *is constrained. Eliminating the quantifier gives*

$$\left[25b^2 - 50b + 22 \leq 0 \right], \tag{8.3}$$

i.e., approximately the interval constraint $b \in (0.66, 1.34)$*. Compare with the dashed rectangle in Figure 8.1.*

Model uncertainties are easy to incorporate in the stability analysis, which is shown in the following example.

Example 8.2 *Suppose that the system is described by*

$$G(s) = \frac{1}{(s - \frac{1}{2})(s^2 + 2\zeta\omega_0 s + \omega_0^2)},$$

where the damping ratio ζ *and frequency* ω_0 *are uncertain but constant parameters where* $\zeta \in (\frac{1}{4}, \frac{1}{3})$ *and* $\omega_0 \in (3, 4)$*. The objective is to stabilize the system with a proportional compensator,* $F(s) = K$ *despite the uncertainties.*

The closed-loop system becomes

$$G_c(s) = \frac{GF}{1+GF} = \frac{2K}{2s^3 + (4\zeta\omega_0 - 1)s^2 + (2\omega_0^2 - 2\zeta\omega)s + 2K - \omega_0^2}.$$

According to the stability criterion in Theorem 8.1 the closed-loop system is stable if and only if

$$2K - \omega_0^2 > 0, \quad 4\zeta\omega_0 - 1 > 0, \quad (4\zeta\omega_0 - 1)(2\omega_0^2 - 2\zeta\omega) - 2(2K - \omega_0^2) > 0 \tag{8.4}$$

and in addition the following inequalities have to be satisfied

$$\tfrac{1}{4} < \zeta < \tfrac{1}{3}, \quad 3 < \omega_0 < 4, \tag{8.5}$$

due to the uncertainties. We can now formulate the stabilization problem as follows:

> *There exists a K such that for all ζ and ω_0 satisfying (8.5), the conditions (8.4) hold.*

The corresponding formula becomes

$$\begin{aligned} \exists K\, \forall\omega_0\, \forall\zeta\, \big[\, [\, \tfrac{1}{4} < \zeta < \tfrac{1}{3} \;\wedge\; 3 < \omega_0 < 4\,] \;\rightarrow \\ [\, 2K - \omega_0^2 > 0 \;\wedge\; 4\zeta\omega_0 - 1 > 0 \;\wedge \\ (4\zeta\omega_0 - 1)(2\omega_0^2 - 2\zeta\omega_0) - 2(2K - \omega_0^2) > 0\,]\,\big], \end{aligned}$$

where we have used equivalence (4.3) since the quantified variables are constrained. Performing quantifier elimination on this decision problem the formula can be shown to be true. If we leave out the first quantifier and eliminate the others, we obtain $8 \le K \le \frac{51}{4}$ which guarantees stability of the closed-loop system.
Given these stabilizing values of K we can compute the region over which ω_0 and ζ are allowed to vary without loss of stability. The formula becomes

$$\begin{aligned} \forall K\, \big[\, \big[\, 8 \le K < \tfrac{51}{4}\,\big] \;\rightarrow \\ [\, 2K - \omega_0^2 > 0 \;\wedge\; 4\zeta\omega_0 - 1 > 0 \;\wedge \\ (4\zeta\omega_0 - 1)(2\omega_0^2 - 2\zeta\omega_0) - 2(2K - \omega_0^2) > 0\,]\,\big], \end{aligned}$$

which is equivalent to

$$0 \le \omega_0 \le 4 \;\wedge\; 8\zeta^2\omega_0^2 - 8\zeta\omega_0^3 - 2\zeta\omega_0 + 51 \le 0.$$

This region is displayed in Figure 8.2.
We observe that the region is larger than the specifications on the uncertainties.

Generalized stability

The stability conditions in terms of inequalities as presented in Theorem 8.1 can be generalized to conditions of what is known as generalized stability [57]. According to properties of linear fractional transformations [16]

$$z = \frac{as + b}{cs + d}$$

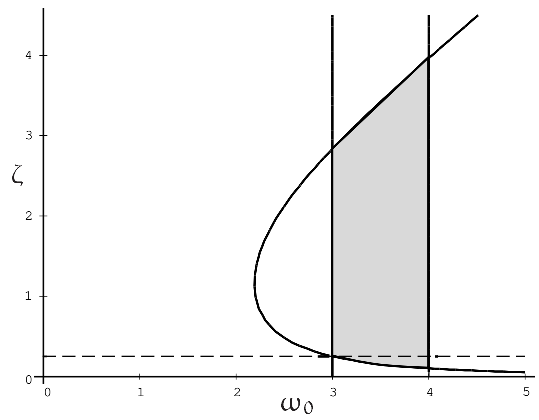


Figure 8.2 *The gray-shaded region corresponds to values of ω_0 and ζ for which any $8 < K < \frac{51}{4}$ stabilizes the system. (Dashed line $\zeta = \frac{1}{4}$).*

the half plane $\mathrm{Re}(s) < 0$ can be mapped onto a new half plane or the interior/exterior of an arbitrarily circle[1]. This together with Theorem 8.1 can be used to derive necessary and sufficient conditions on the coefficients of a transformed polynomial such that its zeros lie in the transformed half plane. We illustrate the idea by an example.

Example 8.3 *Suppose that the plant is given by*

$$G(s) = \frac{2s+1}{s^2 - 2as + 1 + a^2},$$

where a is a constant but unknown uncertainty satisfying $\frac{3}{4} < a < \frac{5}{4}$. The plant is controlled using a proportional controller $F(s) = K,\ K > 0$ such that the closed-loop system becomes

$$G_c(s) = \frac{GF}{1+GF} = \frac{K(2s+1)}{(s^2 - 2as + 1 + a^2) + K(2s+1)}.$$

The objective is to choose K such that the poles of the closed-loop system are located inside a disk of radius 2, centered at -3 on the real axis, regardless of the uncertainty.
The linear fractional transformation

$$z = \frac{s+1}{s+5}, \qquad s = \frac{5z-1}{1-z}$$

[1] Only transformations where a, b, c, d are real numbers can be used since the transformed pole polynomial must be a real rational function for Theorem 8.1 to apply.

maps the disk

$$(Re(s)+3)^2 + Im(s)^2 < 4 \tag{8.6}$$

onto the left half plane $Re(z) < 0$. The zeros of the pole polynomial p of G_c are mapped to the zeros of the transformed pole polynomial, which is a rational function, i.e.,

$$f(z) = p\left(\frac{5z-1}{1-z}\right) = \frac{(10a - 9K + 26 + a^2)z^2 + (10K - 12 - 2a^2 - 12a)z + 2 + a^2 + 2a - K}{(-1+z)^2}. \tag{8.7}$$

Now, Theorem 8.1 applied to the numerator of this rational function, gives necessary and sufficient conditions for the zeros of $f(z)$ to have negative real parts, i.e., for the zeros of $p(s)$ to belong to the disk (8.6). Since the numerator is a polynomial of degree two, a necessary and sufficient condition for stability is that all its coefficients have the same sign.
The stabilization problem may thus be formulated as:

> *For which values on $K > 0$, hold that for all $a \in (\frac{3}{4}, \frac{5}{4})$, the coefficients of the numerator of (8.7) all have the same sign.*

Utilizing equivalence (4.3) we get the following formula describing the conditions on K that guarantee stability.

$$\begin{aligned}\forall a \Big[\; & [\, \tfrac{3}{4} < a < \tfrac{5}{4} \,] \; \rightarrow \\ & \big[[\,(10a - 9K + 26 + a^2) > 0 \;\wedge\; (10K - 12 - 2a^2 - 12a) > 0 \;\wedge \\ & 2 + a^2 + 2a - K > 0\,] \;\vee \\ & [\,(10a - 9K + 26 + a^2) < 0 \;\wedge\; (10K - 12 - 2a^2 - 12a) < 0 \;\wedge \\ & 2 + a^2 + 2a - K < 0\,]\big]\Big].\end{aligned}$$

Performing quantifier elimination results in the stability condition $K \in (\frac{241}{80}, \frac{545}{144})$. The root locus is shown in Figure 8.3 where the roots are located inside the dashed circle for $K \in (\frac{241}{80}, \frac{545}{144}) \approx (3.02, 3.78)$.

8.2 Feedback Design

In this section we apply quantifier elimination to problems in feedback design of linear time-invariant systems. The configuration of the closed-loop system is shown in Figure 8.4.

A common approach to feedback design is to shape the "open-loop" return ratio $G(s)F(s;\eta)$ by choosing a number of parameters, η in the controller, e.g., PID or lead-lag parameters. The objectives of the design are often given in the following form [57]:

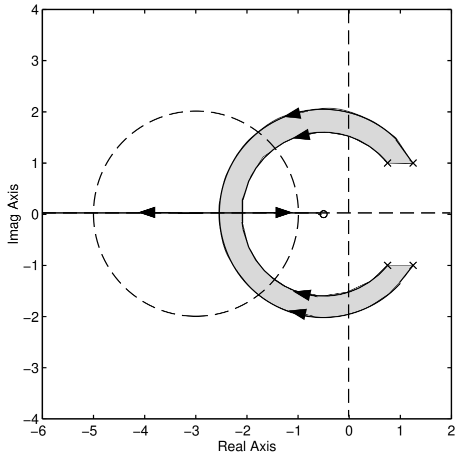


Figure 8.3 *The root locus of* $p(s;K) = 0$, $K > 0$*. Starting points:* $\times$*. End points:* $\circ$*. The uncertainty corresponds to a locus in the gray-shaded region.*

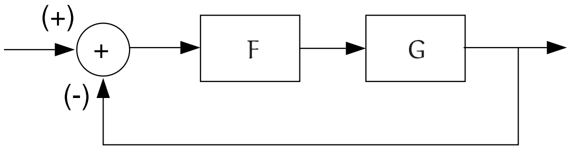


Figure 8.4 *The closed-loop system.*

(i) $|G(i\omega)F(i\omega;\eta)| > L \quad (\gg 1), \quad 0 < \omega < \omega_1.$

(ii) $|G(i\omega)F(i\omega;\eta)| < \epsilon \quad (\ll 1), \quad \omega_1 < \omega < \omega_2.$

(iii) Gain margin $> \mu$ and phase margin $> \phi$.

(iv) The graph of $G(i\omega)F(i\omega;\eta)$ is to remain outside a neighborhood of the point $-1 + i\,0$.

It is well known that robustness and performance specifications on the closed-loop system can be translated to the above requirements. More about this subject can be found in any textbook on control systems, see e.g., [57].

The above objectives can all be formulated as logical combinations of equality and inequality constraints on the real and imaginary part of the return ratio. Let

$$x_1(\omega;\eta) = \mathrm{Re}[G(i\omega)F(i\omega;\eta)] \quad \text{and} \quad x_2(\omega;\eta) = \mathrm{Im}[G(i\omega)F(i\omega;\eta)],$$

where we assume that x_1 and x_2 are rational expressions in ω and η. In what follows we suppress the ω and η dependence of x_1 and x_2. The requirements (i) and (ii) are both constraints on the modulus of the complex number $x_1 + ix_2$ for certain frequencies and hence correspond to the exterior and interior of two circles, respectively. This can be formalized as

$$0 < \omega < \omega_1 \;\rightarrow\; x_1^2 + x_2^2 > L^2, \tag{8.8}$$

$$\omega > \omega_2 \;\rightarrow\; x_1^2 + x_2^2 < \epsilon^2. \tag{8.9}$$

We notice that the denominator polynomial on the left hand side of these equations always is positive. Hence, we can clear fractions and get equivalent polynomial inequalities.

The gain and phase margin are defined as in Figure 8.5.

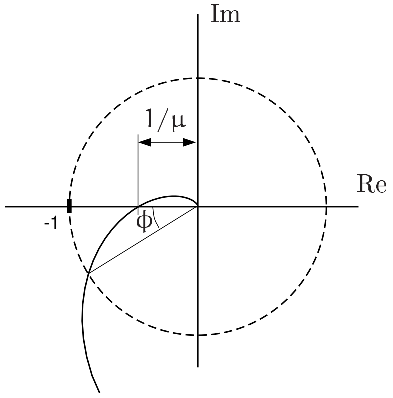


Figure 8.5 *The definition of phase margin, ϕ and gain margin, μ.*

The gain margin requirement can be stated as $A_m > \mu$ or in terms of x_1 and x_2 as

$$x_2 = 0 \quad \rightarrow \quad x_1 > -1/\mu. \tag{8.10}$$

The constraint on the phase margin can be formulated as follows: if the modulus of the return ratio equals one and its real and imaginary part are both negative then the angle to the negative real axis should exceed ϕ. On a more compact form we have $(0 < \phi < \pi)$

$$\left[x_1^2 + x_2^2 = 1 \;\wedge\; x_2 < 0\right] \;\rightarrow\; \frac{x_2}{x_1} > \tan(\phi). \tag{8.11}$$

The "forbidden" area defined in requirement (iv) is often taken as the interior of an M-circle which imposes some minimum degree of damping on the closed-loop poles [57]. The interior of an M-circle is

$$\left\{ z \in \mathbb{C} \,\middle|\, \left|\frac{z}{1+z}\right| > M,\; M > 1 \right\},$$

where the condition $M > 1$ ensures that the M-circle encloses $-1+i0$. Requirement (iv) corresponds to

$$\frac{x_1^2 + x_2^2}{(1+x_1)^2 + x_2^2} < M^2. \tag{8.12}$$

Investigating the rational expressions in (8.8) to (8.12) one realizes that we can clear fractions without having to keep track on the directions of the inequality signs except in (8.11) where the inequality sign switches direction due to the assumption $x_1 < 0$. We now have a number of formulas where the components are polynomials in x_1 and x_2.

$$\begin{aligned} [0 < \omega < \omega_1] \;&\rightarrow\; x_1^2 + x_2^2 > L^2, \\ \omega > \omega_2 \;&\rightarrow\; x_1^2 + x_2^2 < \epsilon^2, \\ x_2 = 0 \;&\rightarrow\; x_1 > -1/\mu, \\ \left[x_1^2 + x_2^2 = 1 \;\wedge\; x_2 < 0\right] \;&\rightarrow\; x_2 < x_1 \tan(\phi), \\ x_1^2 + x_2^2 &< M^2 \left((1+x_1)^2 + x_2^2\right), \end{aligned} \tag{8.13}$$

where $\omega_1, L, \epsilon, \mu, \phi$ and M are predefined constants chosen by the designer. The real and imaginary parts of the return ratio are rational functions in ω and η but it is easy to rewrite the above formulas such that all inequalities become polynomial. If we now require the resulting formulas to be true for all real values of ω we get a quantifier elimination problem which can be solved using the algorithms in Section 4.5. We do not write down the general formula but instead give an example of the method.

Example 8.4 *Consider the following second order system and PI-controller*

$$G(s) = \frac{2}{(s+1)(s+2)}, \qquad F(s; k_p, k_i) = k_p + \frac{k_i}{s},$$

which result in the open-loop return ratio

$$G(i\omega)F(i\omega; k_p, k_i) = \frac{2(k_p s + k_i)}{s(s+1)(s+2)}.$$

Suppose that the objective is to find k_p and k_i such that the following specifications are fulfilled

(i) $|G(i\omega)F(i\omega; k_p, k_i)| > 2, \quad 0 < \omega < \frac{3}{4}$,

(ii) $|G(i\omega)F(i\omega; k_p, k_i)| < \frac{1}{4}, \quad \omega > 3$,

(iii) The Nyquist curve is to remain outside the M-circle with $M = \sqrt{2}$.

Following the previous discussion we first derive rational expressions for the real and imaginary part of the return ratio, x_1 and x_2, respectively

$$\begin{aligned} x_1 &= \mathit{Re}[G(i\omega)F(i\omega; k_p, k_i)] = \frac{-2k_p\omega^2 + 4k_p - 6k_i}{(1+\omega^2)(4+\omega^2)}, \\ x_2 &= \mathit{Im}[G(i\omega)F(i\omega; k_p, k_i)] = \frac{(2k_i - 6k_p)\omega^2 - 4k_i}{\omega(1+\omega^2)(4+\omega^2)}. \end{aligned}$$

Using these expressions the formulas corresponding to (8.13) become

$$\forall\omega\left[\left[0<\omega<\tfrac{3}{4}\right]\ \rightarrow\ k_p^2\omega^2+k_i^2>\omega^2(1+\omega^2)(4+\omega^2)\right],$$

$$\forall\omega\left[\omega>3\ \rightarrow\ 4k_p^2\omega^2+4k_i^2<\tfrac{1}{4^2}\omega^2(1+\omega^2)(4+\omega^2)\right],$$

$$\forall\omega\left[\omega^6+(5-4k_p)\omega^4+(2k_p^2+8k_p-12k_i+4)\omega^2+2k_i^2>0\right].$$

If we perform quantifier elimination on each of the above formulas, then the following conditions on k_p and k_i can be established

$$2304k_p^2+4096k_i^2>16425,$$

$$288k_p^2+32k_i^2<585,$$

which are the exterior and interior of two ellipsoids and

$$\begin{aligned}k_i\neq 0\ \wedge\ &8k_p^6-8k_p^5-48k_p^4k_i+56k_p^3k_i^2-407k_p^4+864k_p^3k_i-966k_p^2k_i^2\\&+432k_pk_i^3-27k_i^4-1240k_p^3+4116k_p^2k_i-4080k_pk_i^2+1188k_i^3\\&-940k_p^2+2064k_pk_i-898k_i^2-144k_p-24k_i+36<0,\end{aligned}$$

which is a more complicated semialgebraic set[2]*. A visualization of a part of these constraints is given in Figure 8.6.*

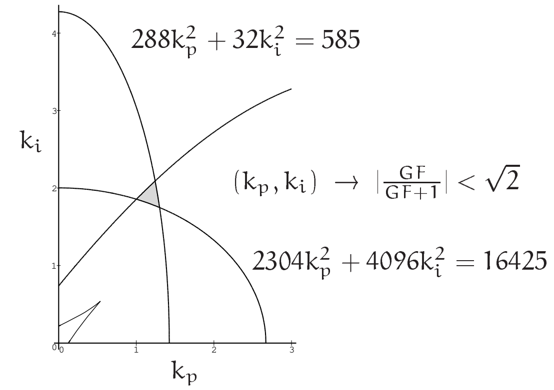


Figure 8.6 *The region in the k_pk_i-plane (gray-shaded), which gives a return ratio satisfying the specifications. The peak at the origin seems to be an anomaly due to that the quantifier elimination program only succeeded in giving a sufficient but not equivalent condition.*

The pair $k_p=1.2, k_i=2$ gives a return ratio that fulfills the specifications and the Nyquist curve of the return ratio for these specific values is shown in Figure 8.7.

[2]The program QEPCAD which is used to perform quantifier elimination stops with these as sufficient conditions.

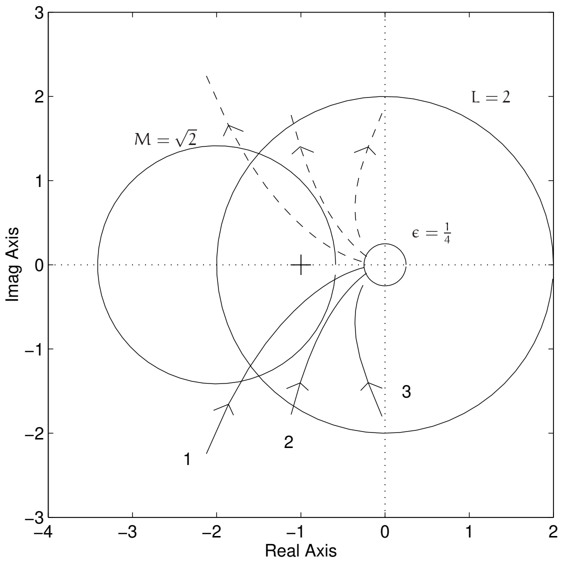


Figure 8.7 *The Nyquist curve of* $GF(i\omega)$ *plotted for* $\frac{3}{4} < \omega < 3$. *Curve 1:* $k_p = 1, k_i = 3$, *curve 2:* $k_p = 1.2, k_i = 2$ *and curve 3:* $k_p = 2, k_i = 1$. *We notice that curve 2 fulfills the objectives.*

To ensure that the Nyquist curve does not encircle the M-circle we could have added a condition of the type $x_2 = 0 \rightarrow x_1 > -1$.

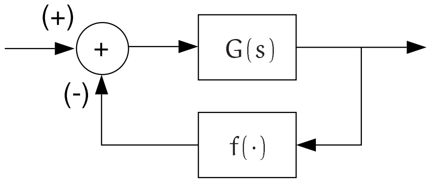
Observe that the solution set in this example is a non-convex set, which means that the problem cannot be treated successfully with a convex programming approach.

The Circle and Popov Criterion

Conditions given on the Nyquist curve also appear in stability analysis of feedback schemes where one of the blocks is a static nonlinearity, see Figure 8.8. The small gain theorem and its implications for these schemes such as the circle and Popov criterion give various sufficient conditions on the linear system to ensure stability, see [81].

The circle criterion states that the closed-loop system is stable if the Nyquist curve $G(i\omega)$ does not enclose or enter the disk

$$\mathcal{D} \triangleq \left\{ z \in \mathbb{C} \middle| \left| z + \frac{k_2 + k_1}{2k_1 k_2} \right| \leq \left| \frac{k_2 - k_1}{2k_1 k_2} \right| \right\}. \tag{8.14}$$



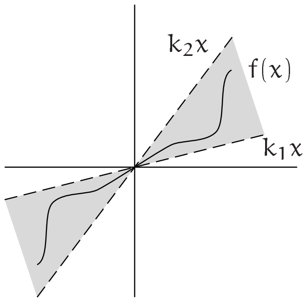


Figure 8.8 *Left: The system configuration for the circle and Popov criterion. Right: Limits on the static nonlinearity ($k_1 = 0$ in the Popov criterion.)*

This is a condition of the same type as the requirement that the Nyquist curve is to remain outside an M-circle which was treated previously.

Example 8.5 *Suppose we are interested in bounds (k_1, k_2) on a static nonlinearity $f(\cdot)$ such that the linear system described by*

$$G(s) = \frac{1}{s^3 + 2s^2 + 3s + 1}$$

remains stable under feedback according to Figure 8.8. The real and imaginary part of the Nyquist curve become

$$x_1 = \frac{1 - 2\omega^2}{(1 - 2\omega^2)^2 + (3\omega - \omega^3)^2},$$
$$x_2 = \frac{3\omega - \omega^3}{(1 - 2\omega^2)^2 + (3\omega - \omega^3)^2}.$$

and the Nyquist curve remains outside the disk $\mathcal{D}$ if for all ω the following inequality is satisfied

$$\left(x_1 + \frac{k_1 + k_2}{2k_1k_2}\right)^2 + x_2^2 > \frac{(k_2 - k_1)^2}{4k_1^2k_2^2},$$

where the ω dependence is suppressed. Clearing denominators and simplifying this expression give the following equivalent formula

$$\forall\omega\,\big[\,\omega^6 - 2\omega^4 + (5 - 2k_1 - 2k_2)\omega^2 + k_1 + k_2 + k_1k_2 + 1 > 0 \,\wedge\, k_1 > 0 \,\wedge\, k_2 > k_1\,\big], \quad (8.15)$$

where the original conditions on k_1 and k_2 also are included.

However, we also have to ensure that the Nyquist curve does not enclose $\mathcal{D}$. This can be prevented in various ways, e.g., if for all $\omega > 0$

$$x_2(\omega) > 0 \;\rightarrow\; x_1(\omega) \neq -\frac{1}{k_2}$$

which corresponds to the formula

$$\begin{aligned}\forall\omega \Big[\, &[\,\omega > 0\,] \;\rightarrow \\ &\big[\,[\,\omega^3 - 3\omega > 0\,] \;\rightarrow\; [\,\omega^6 - 2\omega^4 + (5 - 2k_2)\omega^2 + 1 + k_2 \neq 0 \\ &\qquad\qquad \wedge\; k_1 > 0 \;\wedge\; k_2 > k_1\,]\big]\Big].\end{aligned} \tag{8.16}$$

Performing quantifier elimination in (8.15) and (8.16) gives

$$\begin{aligned}\big[\,0 < k_1 < k_2 \;\wedge\; k_2 < 5 \;\wedge\; &-27k_1^2k_2^2 + 32k_1^3 + 114k_1^2k_2 + 114k_1k_2^2 \\ &+ 32k_2^3 - 179k_1^2 - 560k_1k_2 - 179k_2^2 + 390k_1 + 390k_2 - 575 < 0\,\big]\end{aligned} \tag{8.17}$$

as sufficient conditions for the bounds (k_1, k_2) on a static nonlinearity to ensure closed-loop stability. The semialgebraic set in the k_1k_2-plane described by (8.17) is shown in Figure 8.9.
We also show the Nyquist curve and a number of limiting circles. The vertical line in this figure corresponds to $k_1 = 0, k_2 = \frac{115}{32}$.

In the above example the system $G(s)$ was fixed. Additional "parameters to be determined" in the linear system as well as fixed bounds on the nonlinearity can also be taken into consideration in various ways.

To infer stability using the Popov criterion, one has to show that the so called Popov curve lies to the right of a straight line, whose location depends on the nonlinearity, see [81]. This condition can easily be given as a formula involving quantified variables and stability can be proved by deciding if the formula is true or not.

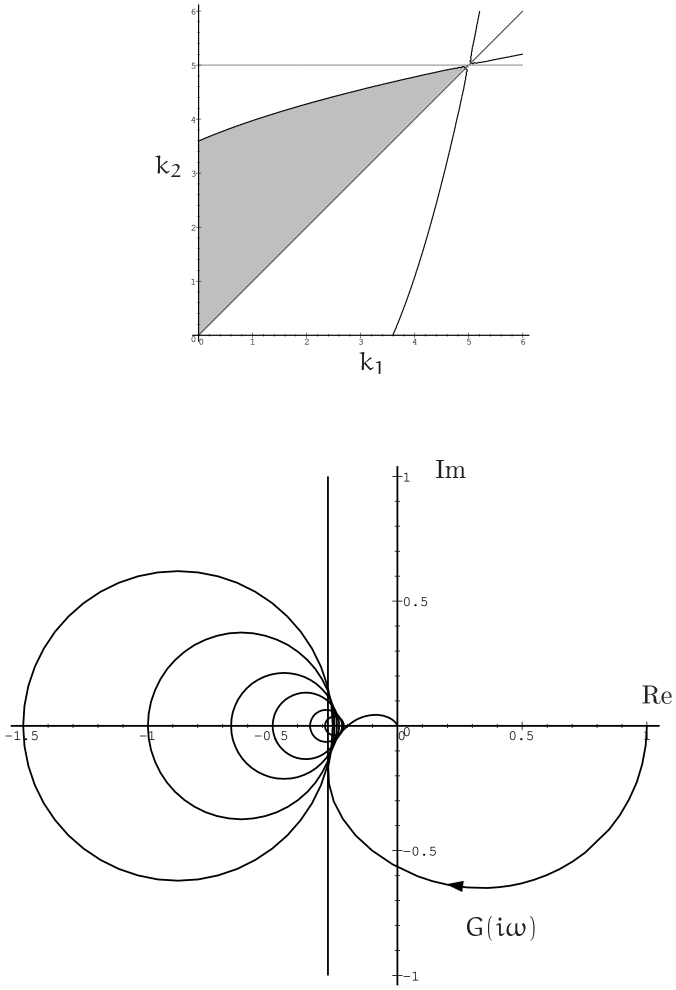


Figure 8.9 *Top: Region of* (k_1, k_2)*-pairs that guarantees closed-loop stability (gray-shaded). Bottom: The Nyquist curve and a number of circles corresponding to different* (k_1, k_2)*-pairs just fulfilling the conditions.*

9

Applications to Nonlinear Systems

In this chapter we discuss some applications of quantifier elimination to nonlinear control theory. Since the basic framework is real algebra and real algebraic geometry we consider dynamical systems described by differential and non-differential equations and inequalities in which all nonlinearities are of *polynomial* type. This represents a rather large class of systems and it can be shown that systems where the nonlinearities are not originally polynomial may be rewritten on this polynomial form if the nonlinearities themselves are solutions to algebraic differential equations. For more details on this, see [55, 74]

Given a classical state space description of the system (i.e., the system is described by a number of first order differential equations) and constraints on the states as well as control signals we consider three classes of problems which can be solved by eliminating quantifiers:

(i) Which states correspond to equilibrium points of the dynamical system for some admissible control signal?

(ii) Which stability properties do these equilibrium points have?

(iii) Given a parameterized curve in the state space of the system. Is it possible to follow the curve by use of available control signals? More generally, given a set of parameterized curves, which states can be reached by following one of these curves?

Stationary (equilibrium, critical) points play an important role in both analysis and design of dynamical systems and for synthesis of control laws. These points are the possible "operating points" of the system and often a control law is designed such that the state of the system will return to such a point after moderate disturbances. The set of (stable) equilibrium points of a dynamical system is parameterized by the available control signals and the first and second problem address the construction of this set for polynomial systems.

The last problem is a natural question in many control situations where the objective is to steer the dynamical system from one point to another along a certain path. Observe that the prescribed path belongs to the state space, which implies that the whole system dynamics is specified. Hence this is an extension of the motion planning problem also taking into account the system dynamics. This problem is also generalized to a constrained form of computable reachability.

The chapter is organized as follows. In Section 9.1 we show how the states that can be made stationary by use of an admissible control can be computed. In Section 9.2 we consider the same problem but also require asymptotical stability. Finally, curve following and constrained reachability are presented in Section 9.3.

9.1 Stationarizable Points

It will be assumed that the dynamical system is described by a nonlinear differential equation written in state space form

$$\begin{aligned} \dot{x} &= f(x, u) \\ y &= h(x) \end{aligned} \tag{9.1}$$

where x is an n-vector, u an m-vector, y a p-vector and each component of f and h is a real polynomial, $f_i \in \mathbb{R}[x, u]$, $h_j \in \mathbb{R}[x]$. The x, u and y vectors will be referred to as the *state, control* and *output* of the system respectively.

Suppose also that the system variables have to obey some additional constraints

$$x \in \mathcal{X} \quad \text{and} \quad u \in \mathcal{U}, \tag{9.2}$$

where $\mathcal{X}$ and $\mathcal{U}$ are semialgebraic sets which define the constraints on the state and control variables. We call $x \in \mathcal{X}$ the *admissible* states and $u \in \mathcal{U}$ the *admissible* controls.

A variety of constraints can be represented in the semialgebraic framework, e.g., amplitude and direction constraints.

Example 9.1 *Let* $\mathbf{F}$ *be a two dimensional thrust vector which can be pointed in any direction,* ϕ *and whose magnitude,* $|\mathbf{F}|$ *can be varied between* 0 *and* F_{max}*. Let*

$$u_1 = \cos(\phi), \quad u_2 = \sin(\phi) \quad \textit{and} \quad u_3 = |\mathbf{F}|.$$

Then the semialgebraic set describing these constraints becomes

$$\mathcal{U} = \left\{ u \in \mathbb{R}^3 \mid u_1^2 + u_2^2 = 1 \ \wedge \ 0 \le u_3 \le F_{max} \right\}$$

Similarly, constraints on the states may originate from specifications on the system outputs, e.g., $|h(x)| \le \epsilon$.

The main question in this section concerns equilibrium or stationary points of a dynamical system, i.e., solutions of (9.1) which correspond to constant values of the admissible states and controls. In other words, we are interested in those admissible states for which the system can be brought to rest by use of an admissible control. The conditions for a point, x_0 to be stationary are easily seen to be $f(x_0, u_0) = 0$, where $x_0 \in \mathcal{X}$ and $u_0 \in \mathcal{U}$. For the class of dynamical system considered here, this set of *stationarizable* states turns out to be a semialgebraic set.

Definition 9.1 *The stationarizable states of system (9.1) subjected to the constraints (9.2) is the set of states satisfying the formula*

$$\mathcal{S}(x) \triangleq \exists u \left[f(x,u) = 0 \ \wedge \ \mathcal{X}(x) \ \wedge \ \mathcal{U}(u) \right]. \tag{9.3}$$

The computation of a “closed form” of the set of stationarizable states, i.e., an expression not including u, is a quantifier elimination problem and hence this set is semialgebraic.

Example 9.2 *Consider the following system*

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2^2 + u \\ \dot{x}_2 &= -x_2 + x_1^2 \end{aligned}$$

subjected to the constraints

$$\begin{aligned} u^2 &\leq 1 \\ x_1^2 + x_2^2 &\leq 1. \end{aligned}$$

According to Definition 9.1 the stationarizable set is described by the formula

$$\exists u \left[-x_1 + x_2^2 + u = 0 \ \wedge \ -x_2 + x_1^2 = 0 \ \wedge \ u^2 \leq 1 \ \wedge \ x_1^2 + x_2^2 \leq 1 \right],$$

which after quantifier elimination becomes

$$\left[x_2^2 - x_1 - 1 \leq 0 \ \wedge \ x_1^2 + x_2^2 - 1 \leq 0 \ \wedge \ x_2 - x_1^2 = 0 \right]$$

In this case the stationarizable set is easy to visualize, see Figure 9.1.

As an example of a specific application of the stationarizability result we consider the control of an aircraft.

Example 9.3 *In advanced aircraft applications the orientation of the aircraft with respect to the airflow can be controlled. The orientation is usually described by the angle of attack, α and sideslip angle, β of the aircraft, see Figure 9.2.*
An interesting question is for which α and β the orientation of the aircraft can be kept constant by admissible control surface configurations? It can be shown,

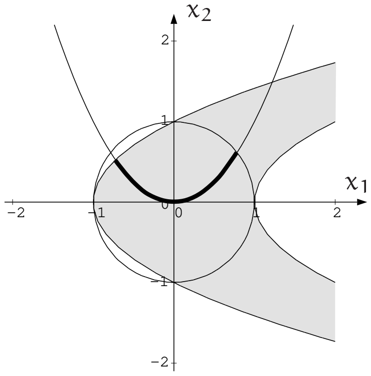


Figure 9.1 The projection onto the x_1x_2-plane of the zero sets of $-x_2+x_1^2$ and $-x_1+x_2^2+u$ (gray region) for admissible controls. The admissible states are those inside the unit circle and the stationarizable set is the bold curve.

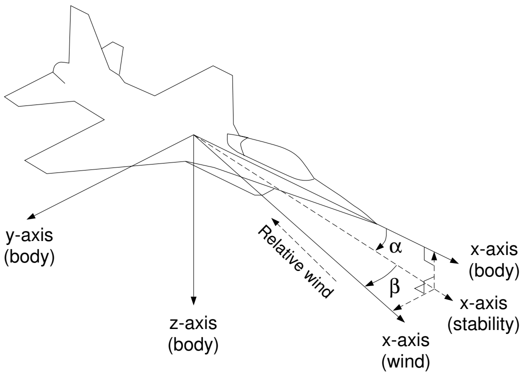


Figure 9.2 The orientation of an aircraft with respect to the airflow.

using the equations of motion of an aircraft [76], that α and β are constant if the aerodynamic moments acting on the aircraft are zero. These moments are nonlinear functions of α, β and the control surface deflections, and they are usually given in tabular form together with some interpolation method. In [76] these functions are given as look-up tables together with Fortran code and the following are scaled *polynomial approximations of these functions*

$$
\begin{aligned}
C_L(x_1, x_2, u1, u3) =& -38x_2 - 170x_1x_2 + 148x_1^2x_2 + 4x_2^3 + \\
& u_1(-52 - 2x_1 + 114x_1^2 - 79x_1^3 + 7x_2^2 + 14x_1x_2^2) + \\
& u_3(14 - 10x_1 + 37x_1^2 - 48x_1^3 + 8x_1^4 - 13x_2^2 - 13x_1x_2^2 + \\
& 20x_1^2x_2^2 + 11x_2^4) \\
C_M(x_1, u_2) =& -12 - 125u_2 + u_2^2 + 6u_2^3 + 95x_1 - 21u_2x_1 + 17u_2^2x_1 \\
& -202x_1^2 + 81u_2x_1^2 + 139x_1^3 \\
C_N(x_1, x_2, u1, u3) =& 139x_2 - 112x_1x_2 - 388x_1^2x_2 + 215x_1^3x_2 - 38x_2^3 + 185x_1x_2^3 + \\
& u_1(-11 + 35x_1 - 22x_1^2 + 5x_2^2 + 10x_1^3 - 17x_1x_2^2) + \\
& u_3(-44 + 3x_1 - 63x_1^2 + 34x_2^2 + 142x_1^3 + 63x_1x_2^2 - 54x_1^4 \\
& -69x_1^2x_2^2 - 26x_2^4)
\end{aligned}
$$

where x_1 and x_2 are normalized angle of attack and sideslip angle and u_1, u_2, u_3 are the aileron, elevator and rudder deflections respectively.
The question of constant orientation may now be posed as

$$\exists u_1 \exists u_2 \exists u_3 \left[C_L = 0 \ \wedge \ C_M = 0 \ \wedge \ C_N = 0 \ \wedge \ u_i^2 \leq 1, i = 1, 2, 3 \right],$$

where the answer, a formula in x_1 and x_2, defines the semialgebraic set describing the possible stationarizable angle of attack and sideslip angle.
Unfortunately, this quantifier elimination problem was too complex to be solved by the version of QEPCAD *available to us.*

9.2 Stability

In stability theory for nonlinear dynamical systems one is often interested in the character of the solution in the neighborhood of a stationary point. If all solutions starting in a small neighborhood of a stationary point x_0 stays within this neighborhood for all future times, then the stationary point is called ***stable***. If in addition the solutions converges towards x_0, the stationary point is called (locally) ***asymptotically stable***. For an extensive treatment of stability of dynamical systems see [39].

The following theorem gives a sufficient condition for asymptotic stability of a stationary point.

Theorem 9.1 *Let x_0 be a stationary point of system (9.1) corresponding to $u = u_0$. Then x_0 is asymptotically stable if all eigenvalues of $f_x(x_0, u_0)$ have strictly negative real parts.*

Proof *See [39].* □

Here f_x denotes the functional matrix (or Jacobian) of $f(x, u)$ regarding u as a constant.

Since the eigenvalues of a matrix are the zeros of its characteristic polynomial we are interested in determining if all the zeros of this polynomial have strictly negative real parts. The question can be answered in a number of different ways, by examining the coefficients of the characteristic polynomial, e.g., by the criteria of Hurwitz, Routh or Lienard-Chipart.

In Theorem 8.1 we presented the Lienard-Chipart criterion. This gives a polynomial criterion for testing the stability of a stationary point. The characteristic polynomial of $f_x(x_0, u_0)$ in Theorem 9.1 is $\det(\lambda I_n - f_x(x_0, u_0))$, i.e., a polynomial in λ with coefficients that are polynomials in x_0 and u_0. Utilizing Theorem 8.1 we get n polynomial inequalities in x_0 and u_0, which are necessary and sufficient conditions for the stationary point x_0 to be asymptotically stable.

We summarize the above discussion in the following theorem.

Theorem 9.2 *The stationarizable states of system (9.1) subjected to the constraints (9.2) that are asymptotically stable are given by the formula*

$$\mathcal{AS}(x) \triangleq \exists u \left[f(x, u) = 0 \ \wedge \ \mathcal{X}(x) \ \wedge \ \mathcal{U}(u) \ \wedge \ \mathit{Re}(\mathit{eig}(f_x(x_0, u_0))) < 0 \right], \tag{9.4}$$

where $\mathit{Re}(\mathit{eig}(f_x(x_0, u_0))) < 0$ *denotes the set of inequalities corresponding to Theorem 8.1.*

Example 9.4 *Consider the following system*

$$\begin{aligned} \dot{x}_1 &= -x_1^3 + x_2 \\ \dot{x}_2 &= -x_1^2 - x_2 - x_2^3 + u \end{aligned} \tag{9.5}$$

subjected to the constraints

$$u^2 \leq 1.$$

We get the functional matrix

$$f_x(x, u) = \begin{bmatrix} -3x_1^2 & 1 \\ -2x_1 & -1 - 3x_2^2 \end{bmatrix}$$

and its corresponding characteristic polynomial

$$\lambda^2 + \left(3 x_1{}^2 + 1 + 3 x_2{}^2\right) \lambda + 3 x_1{}^2 + 9 x_1{}^2 x_2{}^2 + 2 x_1$$

The inequalities $\mathit{Re}(\mathit{eig}(f_x(x_0, u_0))) < 0$ *become*

$$3 x_1{}^2 + 1 + 3 x_2{}^2 > 0, \quad 3 x_1{}^2 + 9 x_1{}^2 x_2{}^2 + 2 x_1 > 0,$$

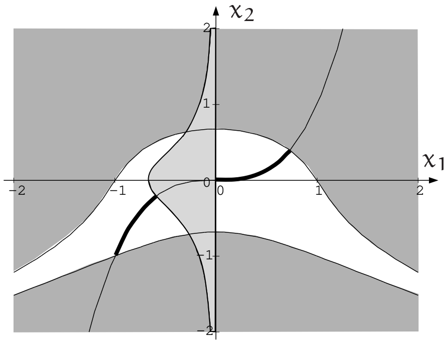


__Figure 9.3__ The set of states of system (9.5) which is stationarizable and asymptotically stable (bold curve). Any point on the cubic which is not in the dark gray region is a stationarizable state. The part of the cubic in the light gray region corresponds to stationary points which are not asymptotically stable.

where the first inequality is trivially satisfied for all real x_1 and x_2. The asymptotically stable stationarizable points of system (9.5) are given by formula (9.4)

$$\begin{aligned}\mathcal{AS}(x) = \exists u \Big[&-x_1^3 + x_2 = 0 \ \wedge \ -x_1^2 - x_2 - x_2^3 + u = 0 \ \wedge \\ &u^2 \leq 1 \ \wedge \\ &2x_1 + 3x_1^2 + 9x_1^2 x_2^2 > 0 \Big],\end{aligned}$$

which after quantifier elimination becomes

$$\begin{aligned}\mathcal{AS}(x) = \Big[&-x_1^3 + x_2 = 0 \ \wedge \\ &-x_1^2 - x_2 - x_2^3 - 1 \leq 0 \ \wedge \ x_1^2 + x_2 + x_2^3 - 1 \leq 0 \ \wedge \\ &\big[x_1 > 0 \ \vee \ 3x_1 + 9x_1 x_2^2 + 2 < 0 \big] \Big],\end{aligned} \tag{9.6}$$

see Figure 9.3.

Observe that points which are stationarizable but not asymptotically stable equilibrium points can be chosen as operating points in applications as well, but the control in this case has to be active which in general is a harder problem (e.g., stabilization of an inverted pendulum).

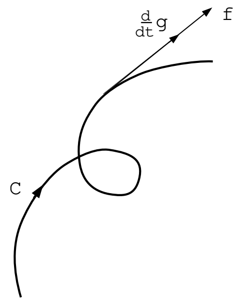


Figure 9.4 *The direction constraint on* $f(x,u)$.

9.3 Curve Following

Consider a parameterized curve in $\mathbb{R}^n$

$$C:\ x = g(t), \quad t \in [\alpha, \beta], \quad g : \mathbb{R} \to \mathbb{R}^n,$$

whose orientation is defined by increasing t. Furthermore, all components of g are assumed to be polynomials in t. Given the system (9.1) subjected to the semialgebraic state and control constraints (9.2), is it then possible to steer the system from the initial state $x^0 = g(\alpha)$ to the final state $x^1 = g(\beta)$ following the curve?

To steer the system along the curve there has to be an admissible control u at each point on the curve such that the solution trajectory tangent vector, $f(x,u)$ points in the same direction as a forward pointing tangent vector of the curve, i.e.,

$$f(g(t),u) = \lambda \frac{d}{dt} g(t), \ \ \lambda > 0, \quad t : \alpha \to \beta,$$

see Figure 9.4.

The above question can be formulated as a quantifier elimination problem as follows:

$$\Big(\forall t \in [\alpha, \beta]\Big)\ \Big(\exists u \in \mathcal{U}\Big)\ \Big(\exists \lambda > 0\Big)\ \Big[\, f(g(t),u) = \lambda \frac{d}{dt} g(t)\ \wedge\ \mathcal{X}(g(t))\Big],$$

where the condition $\mathcal{X}(g(t))$ corresponds to the obvious assumption that every point on C is an admissible state.

According to the simplification rules in Section 4.5 the above formula is equivalent to

$$\forall t\, \exists u\, \exists \lambda \Big[\big[\, \alpha \le t \le \beta \,\big] \ \Rightarrow \\ \big[\, f(g(t),u) = \lambda \frac{d}{dt} g(t)\ \wedge\ \lambda > 0\ \wedge\ \mathcal{X}(g(t))\ \wedge\ \mathcal{U}(u)\big]\Big]. \tag{9.7}$$

This is in fact a decision problem since there are no free variables.

How do we construct a control law that steers the system along the curve once we know that it is possible? Eliminating t from the definition of the curve, $x = g(t)$ gives an implicit description, $c(x) = 0$ say, of which C is a subset, see [23] for an algorithmic way of doing this.

The control can now be computed using the fact that the tangent $f(x, u)$ of the solution trajectory is orthogonal to a normal of C. A normal is given by $c_x(x)$ and we have to solve for u in the following equation

$$c_x(x)f(x, u) = 0$$

In fact, this is not the whole truth since $c(x)$ is zero on C. The general condition a control u has to satisfy is

$$c_x(x)f(x, u) \in \langle c(x) \rangle, \quad \Leftrightarrow \quad c_x(x)f(x, u) = q(x)c(x), \quad q \in \mathbb{R}[x], \tag{9.8}$$

i.e., u has to be chosen such that $c_x(x)f(x, u)$ belongs to the ideal generated by c. These control laws give identical system behavior on C but the extra freedom can be used to tune the system behavior outside C. Outside C we also have to modify the control law such that $u \in \mathcal{U}$ is satisfied. In general the implicit description of the curve $c(x) = 0$ consists of several equations. This gives a number of conditions of the form (9.8), one for each component in $c(x)$.

Example 9.5 *Consider the following system*

$$\begin{aligned}\dot{x}_1 &= -x_1 + 2\\ \dot{x}_2 &= -x_2 - x_1^2 + 4u\end{aligned}$$

subjected to the constraints

$$u^2 \leq 1.$$

We want to decide if it is possible to follow the curve

$$C: \quad x = g(t) = \begin{bmatrix} t \\ 3t^2 - 2t^3 \end{bmatrix}, \quad t: 0 \to 1,$$

using an admissible control.
The quantifier formulation (9.7) of the problem becomes

$$\begin{aligned}\forall t\, \exists u\, \exists \lambda \Big[\big[0 \leq t \leq 1\big] \ \Rightarrow \\ & \big[-t + 2 = \lambda \ \wedge\ -(3t^2 - 2t^3) - t^2 + 4u = \lambda(6t - 6t^2) \ \wedge \\ & \lambda > 0 \ \wedge\ u^2 \leq 1\big]\Big],\end{aligned}$$

which can be shown to be true!

We now compute the control laws that steer the system along C. *An implicit description of* C *is* $x_2 - 3x_1^2 + 2x_1^3 = 0$ *and the orthogonality condition (9.8) give*

$$(-6x_1 + 6x_1^2)(-x_1 + 2) - x_2 - x_1^2 + 4u \in \langle x_2 - 3x_1^2 + 2x_1^3 \rangle.$$

In general one has to check that the chosen control law steers the system in the right direction along C. *In this example we know that there exists a control law which steers the system in the right direction on* C *but there is also only one way of choosing* u *modulo* $\langle c \rangle$ *on* C. *Hence any of the above* u *can be chosen, e.g.,*

$$u = \frac{3}{2}x_1^3 - \frac{17}{4}x_1^2 + 3x_1 + \frac{1}{4}x_2,$$

which is a state feedback control law that steers the system along C *in the right direction.*

Constrained reachability

The important concept of reachability, i.e., questions about which states can be reached from a given set of initial states by a system, is not in general solvable by algebraic methods[1]. The reason is that generically the solution of a system of differential equations such as (9.1) is not an algebraic set or even a subset thereof. However, a more restricted form of reachability can be investigated using semialgebraic tools.

Let $\mathcal{I}$ be a semialgebraic set defining possible initial conditions of system (9.1) and $\mathcal{C}$ a family of parameterized curves

$$\mathcal{C} = \Big\{ C: \ x = g(t; x^0, x^1, \theta), \ t: \alpha \to \beta \,\Big|\, \mathcal{I}(g(\alpha)) \Big\},$$

where each component of g is a polynomial in t, x^0, x^1, θ and $g(\alpha) = x^0$, $g(\beta) = x^1$. Here $\theta \in \Theta$ are some additional parameters to get more flexibility.

Definition 9.2 *We say that a curve,* $C \in \mathcal{C}$ *is admissible if*

(i) $x^0 \in \mathcal{I}$.

(ii) All points on C *belong to the admissible states,* $\mathcal{X}$.

(iii) There is an admissible control u *such that the solution trajectory of (9.1) follows* C.

Definition 9.3 *The set* $\mathcal{R}(\mathcal{I}) \subset \mathcal{X}$ *which can be reached by using an admissible control* u *such that the solution trajectory of (9.1) follows one of the curves in* $\mathcal{C}$ *is called the* $\mathcal{C}$*-reachable set w.r.t.* $\mathcal{I}$.

[1] For finite discrete event systems, reachability can be treated completely algebraically, see [31, 38, 69].

Using a family, $\mathcal{C}$ of parameterized curves which are very flexible, (e.g., Bézier curves, see [23]) the $\mathcal{C}$-reachable set w.r.t. some set of possible initial states should be a good approximation to the ordinary reachable set.

The computation of the set $\mathcal{R}(\mathcal{I})$ can be carried out by eliminating quantifiers as follows.

The first two conditions in Definition 9.2 are easily characterized semialgebraically as $\mathcal{I}(x^0) \wedge \mathcal{X}(g(t;x^0,x^1,\theta))$ and the third condition is the one just treated above. We get the following semialgebraic characterization

$$\Big(\exists x^0 \in \mathcal{I}\Big)\Big(\exists \theta \in \Theta\Big)\Big(\forall t \in (\alpha,\beta)\Big)\Big(\exists u \in \mathcal{U}\Big)\Big(\exists \lambda > 0\Big) \\ \Big[f(g(t),u) = \lambda \frac{d}{dt} g(t) \wedge \mathcal{X}(g(t))\Big],$$

or equivalently

$$\exists x^0\, \exists \theta\, \forall t\, \exists u\, \exists \lambda \Big[\big[\, \alpha \le t \le \beta\,\big] \to \big[\, f(g(t;x^0,x^1,\theta),u) = \lambda \frac{d}{dt} g(t;x^0,x^1,\theta) \\ \wedge\ \lambda > 0\ \wedge\ \mathcal{I}(x^0)\ \wedge\ \mathcal{X}(g(t;x^0,x^1,\theta))\ \wedge\ \mathcal{U}(u)\,\big]\Big]. \tag{9.9}$$

The order of the quantifiers is very important, e.g., θ and x_0 come before t since the direction constraints must be fulfilled along at least one *specific* curve, for the formula to be true. After quantifier elimination we get a real polynomial system in x^1 defining the $\mathcal{C}$-reachable set w.r.t. $\mathcal{I}$.

Example 9.6 *Consider the following system*

$$\begin{aligned} \dot{x}_1 &= x_1 + u \\ \dot{x}_2 &= x_2^2 \end{aligned} \tag{9.10}$$

subjected to the control constraints

$$u^2 \le 1.$$

Which states are reachable along straight lines from the point $(x_1,x_2) = (0,1)$*? The initial state set*

$$\mathcal{I} = \Big\{\, x^0 \in \mathbb{R}^2 \;\Big|\; x_1^0 = 0 \wedge x_2^0 = 1 \,\Big\}$$

and a parameterization of straight lines is

$$C: \quad x = g(t) = t \begin{bmatrix} x_1^1 - x_1^0 \\ x_2^1 - x_2^0 \end{bmatrix} + \begin{bmatrix} x_1^0 \\ x_2^0 \end{bmatrix}, \quad t : 0 \to 1.$$

Formula (9.9) becomes

$$\exists x^0\, \forall t\, \exists u\, \exists \lambda \Big[\big[\, 0 \le t \le 1\,\big] \ \to\ \big[\, t(x_1^1 - x_1^0) + x_1^0 + u = \lambda(x_1^1 - x_1^0) \\ \wedge\ \big(t(x_2^1 - x_2^0) + x_2^0\big)^2 = \lambda(x_2^1 - x_2^0) \\ \wedge\ \lambda > 0\ \wedge\ x_1^0 = 0\ \wedge\ x_2^0 = 1\ \wedge\ u^2 \le 1\,\big]\Big]$$

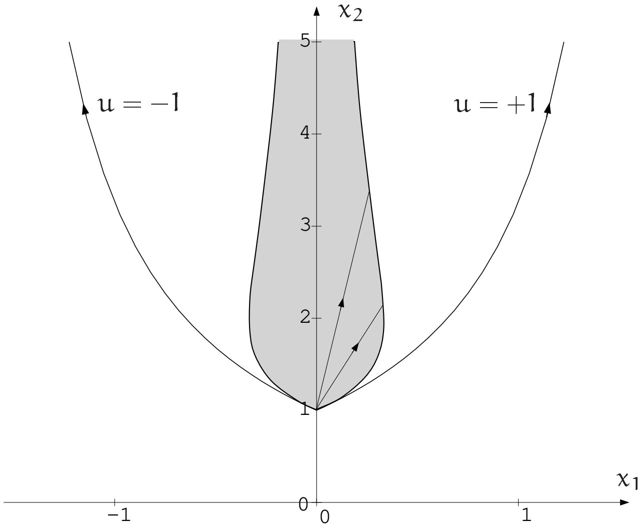


Figure 9.5 *The semialgebraic set defined by (9.11) (gray shaded region) and the reachable set of system (9.10) (the region above the solutions corresponding to* $u = +1$ *and* $u = -1$*).*

and eliminating quantifiers we get

$$\begin{aligned}\big[-x_1^1(x_2^1)^2 + x_1^1x_2^1 - x_2^1 - x_1^1 + 1 \le 0 \;\wedge \\ x_1^1(x_2^1)^2 - x_1^1x_2^1 - x_2^1 + x_1^1 + 1 \le 0\big],\end{aligned} \tag{9.11}$$

see Figure 9.5.
A control law that steers the system along a straight line can be computed as in Example 9.5 noting that C *with slope* k *is a part of the zero set of* $c(x) = x_2 - 1 - kx_1$*. The orthogonality condition (9.8) with the choice* $q(x) = 0$ *gives*

$$-k(x_1 + u) + x_2^2 = 0 \;\rightarrow\; u = \frac{1}{k}x_2^2 - x_1,\; k \ne 0.$$

The cases $k = 0$ *and* $k = \infty$ *cause no problems since the line* $x_2 = 1$ *does not belong to* $\mathcal{R}(\mathcal{I})$ *and for* $k = \infty$ *the control law simply becomes* $u = -x_1$*. Furthermore, this control law steers the system in the right direction.*

Once we know the set of reachable states from a point along straight lines an obvious generalization is to let this set be possible initial states of a new calculation of reachability. The resulting set would be an even better approximation to the real reachable set of states. Unfortunately, this calculation was too complex to be carried out by the QEPCAD program.

10

Conclusions and Extensions

In this part of the thesis we have formulated a number of problems in control theory as formulas in the first-order theory of real closed fields and then applied quantifier elimination to solve them. Many problems in control theory seem to fit into this framework and it is not hard to come up with other ideas worth investigation. An appealing property of this approach is the often close connection between the original problem formulation and the resulting formula.

It is also possible to take into account constraints on both the control and state variables in a direct manner which is not a common feature of other approaches.

Applications to Linear systems

We have singled out two problems in linear control theory to show the applicability of the quantifier elimination approach. First we addressed the problem of stabilization in presence of uncertainties. Given a suitable controller structure and a description of the uncertainties, we end up with a number of conditions on the controller parameters such that the system is stabilized regardless of the actual values of the uncertain parameters.

The second problem was feedback design. Here the objectives were to shape the return ratio of the closed-loop system by an appropriate choice of controller parameters, where model uncertainties also were taken into consideration. Specifications on the Nyquist curve of the return ratio were translated into formulas and quantifier elimination was performed to get conditions on the controller parameters.

A common approach to deal with model uncertainties in design of robust controllers and stability analysis is to use tools such as convex programming, linear matrix inequalities (LMIs) and μ-analysis [6, 10, 57]. When considering real parametric uncertainties the problems often become non-convex and most of these methods fail. The quantifier elimination method does not utilize any information about possible convexity of the problem and hence does not suffer from this drawback. Though, for convex problems the above approaches are known to be very efficient.

The conditions on the controller parameters in both problems are in general given by a system of equations and inequalities, or a union of such systems, produced by the quantifier elimination algorithm. If the controller parameters are bounded by existential quantifiers one may also get specific values on the controller parameters, i.e., get a numerical solution to the problem (a sample point from a cell in the CAD for which the formula is true).

Applications to Nonlinear systems

For nonlinear systems we have restricted the discussion to mainly two problems, computation of stationary points and curve following in the state space. Quantifier elimination provides a way of constructing a real polynomial system in the state variables, whose solution consists of those states that can be made stationary by an appropriate choice of available control signals, the so called stationarizable states. The subset of these stationarizable points that corresponds to asymptotically stable equilibrium points, can also be computed in a similar fashion.

Curve following in the state space gives an answer to the question if it is possible to steer the state between two points along a given curve using an admissible control. This is a kind of motion planning problem where not only the output but also the state of the system is specified at every time instant. We have also given a generalization of this problem where the curve to follow is to be chosen in a parameterized set of curves which starts in a set of initial states and ends in a set of final states.

If we do not put any restrictions on the set of final states and choose a flexible set of curves we can consider the curve following problem as a constructive method to compute global reachability from a given set of initial points. However, due to the many variables of this problem the complexity becomes *very* high.

General Remarks and Complexity

The method of quantifier elimination seems attractive due to its expressive power, e.g., we can easily incorporate control and state constraints. These kind of constraints are known to be very common in practice but hard to take into consideration in classical methods. The main problem is to pin point the specifications and objectives in terms of (semi-)algebraic relations. To write down the question or formula, i.e., the input to the solution algorithm is then often a minor problem.

The expressive power is not only a good thing. It implies that problems which are known to be very hard to solve should also be solvable by a general quantifier elimination algorithm. This means that any algorithm for quantifier elimination has to be complex, by its own nature. The bad time-complexity w.r.t. the number of variables is what at present limits its usefulness for large scale control applications.

Many classical problems in control theory can be formulated using the concepts of semialgebraic sets and formulas in the first-order theory of real closed fields. An important observation is that this implies that these problems can, at least theoretically, be solved after a finite number of steps.

Extensions

In principle nothing prevents us from working with systems given in implicit form, $f(\dot{x}, x, u) = 0$ or more general mixed state and control constraints, $\mathcal{UX}(x, u)$ but we have chosen a simpler setting to demonstrate the ideas.

The problem of simultaneous stabilization addresses the design of a controller that stabilizes a number of different linear systems simultaneously. An application of this can be design of simple controllers for hybrid systems, i.e., systems including switching devices that changes the system properties drastically. According to the methods in this part of the thesis the problem can be reduced to find a common solution of a number of real polynomial systems.

Verification of control laws and further stability analysis of nonlinear systems, e.g., by Lyapunov theory are other interesting subjects to study. Once the system, the controller, and the control objectives are defined in terms of (semi-)algebraic relations, the verification procedure can be described formally according to

$$\forall z \Big[[\text{control law}(z) \;\land\; \text{system description}(z)] \;\rightarrow\; [\text{system behavior}(z)] \Big],$$

i.e., if the above formula can be decided to be true the control law implies a certain behavior of the closed-loop system. In Lyapunov theory there are often a number of free real parameters to be determined such that a certain function becomes positive for a region in the state space which is as large as possible. This problem could also be treated by quantifier elimination.

One way to circumvent the problem of complexity of the quantifier elimination method can be to combine several approaches. This can be done first by trying to solve the linear part of the problem using linear programming techniques. Some of the non-linear constraints may then become linear because the values of certain coefficients were determined. The solver continues until no more simplifications can be done. This approach has been proposed in [20]. Now, either the problem is solved completely or we hope that the number of variables has been decreased substantially, to allow a solution by quantifier elimination in a reasonable amount of time.

Another way to deal with the complexity problem is to utilize the sparse structure of multivariate polynomials which seems to be a common property in many applications. Recent developments along these lines have been made by Canny [13].

If it is possible to decide if a part of a problem is convex one could use convex programming to solve the problem as far as possible. Quantifier elimination would then be a final step in the solution algorithm where many of the original variables already have got their values determined in the preceding convex programming step.

For many problems in control the quantifier elimination method is unnecessary general. It would be interesting to examine if it is possible to utilize some of the subalgorithms of the quantifier elimination method for some specific control problems and in this way reduce the complexity.

Bibliography

[1] W. W. Adams and P. Loustaunau. *An Introduction to Gröbner Bases*, volume 3 of *Graduate Studies in Mathematics*. American Mathematical Society, 1994.

[2] B. Anderson, N. Bose, and E. Jury. Output feedback stabilization and related problems - solution via decision methods. *IEEE Trans. Autom. Control,*, AC(20):53–65, 1975.

[3] D.S. Arnon. A bibliography of quantifier elimination for real closed fields. *J. Symbolic Comput.*, 5(1-2):267–274, Feb-Apr 1988.

[4] D.S. Arnon, G.E. Collins, and S. McCallum. Cylindrical algebraic decomposition I: The basic algorithm. *SIAM J. Comput.*, 13(4):865–877, November 1984.

[5] S. Basu, R. Pollack, and M.-F. Roy. On the combinatorial and algebraic complexity of quantifier elimination. In S. Goldwasser, editor, *Proc. 35th Annual Symposium on Foundations of Computer Science*, pages 632–641, Santa Fe, NM, USA, 1994.

[6] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty. *Nonlinear Programming - Theory and Algorithms*. Wiley, 1993.

[7] R. Benedetti and J. Risler. *Real Algebraic and Semi-Algebraic Sets*. Hermann, Paris, 1990.

[8] V. D. Blondel and J. N. Tsitsiklis. Np-hardness of some linear control design problems. In A. Isidori, S. Bittanti, E. Mosca, A. De Luca, M. D. Di Benedetto,

and G. Oriolo, editors, *ECC'95, European Control Conference*, volume 3, pages 2066–2071, Roma, Italy, 1995.

[9] J. Bochnak, M. Coste, and M-F. Roy. *Géometrie algébrique réelle*. Springer, 1987.

[10] S. Boyd, L.E. Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM Studies in Applied Mathematics. SIAM, Philadelphia, Pennsylvania, 1994.

[11] A. E. Bryson and Y. C. Ho. *Applied Optimal Control*. Ginn and Company, 1969.

[12] B. Buchberger, H. Hong, R. Loos, G. Collins, J. Johnson, A. Mandache, M. Encarnacion, W. Krandick, A. Neubacher, and H. Vielhaber. Saclib 1.1 user's guide. Technical Report 93-19, RISC, Johannes Kepler University, A-4040 Linz, Austria, 1993.

[13] J. Canny. A toolkit for non-linear algebra. Technical report, Computer Science Division, University of California, Berkeley, CA, 94720, August 1993.

[14] B. Char, K.O. Geddes, G.H. Gonnet, B.L. Leong, M.B. Monagan, and S.M. Watt. *Maple V Language Reference Manual*. Springer, 1991.

[15] B. Char, K.O. Geddes, G.H. Gonnet, B.L. Leong, M.B. Monagan, and S.M. Watt. *Maple V Library Reference Manual*. Springer, 1991.

[16] R.V. Churchill and J.W. Brown. *Complex Variables and Applications*. Mathematics Series. McGraw-Hill, fourth edition, 1984.

[17] H. Cohen, editor. *A Course in Computational Algebraic Number Theory*. Springer, 1993.

[18] G.E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Second GI Conf. Automata Theory and Formal Languages, Kaiserslauten*, volume 33 of *Lecture Notes Comp. Sci.*, pages 134–183. Springer, 1975.

[19] G.E. Collins and H. Hong. Partial cylindrical algebraic decomposition for quantifier elimination. *J. Symbolic Comput.*, 12:299–328, Sep 1991.

[20] A. Colmerauer. Naive solving of non-linear constraints. In A. Colmerauer and Frédéric Benhamou, editors, *Constraint Logic Programming - Selected Research*, chapter 6, pages 89–112. The MIT Press, 1993.

[21] M. Coste, L. Mahé, and M.F. Roy, editors. *Real Algebraic Geometry*, volume 1524 of *Lecture Notes in Mathematics*. Springer, 1992. Proceedings of the Conference held in Rennes, France, June 24-28, 1991.

[22] M. Coste and M.F. Roy. Thom's lemma, the coding of real algebraic numbers and the computation of the topology of semi-algebraic sets. *J. Symbolic Computation*, 5:121–129, 1988.

[23] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra.* Undergraduate Texts in Mathematics. Springer, 1992.

[24] J.H. Davenport, Y. Siret, and E. Tournier. *Computer Algebra. Systems and Algorithms for Algebraic Computation.* Academic Press, 1988.

[25] S. Diop. Elimination in control theory. *Math. Control Signals Systems*, 4(1):17–32, 1991.

[26] G. Carrà Ferro. Gröbner bases and differential algebra. In L. Huguet and A. Poli, editors, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, volume 356 of *Lecture Notes Comp. Sci.*, pages 129–140. Springer, 1989. Proc. AAECC-5, Menorca.

[27] M. Fliess and S.T. Glad. An algebraic approach to linear and nonlinear control. In H.L. Trentelman and J.C. Willems, editors, *Essays on Control: Perspectives in the Theory and its Applications*, pages 223–267. Birkhäuser, Boston, 1993.

[28] K. Forsman. *Constructive Commutative Algebra in Nonlinear Control Theory.* PhD thesis, Dept. of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden, 1991.

[29] K. Forsman and M. Jirstrand. Some finiteness issues in differential algebraic systems theory. In *Proc. 33th Conf. on Decision and Control*, volume 2, pages 1112–1117, Lake Bueno Vista, FL, December 1994. IEEE.

[30] F.R. Gantmacher. *Matrix Theory*, volume II. Chelsea, 1960.

[31] R. Germundsson. *Symbolic Systems – Theory, Computation and Applications.* PhD thesis, Dept. of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden, 1995.

[32] R. Germundsson and K. Forsman. A constructive approach to algebraic observability. In *Proc. 30th Conf. on Decision and Control*, Brighton, UK, 1991. IEEE.

[33] S.T. Glad. Nonlinear state space and input output descriptions using differential polynomials. In J. Descusse, M. Fliess, A. Isidori, and D. Leborgne, editors, *New Trends in Nonlinear Control Theory*, volume 122 of *Lecture Notes in Control and Information Sciences*, pages 182–189. Springer, 1988. Proc. Intl. Conf. Nonlinear Systems, Nantes, France, June 13-17, 1988.

[34] S.T. Glad. Nonlinear regulators and Ritt's remainder algorithm. In *Colloque international sur l'analyse des systèmes dynamiques controllés*, volume 2, July 1990. Lyon, France.

[35] S.T. Glad. Implementing Ritt's algorithm of differential algebra. In M. Fliess, editor, *Proc. NOLCOS'92*, pages 610–614, Bordeaux, France, 1992. IFAC.

[36] S.T. Glad. Algebraic Methods in Systems Theory - Differential Algebra. Technical report, Dept. of Electrical Engineering, Linköping University, 1996.

[37] S.T. Glad and L. Ljung. Model structure identifiability and persistence of excitation. In *Proc. 29th Conf. on Decision and Control*, pages 1069–1073, Honululu, Hawaii, December 1990. IEEE.

[38] J. Gunnarsson. *On Modeling of Discrete Event Dynamical Systems, using Symbolic Algebraic Methods*. Licentiate thesis, Dept. of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden, 1995.

[39] W. Hahn. *Stability of Motion*. Springer, 1967.

[40] J. Heintz, M.-F. Roy, and P. Solerno. On the theoretical and practical complexity of the existential theory of reals. *The Computer Journal*, 36(5):427–431, 1993.

[41] J. Hollman and L. Langemyr. Algorithms for non-linear algebraic constraints. In F. Benhamou and A. Colmerauer, editors, *Constraint Logic Programming - Selected Research*, pages 113–131. The MIT Press, 1993.

[42] H. Hong. *Improvements in CAD-based Quantifier Elimination*. PhD thesis, The Ohio State University, 1992.

[43] H. Hong. Simple solution formula construction in cylindrical algebraic decomposition based quantifier elimination. In P. S. Wang, editor, *ISAAC'92, International Symposium on Symbolic and Algebraic Computation*, pages 177–188, Berkely, California, 1992b. ACM Press.

[44] A. Hurwitz. Über die bedingungen unter welchen eine gleichung nur wurzeln mit negativen reellen teilen besitzt. *Matematische Annalen*, 46:273–284, 1895.

[45] N. Jacobson. *Basic Algebra*, volume 1. Freeman, New York, second edition, 1985.

[46] N. Jacobson. *Basic Algebra*, volume 2. Freeman, New York, second edition, 1985.

[47] I. Kaplansky. *An Introduction to Differential Algebra*. Hermann, 1957.

[48] M. Knebusch and C. Schneiderer. *Einführung in die reelle Algebra*. Vieweg, Braunschweig, 1989.

[49] D.E. Knuth. *The Art of Computer Programming. Vol. 2: Seminumerical Algorithms*. Addison-Wesley, second edition, 1981.

[50] E.R. Kolchin. *Differential Algebra and Algebraic Groups.*, volume 54 of *Pure and Applied Mathematics*. Academic Press, 1973.

[51] T.Y. Lam. An introduction to real algebra. *Rocky Mountain J. of Math.*, 14(4):767–814, Fall 84 1984.

[52] S. Lang. *Algebra.* Addison-Wesley, second edition, 1984.

[53] G. Leitmann. *The Calculus of Variations and Optimal Control - An Introduction.* Plenum Press, New York, 1981.

[54] H. Levi. On the structure of differential polynomials and on their theory of ideals. *Trans. Amer. Math. Soc.*, 51:532–568, 1942.

[55] P. Lindskog. *Algorithms and Tools for System Identification Using Prior Knowledge.* Licentiate thesis, Dept. of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden, 1994.

[56] R. Loos. Generalized polynomial remainder sequences. In B. Buchberger, G.E. Collins, and R. Loos, editors, *Computer Algebra. Symbolic and Algebraic Computation*, pages 115–137. Springer, second edition, 1983.

[57] J.M. Maciejowski. *Multivariable Feedback Design.* Electronic Systems Engineering Series. Addison-Wesley, 1989.

[58] E. Mansfield. *Differential Gröbner Bases.* PhD thesis, University of Sidney, 1991.

[59] E.L. Mansfield. Diffgrob2: A symbolic algebra package for analysing systems of PDE using Maple. Technical report, Department of Mathematics, University of Exeter, Exeter, U.K., 1993.

[60] H. Matsumura. *Commutative Ring Theory.*, volume 8 of *Cambridge Studies in Advanced Mathematics.* Cambridge University Press, 1986.

[61] S. McCallum. Solving polynomial strict inequalities using cylindrical algebraic decomposition. *The Computer Journal*, 36(5):432–438, 1993.

[62] D.G. Mead. Differential ideals. *Proc. Amer. Math. Soc.*, 6:430–432, June 1955.

[63] J. Michalik and J.C. Willems. The elimination problem in differential algebraic systems. In *Proc. MTNS*, Regensburg, Germany, 1993.

[64] B. Mishra. *Algorithmic Algebra.* Texts and Monographs in Computer Science. Springer-Verlag, 1993.

[65] K.B. O'Keefe. A property of the differential ideal $[\mathfrak{y}^{\mathfrak{p}}]$. *Trans. Amer. Math. Soc.*, 94:483–497, March 1960.

[66] F. Ollivier. Standard bases of differential ideals. In S. Sakata, editor, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, volume 508 of *Lecture Notes Comp. Sci.*, pages 304–321. Springer, 1990. Proc. AAECC-8, Tokyo.

[67] F. Ollivier. Generalized standard bases with applications to control. In *Proc. First European Control Conf.*, volume 1, pages 170–176, Grenoble, France, July 1991. Hermès.

[68] P. C. Parks and V. Hahn. *Stability Theory.* Prentice Hall international series in systems and control engineering. Prentice Hall, 1993.

[69] J. Plantin. *Algebraic Methods for Verification and Control of Discrete Event Dynamic Systems.* Licentiate thesis, Dept. of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden, 1995.

[70] J.F. Pommaret. *Differential Galois Theory.* Gordon and Breach, 1983.

[71] J. Renegar. On the computational complexity and geometry of the first-order theory of the reals. parts i–iii. *J. Symbolic Computation*, 13:255–352, 1992.

[72] J.F. Ritt. *Differential Algebra.* Dover, 1950.

[73] E.J. Routh. Stability of a given state of motion. In A.T. Fuller, editor, *Stability of Motion - A collection of early scientific papers by Routh, Clifford, Sturm and Bôcher.* Taylor and Francis, London, 1877.

[74] L.A. Rubel and M.F. Singer. A differentially algebraic elimination theorem with applications to analog computability in the calculus of variations. In *Proc.Amer.Math.Soc.*, volume 94, pages 653–658. Springer, 1985.

[75] R.Y. Sharp. *Steps in Commutative Algebra*, volume 19 of *London Mathematical Society Student Texts.* Cambridge University Press, 1990.

[76] B. L. Stevens and F. L. Lewis. *Aircraft Control and Simulation.* Wiley, 1992.

[77] A.W. Strzeboński. An algorithm for systems of strong polynomial inequalities. *The Mathematica J.*, 4(4):74–77, 1994.

[78] V. Syrmos, C. T. Abdallah, P. Dorato, and K. Grigoriadis. Static output feedback: a survey. *Submitted to: Automatica*, 1995.

[79] A. Tarski. *A Decision Method for Elementary Algebra and Geometry.* University of California Press, second edition, 1948.

[80] D. van Dalen. *Logic and Structure.* Springer-Verlag, second edition, 1980.

[81] M. Vidyasagar. *Nonlinear Systems Analysis.* Prentice Hall, second edition, 1993.

[82] J. C. Willems. A new setting for differential algebraic systems. In A. Isidori, S. Bittanti, E. Mosca, A. De Luca, M. D. Di Benedetto, and G. Oriolo, editors, *ECC'95, European Control Conference*, volume 1, pages 218–223, Roma, Italy, 1995.

[83] J.C. Willems. Paradigms and puzzles in the theory of dynamical systems. *IEEE Trans. Aut. Contr.*, AC-36(3):259–294, March 1991.

[84] S. Wolfram. *Mathematica. A System for Doing Mathematics by Computer.* Addison-Wesley, second edition, 1991.

Subject Index