Doctoral Thesis

Comprehensive Gröbner Bases in Various Domains

Katsusuke Nabeshima

April, 2007

Research Institute for Symbolic Computation
Johannes Kepler Universität Linz

Eidesstattliche Erklärung

Ich erkläre an Eides statt, daßich die vorliegende Dissertation selbstständig und ohne fremde Hilfe verfaßt, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörthlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Katsusuke Nabeshima
Hagenberg, April 2007

Zusammenfassung

Diese Arbeit behandelt Algorithmen zur Berechnung von parametrischen Gröbnerbasen und parametrischen Gröbnersystemen in verschiedenen algebraischen Strukturen: in kommutativen Ringen, in Ringen von Differentialoperatoren, in Polynomringen über kommutativen von-Neumann-regulären Ringen und über Moduln. Sowohl bezüglich Zeit- als auch bezüglich Speicherkomplexität ist unser neuer Algorithmus effizienter als andere existierende Algorithmen.

Wir definieren reduzierte Gröbnerbasen für Polynomringe über Polynomringe und führen Algorithmen zu ihrer Berechnung ein. Es sind bereits Algorithmen zur Berechnung von Gröbnerbasen in Polynomringen über Polynomringen angegeben worden. Diese liefern jedoch keine reduzierten Gröbnerbasen. Wir schlagen eine neue Definition für reduzierte Gröbnerbasen in diesen Ringen vor.

Algorithmen zur Berechnung von Gröbnerbasen in Ringen von Differentialoperatoren mit polynomiellen Koeffizienten sind bereits vorgeschlagen worden. In der vorliegenden Arbeit geben wir einen viel effizienteren und einfacheren Algorithmus für diese Situation an, der auf der Beziehung zweier Arten von Gröbnerbasen in Ringen von Differentialoperatoren beruht. Weiters geben wir Algorithmen zur Berechnen von parametrischen Gröbnerbasen in diesen Ringen an. Das heißt, wir beschreiben nicht-kommutative parametrische Gröbnerbasen in Ringen von Differentialoperatoren.

Verschiedene Algorithmen zur Berechnung parametrischer Gröbnerbasen in Polynomringen sind bekannt. Jedoch wurde die Erweiterung von parametrischen Gröbnerbasen auf Moduln noch nicht untersucht. Wir verallgemeinern die Theorie der parametrischen Gröbnerbasen auf Moduln.

Teil dieser Arbeit ist eine Implementierung der vorgestellten Algorithmen in Form eines Software-Pakets für das Computeralgebrasystem Risa/Asir.

Stichwörter Parametrische Gröbnerbasen, Weyl-Algebra, Moduln.

Abstract

This thesis presents algorithms for computing comprehensive Gröbner bases and comprehensive Gröbner systems in various domains: commutative polynomial rings, rings of differential operators, polynomial rings over a commutative von Neumann regular ring and modules. In both space and time complexity, our new algorithm is much more efficient than other existing algorithms.

We define reduced Gröbner bases for polynomial rings over a polynomial ring, and introduce algorithms for computing them. There exist some algorithms for computing Gröbner bases in polynomial rings over a polynomial ring. However, we cannot obtain the reduced Gröbner bases by the algorithms in these rings. We propose a new notion of reduced Gröbner bases in these rings.

Algorithms for computing Gröbner bases in rings of differential operators with coefficients in a polynomial ring, have been proposed in the literature. In this thesis, we present a much more efficient and simpler algorithm than these algorithms by using the relations of two kinds of Gröbner bases in rings of differential operators. Moreover, we introduce algorithms for computing their comprehensive Gröbner bases. Namely, we describe non-commutative comprehensive Gröbner bases in rings of differential operators.

Several algorithms are known for computing comprehensive Gröbner bases in polynomial rings. However, the extension of comprehensive Gröbner bases to modules has not been studied yet. We generalize the theory of comprehensive Gröbner bases to the modules.

Part of the thesis is an implementation of the presented algorithms in form of a software package for the computer algebra system Risa/Asir.

Keywords comprehensive Gröbner bases, Weyl algebra, modules.

Acknowledgements

First of all, I am grateful to my thesis adviser Prof. Franz Winkler for giving me the opportunity to study computer algebra at RISC-Linz and for his support.

Special thanks go to my neighbor Dr. Manual Kauers. He translated a lot of German documents into English ones for me, taught me some funny German words and their concepts, played “Othello (which is a Japanese game)” with me (actually a lot), and of course discussed mathematics with me.

I thank Tsuyako and Johann Gutenbrunner for their support. They have assisted my Austrian life.

I wish to thank the Computer Algebra group at RISC-Linz, especially Dr. Ralf Hemmecke and Dr. Günter Landsmann, for all comments and suggestions.

Furthermore, I would like to thank all the people at RISC-Linz for creating a great scientific and personal environment.

Finally, I would like to thank my parents Tomi and Takayuki Nabeshima who supported me during all the years of my studies.

This work has been partially supported by the Austrian science foundation (FWF) project P16357-N04, and the SFB F1301.

Contents

Chapter 1	Introduction	1
Chapter 2	Preliminaries	5
2.1	Ideals, varieties and term orders	5
2.2	Gröbner bases	8
2.3	Syzygies and Gröbner bases for modules	11
Chapter 3	Gröbner bases in polynomial rings over a polynomial ring	17
3.1	Notations for $K[\bar{A}, \bar{X}]$ and $K[\bar{A}][\bar{X}]$	17
3.2	Approach by Insa and Pauer	18
3.3	Approach via block orders	22
3.4	Problems	23
3.5	Reduced Gröbner bases	24
3.6	Computation examples	31
Chapter 4	Comprehensive Gröbner bases and comprehensive Gröbner systems	33
4.1	Introduction	33
4.2	Comprehensive Gröbner bases and comprehensive Gröbner systems	34
4.3	Stability of ideals	38
4.4	Suzuki-Sato's algorithm	40
4.5	Software	46
Chapter 5	A new algorithm for computing comprehensive Gröbner systems	51
5.1	Motivation	51
5.2	A new algorithm	54
5.3	Optimization techniques	61
5.4	Benchmark tests and improvements	62
Chapter 6	Comprehensive Gröbner bases and von Neumann regular rings	65
6.1	Von Neumann regular rings and Boolean algebra	65
6.2	Gröbner bases over von Neumann regular rings	68
6.3	Criteria for computing Gröbner bases	72
6.4	Alternative comprehensive Gröbner bases	75
6.5	ACGB on varieties (ACGB-V)	80
6.6	Computation methods for ACGB-V	81
6.7	A direct products of fields approach to comprehensive Gröbner bases over finite fields	86
Chapter 7	Two kinds of Gröbner bases in rings of differential operators	93
7.1	Notations	93
7.2	Gröbner bases in $K[\bar{X}, \bar{D}]$	95
7.3	Approach by Insa and Pauer for computing Gröbner bases in $K[\bar{X}][\bar{D}]$	97

7.4	Approach via block orders for computing Gröbner bases in $K[\bar{X}][\bar{D}]$	100
7.5	Reduced Gröbner bases in rings over a polynomial ring	102
Chapter 8	**Comprehensive Gröbner bases in rings of differential operators**	**105**
8.1	Comprehensive Gröbner bases in $K[\bar{A}][\bar{X},\bar{D}]$	105
8.2	Comprehensive Gröbner bases in $(K[\bar{A}][\bar{X}])[\bar{D}]$	118
8.3	Other approaches	120
8.4	Applications	121
Chapter 9	**Comprehensive Gröbner bases for modules**	**123**
9.1	Notations	123
9.2	Comprehensive Gröbner systems for modules	124
9.3	Comprehensive Gröbner bases for modules	132
9.4	Applications	136
9.5	Reduced Gröbner bases in $K[\bar{A}][\bar{X}]^r$	142
9.6	Concluding Remarks	145
Chapter 10	**Implementation: PGB package**	**147**
10.1	CGBs and CGSs in commutative polynomial rings	147
10.2	CGBs and CGSs in rings of differential operators	156
10.3	CGBs and CGSs for modules	159
10.4	Related objects	164
10.5	Concluding remarks	167
Bibliography		**169**
Index		**175**
Curriculum Vitae		**179**

Chapter 1

Introduction

I know nothing except the fact of my ignorance.
Socrates (470 BC- 399 BC)

Nowadays it is common knowledge that Gröbner bases and Buchberger's algorithm are key ingredients in computational commutative algebra, and are hence fundamental tools for applications in several fields both inside and outside mathematics (see [Buc85, BW98]). For every concept and construction in computer algebra the question of uniformity in the input parameters is crucial both from a theoretical and a practical viewpoint. This applies in particular to the concept of Gröbner bases.

Parametric polynomial systems have been studied by many researchers. Mainly, we have two kinds of parametric polynomials
(1) polynomials with parametric **coefficients**, and
(2) polynomials with parametric **exponents**.
Recently, Gröbner bases for parametric polynomials have been actively investigated. For example, one can see several papers for Gröbner bases of a polynomial ideal with parametric coefficients in [Mon02, MM06, Wei92, SS02, SS06, Wei06], and for Gröbner bases of a polynomial ideal with parametric exponents in [Yok04, Yok07, Wan05, PW06, Wei04].

In this thesis, we treat the theory of Gröbner bases for a polynomial ideal with parametric coefficients, i.e., the case (1). In general, these Gröbner bases are called **"comprehensive Gröbner bases"**. Comprehensive Gröbner bases for parametric ideals were introduced, constructed, and studied by Weispfenning [Wei92] in 1992. Since then comprehensive Gröbner bases were studied by several researchers and implemented in several computer algebra systems. A comprehensive Gröbner basis is a finite subset G of a parametric polynomial ideal I such that $\sigma(I)$ constitutes a Gröbner basis of the ideal generated by $\sigma(F)$ under all specialization σ of the parameters. As we said above, the theory of Gröbner bases is a fundamental tool in several fields. Therefore by studying comprehensive Gröbner bases in various domains, we are able to solve a lot of parametric problems in these domains. That is, the number of applications of comprehensive Gröbner bases is large and manifold; it comprises most basic parametric problems in polynomial algebra and algebraic geometry that have been inaccessible to Gröbner basis methods so far [Wei92]. For example:

1. Parametric ideal membership and parametric modules of syzygies.
2. The study of parametric varieties, their size and their dimension functions; in particular one-parameter varieties and algebraic bifurcation problems.

In this thesis, we give new algorithms for computing parametric Gröbner bases, and generalize the theory of parametric Gröbner bases to various domains. The plan of this thesis is as follows:

We give basic concepts used in this thesis in chapter 2.

In chapter 3 we describe algorithms for computing Gröbner bases in polynomial rings over a polynomial ring. These algorithms are applied for constructing algorithms for computing parametric Gröbner bases. Moreover, we define reduced Gröbner bases in polynomial rings over a polynomial ring and introduce algorithms for computing them. That is, we propose a new notion of reduced Gröbner bases in polynomial rings over a polynomial ring and we show that every ideal has a unique reduced Gröbner basis.

Chapter 4 presents comprehensive Gröbner bases and comprehensive Gröbner systems. Furthermore we describe the history and recent trend. In this chapter we introduce the Suzuki-Sato algorithms for computing comprehensive Gröbner bases and comprehensive Gröbner systems, because these algorithms are faster then other existing algorithms. We also introduce the existing software for computing comprehensive Gröbner bases and comprehensive Gröbner systems.

Chapter 5 treats a new algorithm for computing comprehensive Gröbner systems. Roughly speaking, a comprehensive Gröbner system is a parametric Gröbner basis with parameter spaces. If we take a parameter space $\mathbb{P}$ and its set of parametric polynomials G from a comprehensive Gröbner systems for a parametric polynomial ideal I, then $\sigma(G)$ constitutes a Gröbner basis of the ideal generated by $\sigma(I)$ under the specialization σ with respect to the parameter space $\mathbb{P}$ of the parameters. Comprehensive Gröbner systems are also important ingredients for solving problems of parametric polynomials. Actually, our algorithm is much faster and more efficient than other existing algorithm.

Chapter 6 describes the relations between comprehensive Gröbner bases and non-parametric Gröbner bases over commutative von Neumann regular rings. Actually, there is a surprisingly close relationship between them. Thus, the Gröbner bases over a commutative von Neumann regular ring can be viewed as an alternative to comprehensive Gröbner bases. (Therefore, this Gröbner basis is called an "alternative comprehensive Gröbner basis (ACGB)".) In the second part of the chapter, we describe the special type of comprehensive Gröbner bases which is called alternative comprehensive Gröbner bases on varieties (ACGB-V).

In chapter 7 and chapter 8, we present relations of two kinds of Gröbner bases in rings of differential operators, and algorithms for computing their comprehensive Gröbner bases and comprehensive Gröbner systems. In 1998, Insa and Pauer studied an algorithm for computing Gröbner bases in rings of differential operators with coefficients in a polynomial ring. This algorithm is very complicated and expensive. In chapter 7, we present a much more efficient and simple algorithm than the Insa-Pauer algorithm by using the relations of two kinds of Gröbner bases in rings of differential operators. Moreover, in chapter 8, we introduce algorithms for computing their comprehensive Gröbner bases and comprehensive Gröbner systems. Namely, we describe non-commutative comprehensive Gröbner bases in rings of differential operators.

In chapter 9, we present an algorithm for computing comprehensive Gröbner bases and comprehensive Gröbner systems for modules. Several algorithms are known for computing comprehensive Gröbner bases in polynomial rings. However, nobody has studied the extension of comprehensive Gröbner bases to modules. We present a generalization of the Suzuki-Sato algorithm.

In chapter 10, we describe our software package sf PGB in which almost all the algorithms of this thesis are implemented. The purpose of this chapter is to illustrate how to use the package and solve problems using the package in the computer algebra system Risa/Asir.

In summary, the main achievements of this thesis are:

- defining reduced Gröbner bases in polynomial rings over a polynomial ring,
- presenting a new algorithm for computing comprehensive Gröbner systems which is more efficient than other existing algorithms,
- presenting a computation method for ACGB-V,
- presenting an efficient algorithm for computing Gröbner bases in rings of differential operators with coefficients in a polynomial ring,
- extending the theory of comprehensive Gröbner bases to the rings of differential operators and modules, and
- developing the software package PGB for computing parametric Gröbner bases in various domains.

Chapter 2

Preliminaries

This chapter contains a collection of well-known definitions and facts, which are needed later. Its main purpose is to fix the notation and provide labels for certain general theorems. The advanced reader may securely skip this chapter and proceed directly to Chapter 3.

Throughout this text, we assume that K and L are fields such that L is an extension of K. If K and L appear in the description of an algorithm, it is also assumed that K and L are computable. $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ and $\mathbb{C}$ denote as the set of natural numbers, the set of integers, the field of rational numbers, the field of real numbers and the field of complex numbers, respectively. Note that in this thesis, the set of natural number $\mathbb{N}$ includes zero 0.

2.1 Ideals, varieties and term orders

This section presents some well-known basic notions. It is sufficient for most of this text if the reader is familiar with the most basic facts about commutative algebra as they are presented, for instance, in the introductory book of Cox *et al.* [CLO92].

We will need to discuss polynomials in n variables $x_1, \dots, x_n$ with coefficients in an arbitrary field K. We start by defining power products and polynomials.

Definition 2.1.1. Let $x_1, \dots, x_n$ be n variables.

1. A **power product** (or **term**) in $x_1, \dots, x_n$ is a an expression of the form $X^\alpha := x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ for some exponent vector $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}^n$. When $\alpha = (0, \dots, 0)$ note that $X^\alpha = 1$. The **total degree** of this power product (written: $\deg(X^\alpha)$) is the sum $\alpha_1 + \cdots + \alpha_n$. The set of all power products in $x_1, \dots, x_n$ is denoted by $\mathrm{pp}(x_1, \dots, x_n)$.
2. A **polynomial** f in $x_1, \dots, x_n$ with coefficients in K is a finite linear combination (with coefficients in K) of power products. We write a polynomial f in the form
$$f = a_1 X^{\alpha_1} + \cdots + a_r X^{\alpha_r},$$
where $a_1, \dots, a_r \in K$ and pairwise different $\alpha_1, \dots, \alpha_r \in \mathbb{N}^n$. The **total degree** of f, denoted $\deg(f)$, is the $\max_{i=1}^r \deg(X^{\alpha_i})$ such that the coefficient a_i is non-zero.
3. The set of all polynomials in $x_1, \dots, x_n$ with coefficients in K is denoted $K[x_1, \dots, x_n]$.

Definition 2.1.2. Let $f = \sum_\alpha a_\alpha X^\alpha$ be a polynomial in $K[x_1, \dots, x_n]$.

1. We call a_α the **coefficient** of the power product X^α.
2. If $a_\alpha \neq 0$, then we call $a_\alpha X^\alpha$ a **monomial** of f.

Hereafter the notation $\bar{X}$ will be used as an abbreviation for the set of n variables $\{x_1, \ldots, x_n\}$. (I.e., $K[\bar{X}] := K[x_1, \ldots, x_n]$.) The notation $\mathrm{pp}(\bar{X})$ denotes **the set of power products** of $\bar{X}$. We define the basic geometric and algebraic object of this thesis.

Definition 2.1.3. Let $f_1, \ldots, f_s$ be polynomials in $K[\bar{X}]$. Then we set

$$\mathbb{V}(f_1, \ldots, f_s) = \{(a_1, \ldots, a_n) \in K^n \mid f_i(a_1, \ldots, a_n) = 0 \text{ for all } 1 \le i \le s\}.$$

We call $\mathbb{V}(f_1, \ldots, f_s)$ the **affine variety** defined by $f_1, \ldots, f_s$ over K.

Definition 2.1.4. Let R be a commutative ring. A set $I \subseteq R$ is called an **ideal** in R if it satisfies:

1. If $f, g \in I$, then $f + g \in I$.
2. If $f \in I$ and $h \in R$, then $hf \in I$.

Let $f_1, \ldots, f_s \in R$. Then we set

$$\langle f_1, \ldots, f_s \rangle = \left\{ \sum_{i=1}^{s} h_i f_i \,\middle|\, h_1, \ldots, h_s \in R \right\}.$$

The crucial fact is that $\langle f_1, \ldots, f_s \rangle$ is an ideal in R. We call $\langle f_1, \ldots, f_s \rangle$ the ideal generated by $f_1, \ldots, f_s$.

Definition 2.1.5. An ideal I is **radical** if $f^m \in I$ for some integer $m \ge 1$ implies that $f \in I$.

Definition 2.1.6. Let R be a commutative ring and $I \subset R$ an ideal. The **radical** of I, denoted $\mathrm{rad}(I)$, is the set

$$\{f \mid f^m \in I \text{ for some integer } m \ge 1\}.$$

Definition 2.1.7. A total (linear) order $\succ$ on the set of power products $\mathrm{pp}(\bar{X})$ is called a **term order** (also called an **admissible order**) if and only if it satisfies the following two additional properties.

1. $t \succ 1$ for all $t \in \mathrm{pp}(\bar{X}) \backslash \{1\}$, and
2. $t_1 \succ t_2 \Longrightarrow t_1 s \succ t_2 s$ for all $t_1, t_2, s \in \mathrm{pp}(\bar{X})$.

Below are some examples of term orders. Some of them are the most frequently used in the literature and in computer algebra systems.

Example 2.1.8. The commonly used term orders are defined as follows for arbitrary power products (or terms) $t_1 = x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ and $t_2 = x_1^{\beta_1} \cdot x_2^{\beta_2} \cdots x_n^{\beta_n}$.

1. The **lexicographic order** $\succ_{lex}$.

$$t_1 \succ_{lex} t_2 :\Longleftrightarrow \exists 1 \le k \le n \text{ such that } \alpha_i = \beta_i \text{ for all } 1 \le i < k \text{ and } \alpha_k > \beta_k.$$

2. The **reverse lexicographic order** $\succ_{rlex}$.

$$t_1 \succ_{rlex} t_2 :\Longleftrightarrow \exists 1 \le k \le n \text{ such that } \alpha_i = \beta_i \text{ for all } k < i \le n \text{ and } \alpha_k < \beta_k.$$

3. The **graded lexicographic order** $\succ_{glex}$.

$$t_1 \succ_{glex} t_2 :\Longleftrightarrow \deg(t_1) > \deg(t_2) \text{ or } (\deg(t_1) = \deg(t_2) \text{ and } t_1 \succ_{lex} t_2).$$

4. The **graded reverse lexicographic order** $\succ_{grlex}$.

$$t_1 \succ_{grlex} t_2 :\iff \deg(t_1) > \deg(t_2) \text{ or } (\deg(t_1) = \deg(t_2) \text{ and } t_1 \succ_{rlex} t_2).$$

Throughout this text, we will use the notations $\succ_{lex}$, $\succ_{rlex}$, $\succ_{glex}$ and $\succ_{grlex}$ for these orders. In this text, we often use the following special term order "block order". Furthermore, in chapter 10, we will see matrix orders and weight orders. Therefore, we introduce these special orders in the following.

Definition 2.1.9 (Block orders). Let $\succ_1$ and $\succ_2$ be term orders on $\mathrm{pp}(\bar{X})$ and $\mathrm{pp}(\bar{A})$, respectively, and $t_1, s_1 \in \mathrm{pp}(\bar{X})$, $t_2, s_2 \in \mathrm{pp}(\bar{A})$,

$$t_1 t_2 \succ_{\bar{X},\bar{A}} s_1 s_2 \iff t_1 \succ_1 s_1 \text{ or } (t_1 = s_1, \text{ and } t_2 \succ_2 s_2).$$

This type of order $\succ_{\bar{X},\bar{A}}$ is called a **block order** on $\mathrm{pp}(\bar{X}, \bar{A})$. This order is written as $\succ_{\bar{X},\bar{A}} := (\succ_1, \succ_2)$.

Definition 2.1.10 (Matrix orders). Let A be a matrix in $\mathrm{GL}(n, \mathbb{R})$. The matrix M satisfies the following

1. $M \cdot v = 0 \iff v = 0$, where $v \in \mathbb{N}^n$ and $\cdot$ denotes the usual matrix-by-vector product,
2. the first non-zero coordinate of the vector $M \cdot v$ is positive for all $v \in \mathbb{N}^n$.

Then, for $u, v \in \mathbb{N}^n$,

$$u \succ_M v :\iff \text{the first non-zero coordinate of the vector } M(u - v) \text{ is positive.}$$

This order is a term order. We call the order $\succ_M$ **matrix order**.

Definition 2.1.11 (Weight order). Let $\boldsymbol{u} = (u_1, \ldots, u_n)$ be a vector in $\mathbb{R}^n$. We say that $\boldsymbol{u}$ is a weight vector. Then, for $\alpha, \beta \in \mathbb{N}^n$, define

$$\alpha \succ_{\boldsymbol{u}} \beta \iff \boldsymbol{u} \cdot \alpha > \boldsymbol{u} \cdot \beta$$

where the centered dot is the usual dot product of vector. We call $\succ_{\boldsymbol{u}}$ the **weight order** determined by $\boldsymbol{u}$.
Note that in this thesis, we treat only "term orders (admissible orders)". If we have a vector $\boldsymbol{u} = (u_1, \ldots, u_n)$ such that some of $\{u_1, \ldots, u_n\}$ are negative, then this order $\succ_{\boldsymbol{u}}$ does not hold Definition 2.1.7. Therefore, in this thesis we assume that $u_1, \ldots, u_n$ are positive.

The following proposition is the famous relation between term orders and matrix orders.

Proposition 2.1.12 (Robbiano [Rob85]). An arbitrary term order can be defined by a matrix order.

Example 2.1.13. We define well-known orders by matrices

$$M_{lex} = \begin{pmatrix} 1 & & \text{\Large 0} \\ & \ddots & \\ \text{\Large 0} & & 1 \end{pmatrix}, \quad M_{rlex} = \begin{pmatrix} \text{\Large 0} & & -1 \\ & \iddots & \\ -1 & & \text{\Large 0} \end{pmatrix},$$

$$M_{glex} = \left(\begin{array}{ccc|c} 1 & \cdots & 1 & 1 \\ 1 & & \text{\Large 0} & 0 \\ & \ddots & & \vdots \\ \text{\Large 0} & & 1 & 0 \end{array}\right), \quad M_{grlex} = \left(\begin{array}{c|ccc} 1 & 1 & \cdots & 1 \\ 0 & \text{\Large 0} & & -1 \\ \vdots & & \iddots & \\ 0 & -1 & & \text{\Large 0} \end{array}\right).$$

The matrices $M_{lex}, M_{rlex}, M_{glex}$ and M_{grlex} are matrix orders for the lexicographic order $\succ_{lex}$, the reverse lexicographic order $\succ_{rlex}$, the graded lexicographic order $\succ_{glex}$ and the graded reverse lexicographic order $\succ_{grlex}$, respectively.

The following matrix is for a graded weight order with lexicographic order.

$$M_{wlex} = \left(\begin{array}{ccc|c} w_1 & \cdots & w_{n-1} & w_n \\ 1 & & \text{\Large 0} & 0 \\ & \ddots & & \vdots \\ \text{\Large 0} & & 1 & 0 \end{array} \right)$$

This matrix defines the term order as follows.
For $t_1 = x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ and $t_2 = x_1^{\beta_1} \cdot x_2^{\beta_2} \cdots x_n^{\beta_n}$ in $\mathrm{pp}(x_1, \ldots, x_n)$,

$$t_1 \succ_{M_{wlex}} t_2 \iff \sum_{i=1}^{n} w_i\alpha_i > \sum_{i=1}^{n} w_i\beta_i \text{ or } \left(\sum_{i=1}^{n} w_i\alpha_i = \sum_{i=1}^{n} w_i\beta_i \text{ and } t_1 \succ_{lex} t_2 \right).$$

Example 2.1.14. The block orders can also be defined by matrices.

$$M_{block} = \begin{pmatrix} M_1 & & \text{\Large 0} \\ & \ddots & \\ \text{\Large 0} & & M_l \end{pmatrix}.$$

Each M_k is a matrix which defines the term order on each block where $1 \leq k \leq l$.

2.2 Gröbner bases

In this section, we describe the method of Gröbner bases, which allow us to solve problems about polynomial ideals in an algorithmic or computational fashion. The method of Gröbner bases is also used in several powerful computer algebra systems to study specific polynomial ideals that arise in applications.

Definition 2.2.1. Let f and g be non-zero polynomials in $K[\bar{X}]$ and $\succ$ be an arbitrary term order on the set $\mathrm{pp}(\bar{X})$.

- The set of power products of f that appear with a non-zero coefficient, is written $\mathrm{pp}(f)$.
- The biggest power product of $\mathrm{pp}(f)$ with respect to $\succ$ is denoted by $\mathrm{lpp}(f)$ and is called the **leading power product** of g with respect to $\succ$.
- The coefficient corresponding to $\mathrm{lpp}(f)$ is called the **leading coefficient** of f with respect to $\succ$ which is defined by $\mathrm{lc}(f)$.
- The product $\mathrm{lc}(f)\,\mathrm{lpp}(f)$ is called the **leading monomial** of f with respect to $\succ$ which is defined by $\mathrm{lm}(f)$.
- The **least common multiple** (LCM) of $\mathrm{lpp}(f)$ and $\mathrm{lpp}(g)$ is denoted by $\mathsf{LCM}(\mathrm{lpp}(f), \mathrm{lpp}(g))$.
- The **set of monomials** of f is denoted by $\mathrm{Mono}(f)$.
- If $\mathrm{lpp}(f) = X_1^{\beta_1} \cdots X_n^{\beta_n} \in \mathrm{pp}(\bar{X})$, then $\deg(f) := (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n$. The degree of $\mathrm{lpp}(f)$ in the variables x_i is defined by $\deg_{x_i}(\mathrm{lpp}(f)) = \beta_i$.

We define a reduction and an S-polynomial which we need to construct an algorithm for computing Gröbner bases.

Definition 2.2.2 (reduction1). Fix a term order. Let $f = a\alpha + f_1, g = b\alpha\beta + g_1$ with $\mathrm{lm}(f) = a\alpha$ in $K[\bar{X}]$ where $a, b \in K$, $\alpha, \beta \in \mathrm{pp}(\bar{A}, \bar{X})$ and $f_1, g_1 \in K[\bar{X}]$ such that α does not occur in f_1. Then a reduction $\xrightarrow{r1}_f$ is defined as follows:

$$g \xrightarrow{r1}_f b\alpha\beta + g_1 - ba^{-1}\beta(a\alpha + f_1),$$

where $b\alpha\beta$ need not be the leading monomial of g. In this thesis we call this reduction " Reduction1 ".
Of course, we can continue this reduction step until we have a polynomial which can not be reduced by f. (Since $K[\bar{X}]$ is a Noetherian ring, a number of reduction steps is always finite.) In this case, we use the notation $\xrightarrow{r1*}_f$, i.e., $g \xrightarrow{r1*}_f h$ means that for all $p \in \mathrm{pp}(h)$, α does not divide p.
A reduction $\xrightarrow{r1}_F$ by a set F of polynomials is also natural defined [BW93, CLO92, Win96]. For the same reason, we also use the notation $\xrightarrow{r1*}_F$.

Example 2.2.3. Let $f = x^2y - y^2$ and $g = x^5y$ in $\mathbb{R}[x, y]$. Then,

$$\begin{aligned} f &\xrightarrow{r1}_g f - x^3 \cdot g = x^3y \\ &\xrightarrow{r1}_g x^3y - xy \cdot g = xy^3. \end{aligned}$$

That is, $f \xrightarrow{r1*}_g xy^3$.

In the next section, we will see another special reduction "Reduction2" in polynomial rings over a polynomial ring. Therefore, we numbered the notation Reduction and $\xrightarrow{r}$ "1". For the same reason, in this section we define an (normal) S-polynomial as Spoly1, and in the next section we define a special S-polynomial as Spoly2.

Definition 2.2.4 (S-polynomial1). Fix a term order. Let $f, g \in K[\bar{X}]$ be non-zero polynomials. The S-polynomial of f and g is the following

$$\mathsf{Spoly1}(f, g) = \frac{\mathsf{LCM}(\mathrm{lpp}(f), \mathrm{lpp}(g))}{\mathrm{lm}(f)} f - \frac{\mathsf{LCM}(\mathrm{lpp}(f), \mathrm{lpp}(g))}{\mathrm{lm}(g)} g.$$

In this text, we define this S-polynomial as " Spoly1 ".

Example 2.2.5. Let $f = x^3y^2 - x^2y^3 + x$ and $g = 3x^4y + y$ in $\mathbb{R}[x, y]$ with $\succ_{glex}$. Then, $\mathrm{lcm}(\mathrm{lpp}(f), \mathrm{lpp}(g)) = x^4y^2$ and

$$\begin{aligned} \mathsf{Spoly1}(f, g) &= \frac{x^4y^2}{x^3y^2} \cdot f - \frac{x^4y^2}{3x^4y} \cdot g \\ &= x \cdot f - \frac{1}{3} \cdot y \cdot g \\ &= -x^3y^3 + x^2 - \frac{1}{3}y^2. \end{aligned}$$

An S-polynomial $\mathsf{Spoly1}(f, g)$ is designed to produce cancellation of leading monomials. A Gröbner basis [Buc65] of an ideal $I \subseteq K[\bar{X}]$ is a basis of I with special properties that make it possible to answer a lot of questions about I algorithmically.

Lemma 2.2.6. Let I be an ideal in $K[\bar{X}]$. Then $\mathrm{lm}(I)$ is also an ideal in $K[\bar{X}]$ where $\mathrm{lm}(I) := \{\mathrm{lm}(g) | g \in I\}$.

The following is a definition of Gröbner bases.

Definition 2.2.7 (Gröbner bases). Fix a term order $\succ$. A finite subset $G = \{g_1, \ldots, g_s\}$ of an ideal I in $K[\bar{X}]$ is said to be a **Gröbner basis** with respect to $\succ$ if

$$\langle \mathrm{lm}(g_1), \ldots, \mathrm{lm}(g_s) \rangle = \mathrm{lm}(I).$$

There are a lot of good properties of Gröbner bases. In fast, nowadays it is common knowledge that Gröbner bases and the Buchberger algorithm are fundamental tools for applications in several fields both inside and outside mathematics [Buc85]. If one is interested in the theory deeply in $K[\bar{X}]$, the author recommends to read one of books [BW93, CLO92, KR00, Win96]. In this section, we introduce the following three theorems which are needed later.

Theorem 2.2.8. Let I be an ideal in $K[\bar{X}]$. Then a basis G for I is a Gröbner basis for I if and only if for an pair f and g in G, $\mathsf{Spoly1}(f, g) \xrightarrow{r1*}_G 0$.

Theorem 2.2.9. Let $G = \{g_1, \ldots, g_t\}$ be a Gröbner basis for an ideal $I \subseteq K[\bar{X}]$ and let $f \in K[\bar{X}]$. Then there is a unique $r \in K[\bar{X}]$ with the following two properties:

1. No monomial of r is divisible by any of $\mathrm{lm}(g_1), \ldots, \mathrm{lm}(g_t)$.
2. There is $g \in I$ such that $f = g + r$.

In particular, r is the remainder on division of f by G no matter how the elements of G are listed when using the reduction $\xrightarrow{r1}$.

Theorem 2.2.10. Let G be a Gröbner basis for an ideal $I \subseteq K[\bar{X}]$ and let $f \in K[\bar{X}]$. Then, $f \in I$ if and only if $f \xrightarrow{r1*}_G 0$.

Theorem 2.2.8 enables us to construct a Gröbner basis G for a given finite set F of polynomials such that $\langle G \rangle = \langle F \rangle$. We can repeat computations of S-polynomials and reductions until we get a desired Gröbner basis. Now, we introduce the Buchberger algorithm [Buc65, Buc06] for computing Gröbner bases.

Algorithm 2.2.11. $\mathsf{Buchberger}(F, \succ)$ [Buc65]

Input $F = \{f_1, \ldots, f_s\}$: a finite subset of $K[\bar{X}]$,
  $\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output G: a Gröbner basis for $\langle F \rangle$ with respect to $\succ$ in $K[\bar{X}]$.
begin
  $G \leftarrow F$
  $P \leftarrow \{(f_i, f_j) \,|\, 1 \le i < j \le s\}$
   while $P \neq \emptyset$ **do**
    Take any element (f, f') from P
    $P \leftarrow P \backslash \{(f, f')\}$
    $r \leftarrow (\mathtt{Spoly1}(f, f')) \downarrow_G$ (see below $(*)$)
      if $r \neq 0$ **then**
        $P \leftarrow P \cup \{(g, r) \,|\, g \in G\}$
        $G \leftarrow G \cup \{r\}$
      end-if
   end-while
  `return`(G)
end
($(*)$ $h \downarrow_G$ denotes a normal form of h by $\xrightarrow{r1*}_G$, i.e., $h \downarrow_G$ is irreducible by $\xrightarrow{r1}_G$)

Definition 2.2.12. A **reduced Gröbner basis** for a polynomial ideal $I \subseteq K[\bar{X}]$ is a Gröbner basis for I such that

1. $\mathrm{lc}(p) = 1$ for all $p \in G$.
2. For all $p \in G$, no monomial of p lies in $\langle \mathrm{lm}(G \backslash \{p\}) \rangle$.

Although a Gröbner basis may contain redundant elements, the reduced Gröbner basis does not contain any. The reduced Gröbner basis is uniquely determined by an ideal $I \subseteq K[\bar{X}]$ and a term order.

2.3 Syzygies and Gröbner bases for modules

Syzygies are very important objects and basic ingredients for many constructions in homological algebra and algebraic geometry. In this section we give a method for computing syzygies. There exists some method for computing syzygies. Since we will see "**comprehensive Gröbner bases for modules**" in chapter 9, in this section we describe the relations between syzygies and Gröbner bases in $K[\bar{X}]$-module.

Definition 2.3.1. Let R be a ring, $p_1, \ldots, p_r \in R$ and $p := (p_1, \ldots, p_r)$ an r-tuple. An r-tuple $q = (q_1, \ldots, q_r) \in R^r$ is called a **syzygy** for $p_1, \ldots, p_r$ if and only if

$$q \cdot p = q_1 p_1 + q_2 p_2 + \cdots + q_r p_r = 0.$$

We write

$$\mathrm{Syz}(p_1, \ldots, p_r) := \{(q_1, \ldots, q_r) \in R^r | q_1 p_1 + \cdots + q_r p_r = 0\} \subseteq R^r$$

for the set of syzygies for $p_1, \ldots, p_r$.

It is easy to see that the set of syzygies forms a submodule of R^r. In the case $R = K[\bar{X}]$, it is a standard application of Gröbner bases to compute a basis of the syzygy module (see [Win86, BW93, GMP02]).

In several papers and books [FSK86, KR00, MM86, CLO97, GMP02], an algorithm for computing Gröbner bases for $K[\bar{X}]$-modules and its properties were introduced. We can easily extend the theory of Gröbner bases to modules. In order to describe the theory of Gröbner bases in modules, we need to extend the notation of term order to the free module $K[\bar{X}]^r$. We apply the following notations and definitions for the module structure.

Let $e_1, \ldots, e_r$ be the canonical basis of the free module $K[\bar{X}]^r = \bigoplus_{i=1}^{r} K[\bar{X}] e_i$. I.e, for each $i = 1, \ldots, r$,

$$e_i = \begin{array}{c} \\ \big(0, \ldots, 0, \end{array} \begin{array}{c} i\text{th} \\ 1, \end{array} \begin{array}{c} \\ 0, \ldots, 0\big) \in K[\bar{X}]^r \end{array}$$

denotes the i-th canonical basis vector of $K[\bar{X}]^r$ with 1 at the i-th place. We call

$$x^\alpha e_i = \begin{array}{c} \\ \big(0, \ldots, 0, \end{array} \begin{array}{c} i\text{th} \\ x^\alpha, \end{array} \begin{array}{c} \\ 0, \ldots, 0\big) \in K[\bar{X}]^r \end{array}$$

a **module power product** (involving component i) where $\alpha \in \mathbb{N}^n$, $x^\alpha \in \mathrm{pp}(\bar{X})$ and B^T is a transposed matrix B. The set of module power products with respect to $\bar{X}$ is defined as $\mathrm{pp}(\bar{X})^r$. (I.e., $x^\alpha e_i \in \mathrm{pp}(\bar{X})^r$.) Note that $e_i \cdot e_j = 0$ for $i \neq j$.

Definition 2.3.2 (Module orders). Let $\succ_t$ be a term order on $K[\bar{X}]$. A **module order** on $\mathrm{pp}(\bar{X})^r$ is a total order $\succ_m$ on the set of power product $\{x^\alpha e_i | \alpha \in \mathbb{N}^n, i = 1, \ldots, r\}$, which is compatible with the $K[\bar{X}]$-module structure including the order $\succ_t$, that is, satisfying

1. $x^\alpha e_i \succ_m x^\beta e_j \Rightarrow x^{\alpha+\gamma} e_i \succ_m x^{\beta+\gamma} e_j$,
2. $x^\alpha \succ_t x^\beta \Rightarrow x^\alpha e_i \succ_m x^\beta e_i$,

for all $\alpha, \beta, \gamma \in \mathbb{N}^n$, $i, j = 1, \ldots, r$.

Two module orders are of particular practical interest: POT and TOP which are the following.

Definition 2.3.3. Let $\succ$ be a admissible order on $K[\bar{X}]$ and $\succ_m$ be a module order on $K[\bar{X}]^r$ with $\succ$.

1. A module order $\succ_m$ is called POT (position-over-term) if $\succ_m$ satisfies
$$x^\alpha e_i \succ_m x^\beta e_j :\Leftrightarrow i < j \text{ or } (i = j \text{ and } x^\alpha \succ x^\beta).$$
This module order is written as $\succ_m := (POT, \succ)$.
2. A module order $\succ_m$ is called TOP (term-over-position) if $\succ_m$ satisfies
$$x^\alpha e_i \succ_m x^\beta e_j :\Leftrightarrow x^\alpha < x^\beta \text{ or } (x^\alpha = x^\beta \text{ and } i < j).$$
This module order is written as $\succ_m := (TOP, \succ)$.

As we saw the notations of $K[\bar{X}]$ in Definition 2.2.1, we apply the same notations for $K[\bar{X}]^r$. Let f, g be non-zero vectors in $K[\bar{X}]^r$ and $\succ_m$ an arbitrary module order on $\mathrm{pp}(\bar{X})^r$.

- The set of module power products of f that appear with a non-zero coefficient, is written $\mathrm{pp}(f)$.
- The biggest module power product of $\mathrm{pp}(f)$ with respect to $\succ_m$ is denoted by $\mathrm{lpp}(f)$ and is called the **leading power product** of g with respect to $\succ_m$.
- The coefficient corresponding to $\mathrm{lpp}(f)$ is called the **leading coefficient** of f with respect to $\succ_m$ which is defined by $\mathrm{lc}(f)$.
- The product $\mathrm{lc}(f)\,\mathrm{lpp}(f)$ is called the **leading monomial** of f with respect to $\succ_m$ which is defined by $\mathrm{lm}(f)$.
- The **least common multiple** (LCM) of $\mathrm{lpp}(f)$ and $\mathrm{lpp}(g)$ is defined by $\mathsf{LCM}(\mathrm{lpp}(f), \mathrm{lpp}(g))$.
- The **set of monomials** of f is denoted by $\mathrm{Mono}(f)$.
- If $\mathrm{lpp}(f) = X_1^{\beta_1} \cdots X_n^{\beta_n} e_i \in \mathrm{pp}(\bar{X})^r$, then $\deg_{\bar{X}}(\mathrm{lpp}(f)) := (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n$, and $\deg_{X_i}(\mathrm{lpp}(f)) := \beta_i \in \mathbb{N}$.

Note that, if $r = 1$, we apply $K[\bar{X}]^1$, $\mathrm{pp}(\bar{X})^1$ as $K[\bar{X}]$, $\mathrm{pp}(\bar{X})$.

Example 2.3.4. Let a, b, x, y be variables and $f = \begin{pmatrix} 2ax - bx + y^2 \\ axy + 3 \end{pmatrix}$ be a vector in $\mathbb{Q}[a, b, x, y]^2$. We have a module order $\succ_m := (POT, \succ_{\{x,y,a,b\}}) = (POT, (x \succ_{lex} y \succ_{lex} a \succ_{lex} b))$. Then we have the following

- $\mathrm{pp}(f) = \big\{axe_1, bxe_1, y^2e_1, axye_2, e_2\big\}$,
- $\mathrm{lpp}(f) = axe_1$,
- $\mathrm{lc}(f) = 2$,
- $\mathrm{lm}(f) = 2axe_1$,

- $\mathrm{Mono}(f) = \{2axe_1, bxe_1, y^2e_1, axye_2, 3e_2\}$,
- $\deg_{\{a,b,x,y\}}(\mathrm{lpp}(f)) = (1,0,1,0) \in \mathbb{N}^4$, $\deg_{\{a\}}(\mathrm{lpp}(f)) = 1$.

We can easily generalize the theory of Gröbner bases to the module $K[\bar{X}]^r$. If one can interested in "Gröbner bases for modules", then the author recommends to read one of books [KR00, CLO97, GMP02].

Definition 2.3.5 (Gröbner bases for modules). Fix a module order $\succ_m$. A finite subset $G = \{g_1, \ldots, g_s\}$ of a submodule M in $K[\bar{X}]^r$ is said to be a **Gröbner basis** with respect to $\succ_m$ if

$$\langle \mathrm{lm}(g_1), \ldots, \mathrm{lm}(g_s)\rangle = \mathrm{lm}(M).$$

We can easily apply the Buchberger algorithm for computing Gröbner bases in $K[\bar{X}]^r$. The algorithm for computing Gröbner bases is exactly the same as the Buchberger algorithm. (However, we need module orders.)

The next lemma and theorem show us the relation between syzygies and Gröbner bases in $K[\bar{X}]^r$, and how to compute a basis of syzygy module by using the theory of Gröbner bases in $K[\bar{X}]^r$. The following lemma, theorem and algorithm SYZ are from Greuel and Pfister's book [GMP02].

Lemma 2.3.6. Let $\succ$ be a term order on $\mathrm{pp}(\bar{X})$, $I \subset K[\bar{X}]^r = \bigoplus_{i=1}^r K[\bar{X}]e_i$ a submodule and S a Gröbner basis of I with respect to the module order $\succ_m = (POT, \succ)$. Then, for any $s = 0, \ldots, r-1$, $S' := S \cap \bigoplus_{i=s+1}^r K[\bar{X}]e_i$ is a Gröbner basis of $I' = I \cap \bigoplus_{i=s+1}^r K[\bar{X}]e_i$ with respect to $\succ_m$. In particular, S' generates I'.

Proof. Let $h \in I' \cap \bigoplus_{i=s+1}^r K[\bar{X}]e_i$, then we have to prove that there exists $f \in S'$ such that $\mathrm{lm}(f) | \mathrm{lm}(h)$. Since S is a Gröbner basis of I , there exists $f \in S$ such that there exists $\mathrm{lm}(f) | \mathrm{lm}(h)$. In particular, $\mathrm{lm}(f) \in \bigoplus_{i=s+1}^r K[\bar{X}]e_i$. By the definition of the module order POT, we obtain $f \in \bigoplus_{i=1}^r K[\bar{X}]e_i$, in particular, $f \in S'$. ☐

Theorem 2.3.7. Let $f_1, \ldots, f_k$ be vectors in $K[\bar{X}]^r$. Consider the canonical embedding

$$K[\bar{X}]^r \subseteq K[\bar{X}]^{r+k}$$

and the canonical projection

$$\pi : K[\bar{X}]^{r+k} \to K[\bar{X}]^k.$$

Let $G = \{g_1, \ldots, g_s\}$ be a Gröbner basis of $F = \langle f_1 + e_{r+1}, \ldots, f_k + e_{r+k}\rangle$ with respect to a module order $\succ_m = (POT, \succ)$ where $\succ$ is a term order on $\mathrm{pp}(\bar{X})$. Suppose that $\{g_1, \ldots, g_l\} = G \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$, then

$$\mathrm{Syz}(f_1, \ldots, f_k) = \langle \pi(g_1), \ldots, \pi(g_l)\rangle.$$

Proof. By Lemma 2.3.6, $G \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$ is a Gröbner basis of $F \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$. On the other hand, $\pi(F \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i) = \mathrm{Syz}(f_1, \ldots, f_k)$. Namely, let $h \in F \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$, that is, $h = \sum_{v=r+1}^{r+k} h_v e_v = \sum_{j=1}^k b_j(f_j + e_{r+j})$ for suitable $b_j \in K[\bar{X}]$. This implies that $\sum_{j=1}^k b_j f_j = 0$ and $b_j = h_{r+j}$.
Conversely, if $h = (h_1, \ldots, h_k)\,\mathrm{Syz}(f_1, \ldots, f_k)$, that is, if $\sum_{v=1}^k h_v f_v = 0$, then $\sum_{v=1}^k (f_v + e_{r+v}) \in F \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$. ☐

Now, we can construct an algorithm for computing a basis of a syzygy module which is the following.

Algorithm 2.3.8. $\mathsf{SYZ}(f_1, \ldots, f_k)$

Input $f_1, \ldots, f_k$: vectors in $K[\bar{X}]^r$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output $S = \{s_1, \ldots, s_l\} \subset K[\bar{X}]^k$ such that $\langle S \rangle = \mathrm{Syz}(f_1, \ldots, f_k) \subset K[\bar{X}]^k$.

1. $F := \{f_1 + e_{r+1}, \ldots, f_k + e_{r+k}\}$, where $e_1, \ldots, e_{r+k}$ denote the canonical generators of $K[\bar{X}]^{r+k} = K[\bar{X}]^r \oplus K[\bar{X}]^k$ such that $f_1, \ldots, f_k \in K[\bar{X}]^r = \bigoplus_{i=1}^{r} K[\bar{X}]e_i$.
2. Compute a Gröbner basis G of $\langle F \rangle$ with respect to $(POT, \succ)$.
3. $G_0 := G \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i = \{g_1, \ldots, g_l\}$, with $g_i = \sum_{j=1}^{k} a_{ij}e_{r+j}, i = 1, \ldots, l$.
4. $s_i := (a_{i1}, \ldots, s_{ik}), i = 1, \ldots, l$.
5. $\mathsf{return}(s_1, \ldots, s_l)$

In the algorithm SYZ, we need to compute a Gröbner basis G for $\langle F \rangle$ with respect to $(POT, \succ)$. It is well-known that the Gröbner basis G has the following properties. We can write the G as follows

$$G = \left(\begin{array}{ccccc|c} g_1 & \cdots & g_i & \cdots & g_p & 0 \\ \hline h_{1,r+1} & & h_{i,r+1} & & h_{p,r+1} & \\ \vdots & \cdots & \vdots & \cdots & \vdots & S \\ h_{1,r+k} & & h_{i,r+k} & & h_{p,r+k} & \end{array} \right)$$

where $g_1, \ldots, g_p \in K[\bar{X}]^r$, $h_{i,j} \in K[\bar{X}]$ for $1 \le i \le p$, $r \le j \le r+k$, and $S \subset K[\bar{X}]^k$. Each column is an element of G (i.e., each column is a vector.) Then, the Gröbner basis G satisfies the following;

1. $\{g_1, \ldots, g_p\}$ is a Gröbner basis for $\langle f_1, \ldots, f_k \rangle$ with respect to $(POT, \succ)$,
2. $\langle S \rangle = \mathrm{Syz}(f_1, \ldots, f_k)$, and
3. $g_i = h_{i,r+1}f_1 + \cdots + h_{i,r+k}f_k$ for $i \le i \le k$.

By the property of the module order POT, G holds the first property, and by the theorem 2.3.7, G holds the second property, too. We know that the method of computing inverse matrices. By this method of inverse matrices and the property of the module order POT, one can easily imagine the third property.

We introduce one more algorithm in this section. It is sometimes called "extended Gröbner bases algorithm". This algorithm outputs a Gröbner basis for a submodule generated by vectors in $K[\bar{X}]^r$ and a set of vectors which has the third property above. We need the algorithm in chapter 3,7,8 and 9.

In the algorithm, the following canonical projection maps are needed;

$$\pi_1 : K[\bar{X}]^{r+k} \to K[\bar{X}]^r,$$
$$(a_1, \ldots, a_r, a_{r+1}, \ldots, a_{r+k}) \mapsto (a_1, a_2, \ldots, a_r),$$

$$\pi_2 : K[\bar{X}]^{r+k} \to K[\bar{X}]^k,$$
$$(a_1, \ldots, a_r, a_{r+1}, \ldots, a_{r+k}) \mapsto (a_{r+1}, \ldots, a_{r+k}).$$

Algorithm 2.3.9. $\mathsf{EGA}(f_1, \ldots, f_k)$

Input $f_1, \ldots, f_k$: vectors in $K[\bar{X}]^r$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output $G = \{g_1, \ldots, g_l\} \subset K[\bar{X}]^r$: a Gröbner basis for $\langle f_1, \ldots, f_k \rangle$.
$E = \{h_1, \ldots, h_l\}$: a subset of $K[\bar{X}]^k$ such that $g_i = h_{i,1}f_1 + h_{i,2}f_2 + \cdots + h_{i,k}f_k$ for $1 \leq i \leq k$ where $h_i = (h_{i,1}, \ldots, h_{i,k}) \in K[\bar{X}]^k$.

1. $F := \{f_1 + e_{r+1}, \ldots, f_k + e_{r+k}\}$, where $e_1, \ldots, e_{r+k}$ denote the canonical generators of $K[\bar{X}]^{r+k} = K[\bar{X}]^r \oplus K[\bar{X}]^k$ such that $f_1, \ldots, f_k \in K[\bar{X}]^r = \bigoplus_{i=1}^{r} K[\bar{X}]e_i$.
2. Compute a Gröbner basis G' of $\langle F \rangle$ with respect to $(POT, \succ)$.
3. $G_0 := G' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i = \{g_1, \ldots, g_l\}$,
4. $G_1 := G' \backslash G_0$,
5. $G = \pi_1(G1), E = \pi_2(G1)$,
6. $\mathsf{return}(G, E)$

The compute algebra system singular[*1] [GMPS05] has the built-in commands which compute a Gröbner basis in $K[\bar{X}]^r$ and a basis of a syzygy module. (The author also has implemented Algorithm 2.3.8 and 2.3.9 in the computer algebra system Risa/Asir[*2]. See chapter 10.) In the next example, we execute Algorithm 2.3.8 and Algorithm 2.3.9, and we see the built-in commands std and syz.

Example 2.3.10. Let $g_1 = xy + z, g_2 = yz, g_3 = xz$ be polynomials in $\mathbb{Q}[x, y, z]$. We have the graded reverse lexicographic order $\succ_{grlex}$ such that $x \succ_{grlex} y \succ_{grlex} z$. Then, by Algorithm 2.3.8 and 2.3.9, first, we have to compute a Gröbner basis for $I = \langle g_1 + e_2, g_2 + e_3, g_3 + e_4 \rangle$ with respect to $(POT, \succ_{grlex})$ in $\mathbb{Q}[x, y, z]^4$. The compute algebra system singular works as follows.

```
> ring R=0,(x,y,z),(c,dp);
> module T=[x*y+z,1,0,0],[y*z,0,1,0],[x*z,0,0,1];
> module G=std(T);                    /* GB computation */
> G;
G[1]=[0,0,x,-y]
G[2]=[0,yz,-z,-y2]
G[3]=[0,xz,0,-xy-z]
G[4]=[z2,z,0,-y]
G[5]=[yz,0,1]
G[6]=[xz,0,0,1]
G[7]=[xy+z,1]
```

The set G is a Gröbner basis for I with respect to $(POT, \succ_{grlex})$. This output means

$$\left\{ \begin{pmatrix} 0 \\ 0 \\ x \\ -y \end{pmatrix}, \begin{pmatrix} 0 \\ yz \\ -z \\ -y^2 \end{pmatrix}, \begin{pmatrix} 0 \\ xz \\ 0 \\ -xy - z \end{pmatrix}, \begin{pmatrix} z^2 \\ z \\ 0 \\ -y \end{pmatrix}, \begin{pmatrix} yz \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} xz \\ 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} xy + z \\ 1 \\ 0 \\ 0 \end{pmatrix} \right\}.$$

Then, from `G[4], G[5], G[6], G[7]`, we can obtain a Gröbner basis G_1 for $\langle g_1, g_2, g_3 \rangle$

[*1] http://www.singular.uni-kl.de/
[*2] http://www.math.kobe-u.ac.jp/Asir/

with respect to $\succ_{grlex}$ as follows

$$G_1 = \{z^2, yz, xz, xy + z\}.$$

Moreover, each element of G_1 can be written by g_1, g_2, g_3 as follows

$$\begin{cases} z^2 = zg_1 - yg_3, \\ yz = g_2, \\ xz = g_3, \\ xy + z = g_1. \end{cases}$$

From `G[1]`,`G[2]`,`G[3]`, we obtain the set of the syzygies of $\{g_1, g_2, g_3\}$

$$\mathrm{Syz}(g_1, g_2, g_3) = \left\langle \begin{pmatrix} 0 \\ x \\ -y \end{pmatrix}, \begin{pmatrix} yz \\ -z \\ -y^2 \end{pmatrix}, \begin{pmatrix} xz \\ 0 \\ -xy - z \end{pmatrix} \right\rangle.$$

That is, the basis S of syzygies of $\{g_1, g_2, g_3\}$ is

$$S = \left\{ \begin{pmatrix} 0 \\ x \\ -y \end{pmatrix}, \begin{pmatrix} yz \\ -z \\ -y^2 \end{pmatrix}, \begin{pmatrix} xz \\ 0 \\ -xy - z \end{pmatrix} \right\}.$$

We give one more example. Let $f_1 = xy + y, f_2 = x^2 + y, f_3 = x$ be polynomials in $\mathbb{Q}[x, y]$. We have the graded reverse lexicographic $\succ_{grlex}$ order such that $x \succ_{grlex} y$. The built-in command `syz` outputs a basis of a syzygy module of the input $\{f_1, f_2, f_3\}$.

```
> ideal I=x*y+y,x^2+y,x;
> module M=syz(I);
> M;
M[1]=[0,x,-x2-y]
M[2]=[1,-1,x-y]
```

A basis M of a syzygy module for $\{f_1, f_2, f_3\}$ is

$$\left\{ \begin{pmatrix} 0 \\ x \\ -x^2 - y \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \\ -x - y \end{pmatrix} \right\}.$$

Chapter 3

Gröbner bases in polynomial rings over a polynomial ring

Many researchers have studied Gröbner bases in several domains: polynomial rings over a Euclidean domain [KRK88], over the integers [NG94], over commutative regular rings [Wei87], etc .... In this chapter we treat the theory of Gröbner bases in polynomial rings over a polynomial ring. In [IP98], Insa and Pauer described how to compute Gröbner bases in rings of differential operators with coefficients in a polynomial ring.(We will see the original method in chapter 7.) They worked in non-commutative rings, however we can easily apply this method to the commutative case for computing Gröbner bases in polynomial rings over a polynomial ring. This is one of the methods for computing Gröbner bases in polynomial rings over a polynomial ring. The other method is "computing a Gröbner basis by using a block order in a polynomial ring over a field."

In fact, both methods cannot compute a reduced Gröbner basis in polynomial rings over a polynomial ring. We define reduced Gröbner bases in polynomial rings over a polynomial ring, and give algorithms for computing them. This chapter is bases on the author's papers [Nab06, Nab07c].

3.1 Notations for $K[\bar{A},\bar{X}]$ and $K[\bar{A}][\bar{X}]$

Let $\bar{A} := \{A_1,\dots,A_m\}$ and $\bar{X} := \{X_1,\dots,X_n\}$ be finite sets of variables such that $\bar{A}\cap\bar{X} = \emptyset$. $\mathrm{pp}(\bar{X})$, $\mathrm{pp}(\bar{A})$ and $\mathrm{pp}(\bar{A},\bar{X})$ denote the sets of power products of $\bar{X}$, $\bar{A}$ and $\bar{A}\cup\bar{X}$, respectively. In this thesis, we define $K[\bar{A},\bar{X}]$ as a polynomial ring over a field K, and $K[\bar{A}][\bar{X}] := (K[\bar{A}])[\bar{X}]$ as a polynomial ring over a polynomial ring $K[\bar{A}]$ (the coefficient domain is the polynomial ring $K[\bar{A}]$). Let f and g be non-zero polynomials in $K[\bar{A}][\bar{X}]$ and $\succ$ be an arbitrary term order on the set of power products $\mathrm{pp}(\bar{X})$. If polynomials f and g are in $K[\bar{A}][\bar{X}]$, then we use the subscript $\bar{A}$ for the notations as follows:

- The **set of power products** of f that appear with a non-zero coefficient, is written $\mathrm{pp}_{\bar{A}}(f)$.
- The biggest power product of $\mathrm{pp}_{\bar{A}}(f)$ with respect to $\succ$ is denoted by $\mathrm{lpp}_{\bar{A}}(f)$ and is called the **leading power product** of g with respect to $\succ$.
- The coefficient corresponding to $\mathrm{lpp}_{\bar{A}}(f)$ is called the **leading coefficient** of f with respect to $\succ$ which is defined by $\mathrm{lc}(f)$ (or $\mathrm{lc}_{\bar{A}}(f)$).
- The product $\mathrm{lc}(f)\,\mathrm{lpp}(f)$ is called the **leading monomial** of f with respect to $\succ$ which is defined by $\mathrm{lm}_{\bar{A}}(f)$.
- The **least common multiple** (LCM) of $\mathrm{lpp}_{\bar{A}}(f)$ and $\mathrm{lpp}_{\bar{A}}(g)$ is defined by

$\mathsf{LCM}(\mathrm{lpp}_{\bar{A}}(f), \mathrm{lpp}_{\bar{A}}(g))$.
- The **set of monomials** of f is denoted by $\mathrm{Mono}_{\bar{A}}(f)$.
- If $\mathrm{lpp}(f) = A_1^{\alpha_1} \cdots A_m^{\alpha_m} X_1^{\beta_1} \cdots X_n^{\beta_n} \in \mathrm{pp}(\bar{A}, \bar{X})$, then $\deg_{\{\bar{A},\bar{X}\}}(f) := (\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_n) \in \mathbb{N}^{m+n}$.
If $\mathrm{lpp}_{\bar{A}}(f) = X_1^{\beta_1} \cdots X_n^{\beta_n} \in \mathrm{pp}(\bar{X})$, then $\deg_{\bar{X}}(f) := (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n$. **Note that the subscripts are $\{\bar{A}, \bar{X}\}$ and $\bar{X}$.**

Example 3.1.1. Let a, b, x, y be variables and $f = 2ax^2y + bx^2y + 3x + by + 1, g = abx^2 + 2xy + by + 2$ be polynomials.

If we consider polynomials f and g as members of $\mathbb{Q}[a, b, x, y]$ (the polynomial ring over the field $\mathbb{Q}$) with a block order
$\succ_{\{x,y\},\{a,b\}} := (x \succ_{lex} y, a \succ_{lex} b)$ (where $\succ_{lex}$ is the lexicographic order), then

- $\mathrm{pp}(f) = \{ax^2y, bx^2y, x, y, 1\}$,
- $\mathrm{lpp}(f) = ax^2y$,
- $\mathrm{lc}(f) = 2$,
- $\mathrm{lm}(f) = 2ax^2y$,
- $\mathrm{lcm}(\mathrm{lpp}(f), \mathrm{lpp}(g)) = abx^2y$,
- $\mathrm{Mono}(f) = \{2ax^2y, bx^2y, 3x, by, 1\}$,
- $\deg_{\{a,b,x,y\}}(f) = (1, 0, 2, 1) \in \mathbb{N}^4$.

If we consider polynomials f, g as members of $\mathbb{Q}[a, b][x, y]$ (the polynomial ring over the polynomial ring $\mathbb{Q}[x, y]$) with the lexicographic order $x \succ_{lex} y$, then

- $\mathrm{pp}_{\{a,b\}}(f) = \{x^2y, x, y, 1\}$,
- $\mathrm{lpp}_{\{a,b\}}(f) = x^2y$,
- $\mathrm{lc}_{\{a,b\}}(f) = 2a + b$,
- $\mathrm{lm}_{\{a,b\}}(f) = (2a + b)x^2y$,
- $\mathrm{lcm}(\mathrm{lpp}_{\{a,b\}}(f), \mathrm{lpp}_{\{a,b\}}(g)) = x^2y$,
- $\mathrm{Mono}_{\{a,b\}}(f) = \{(2a + b)x^2y, 3x, by, 1\}$,
- $\deg_{\{x,y\}}(f) = (2, 1) \in \mathbb{N}^2$.

3.2 Approach by Insa and Pauer

In [IP98], Insa and Pauer studied Gröbner bases in $K[\bar{X}][\bar{D}]$, i.e., rings of differential operators with coefficients in a commutative polynomial ring, where $\bar{D} = D_1, \ldots, D_n$ and $D_i = \frac{\partial}{\partial X_i} : K[\bar{X}] \to K[\bar{X}]$ is the partial derivative by X_i and $1 \le i \le n$. (We will see the original method in chapter 7.) They introduced a special S-polynomial and a special reduction in order to compute Gröbner bases in $K[\bar{X}][\bar{D}]$. Clearly, $K[\bar{X}][\bar{D}]$ is not a commutative ring, however in the commutative ring $K[\bar{A}][\bar{X}]$, it is possible to compute Gröbner bases by the same computation method. In this section we introduce the special S-polynomial and the special reduction which are from [IP98], and we give the Insa-Pauer algorithm for computing Gröbner bases in $K[\bar{A}][\bar{X}]$.

Proposition 3.2.1 (Insa and Pauer [IP98]). Let F be a finite set of polynomials in $K[\bar{A}][\bar{X}]$, $g \in K[\bar{A}][\bar{X}]$ and $\succ$ a term order on $\mathrm{pp}(\bar{X})$. Then there is a polynomial $r \in K[\bar{A}][\bar{X}]$ and there is a family $(h_f)_{f \in F}$ such that

- $g = \sum_{f \in F} h_f f + r$ (r is a remainder of g after division by F),

- for all $f \in F, h_f = 0$ or $g \succ h_f$,
- $r = 0$ or $\mathrm{lc}_{\bar{A}}(r) \notin \langle \mathrm{lc}_{\bar{A}}(f) | \ \mathrm{lpp}_{\bar{A}}(f) \text{ divides } \mathrm{lpp}_{\bar{A}}(r)\rangle$.

The polynomials r, h_f $(f \in F)$ can be computed as follows:
First set: $r := g$ and $h_f = 0$ $(f \in F)$.
While $r \neq 0$ and $\mathrm{lc}_{\bar{A}}(r) \in \langle \mathrm{lc}_{\bar{A}}(f) | \ \mathrm{lpp}_{\bar{A}}(f) \text{ divides } \mathrm{lpp}_{\bar{A}}(r)\rangle$ do the following:
let $F' := \{f \in F \,|\, \mathrm{lpp}_{\bar{A}}(f) \text{ divides } \mathrm{lpp}_{\bar{A}}(r)\}$, compute a family $(c_f)_{f \in F'}$ in $K[\bar{A}][\bar{X}]$ such that

$$\sum_{f \in F'} c_f \, \mathrm{lc}_{\bar{A}}(f) = \mathrm{lc}_{\bar{A}}(r).$$

Replace

$$r \text{ by } r - \sum_{f \in F'} c_f X^{\deg_{\bar{X}}(r) - \deg_{\bar{X}}(f)} f$$

and

$$h_f \text{ by } h_f + c_f X^{\deg_{\bar{X}}(r) - \deg_{\bar{X}}(f)}, \quad f \in F',$$

where $X^i := X_1^{i_1} \cdots X_n^{i_n}$, $i \in \mathbb{N}^n$.

By using Algorithm 2.3.9 EGA "extended Gröbner bases algorithm ", we can compute a family $(c_f)_{f \in F'}$ in $K[\bar{A}][\bar{X}]$ such that $\sum_{f \in F'} c_f \, \mathrm{lc}_{\bar{A}}(f) = \mathrm{lc}_{\bar{A}}(r)$. We simplify this proposition to the following definition.

Definition 3.2.2 (reduction2). Let F be a set of polynomials in $K[\bar{A}][\bar{X}]$ and $g = a\beta + g' \in K[\bar{A}][\bar{X}]$ where $a \in K[\bar{A}]$, $\beta \in \mathrm{pp}(\bar{X})$ and $g' \in K[\bar{A}][\bar{X}]$. Moreover, let $F' := \{f \in F \,|\, \mathrm{lpp}_{\bar{A}}(f) \text{ divides } \beta\}$. If $a \in \langle \mathrm{lc}_{\bar{A}}(F')\rangle \subseteq K[\bar{A}]$, the element a can be written as $a = \sum_{f_i \in F'} h_i \, \mathrm{lc}_{\bar{A}}(f_i)$ where $h_i \in K[\bar{A}]$. Then a reduction $\xrightarrow{r2}_F$ is defined as follows:

$$g \xrightarrow{r2}_F g - \sum_{f_i \in F'} h_i \frac{\beta}{\mathrm{lpp}_{\bar{A}}(f_i)} f_i.$$

In this chapter, we define this reduction as Reduction2 (written: $\xrightarrow{r2}$) . Actually, reducing g by F and reducing g by F' is the same. In this case, we can write the reduction $g \xrightarrow{r2}_{F'}$ instead of $g \xrightarrow{r2}_F$.
We can continue this reduction step until we have a polynomial which can not be reduced by F. In this case, we use the notation $\xrightarrow{r2*}_F$, i.e., $g \xrightarrow{r2*}_F h$ means that, for all $p \in \mathrm{Mono}_{\bar{A}}(h)$, there does not exist a set $F' := \{f \in F \,|\, \mathrm{lpp}_{\bar{A}}(f) \text{ divides } \mathrm{lpp}_{\bar{A}}(p)\}$ such that $\mathrm{lc}_{\bar{A}}(p) \in \langle \mathrm{lc}_{\bar{A}}(F')\rangle$. In Algorithm 3.2.8 Insa-Pauer, we will write Reduction2(g, F) as $g \xrightarrow{r2*}_F$.

Example 3.2.3. Let $F = \{f_1 = (a+b+1)y, f_2 = ay + 1\}$ in $\mathbb{Q}[a,b][x,y]$, $\succ$ the lexicographic order such that $x \succ y$ and $g = (b+1)xy - y \in \mathbb{Q}[a,b][x,y]$. Then,

$$\mathrm{lc}_{\bar{A}}(g) = b + 1 \in \langle \mathrm{lc}_{\bar{A}}(f_1), \mathrm{lc}_{\bar{A}}(f_2)\rangle = \langle a+b+1, a\rangle,$$

hence, $\mathrm{lc}_{\bar{A}}(g)$ can be written as

$$\mathrm{lc}_{\bar{A}}(g) = \mathrm{lc}_{\bar{A}}(f_1) - \mathrm{lc}_{\bar{A}}(f_2).$$

Clearly, $\mathrm{lpp}_{\bar{A}}(f_1) | \mathrm{lpp}_{\bar{A}}(g), \mathrm{lpp}_{\bar{A}}(f_2) | \mathrm{lpp}_{\bar{A}}(g)$. Therefore, g can be reduced by F as follows:

$$g \xrightarrow{r2}_F g - (\mathrm{lc}_{\bar{A}}(f_1)xf_1 - \mathrm{lc}_{\bar{A}}(f_2)yf_2) = 0.$$

Insa and Pauer also introduced the following special S-polynomial.

Definition 3.2.4 (S-polynomial2 [IP98]). Let G be a finite set of polynomials in $K[\bar{A}][\bar{X}]$ and let I be an ideal in $K[\bar{A}][\bar{X}]$ generated by G. For $E \subseteq G$, let

$$S_E := \left\{ (c_e)_{e\in E} \;\middle|\; \sum_{e\in E} c_e \operatorname{lc}_{\bar{A}}(e) = 0 \right\}.$$

(We can consider S_E as a set of syzygies for $\operatorname{lc}_{\bar{A}}(E)$.) Then for $s = (c_e)_{e\in E} \in S_E$,

$$\mathsf{Spoly2}(E, s) = \sum_{e\in E} c_e X^{\max(E)-\deg_{\bar{A}}(e)} e$$

is called S-polynomial with respect to s ,where

$$\max(E) := (\max_{e\in E} \deg_{\bar{X}}(e)_1, \ldots, \max_{e\in E} \deg_{\bar{X}}(e)_n) \in \mathbb{N}^n .$$

In this thesis, we call this special S-polynomial " Spoly2".

As we saw in chapter 2, we can compute the set S_E which is a set of syzygies of $\operatorname{lc}_{\bar{A}}(E)$, by using Algorithm 2.3.8 SYZ.

Example 3.2.5. Let $E = \{e_1 = (ab+b)x^2 + y, e_2 = (a+b)x + 1, e_3 = axy + by + 2\}$ in $\mathbb{Q}[a,b][x,y]$ and $\succ$ the lexicographic order such that $x \succ y$. Then, by the algorithm SYZ, we obtain a basis of a syzygy module of $\operatorname{lc}_{\bar{A}}(E)$

$$\{[0, -a, a+b], [-1, 1, b-1]\}.$$

If we take $[0, -a, a+b]$, then

$$\begin{aligned} \mathsf{Spoly2}(E, [0, -a, a+b]) &= 0e_1 + (-a)ye_2 + (a+b)xe_3 \\ &= (a+b)bxy + 2(a+b)x - ay. \end{aligned}$$

If we take $[-1, 1, b-1]$, then

$$\begin{aligned} \mathsf{Spoly2}(E, [-1, 1, b-1]) &= (-1)e_1 + (1)ye_2 + (b-1)xe_3 \\ &= xy + (b^2 - b - 1)y^2 + xy + 2(b-1)y. \end{aligned}$$

The definition of Gröbner bases in $K[\bar{A}][\bar{X}]$ is the following. In [IP98], Insa-Pauer used the remark of the definition as the definition of Gröbner bases in $K[\bar{A}][\bar{X}]$.

Definition 3.2.6 (Gröbner bases). Fix a term order. A finite subset $G = \{g_1, \ldots, g_s\}$ of an ideal I in $K[\bar{A}][\bar{X}]$ is said to be a **Gröbner basis** if

$$\langle \operatorname{lm}_{\bar{A}}(g_1), \ldots, \operatorname{lm}_{\bar{A}}(g_s) \rangle = \operatorname{lm}_{\bar{A}}(I).$$

Remark: This definition is equivalent to the following. We are able to understand the definition as follows: Let I be an ideal in $K[\bar{A}][\bar{X}]$ and let G be a finite subset of J. For $i \in \mathbb{N}^n$ let

$$\operatorname{lc}(i, I) := \langle \operatorname{lc}_{\bar{A}}(f) | f \in I, \deg_{\bar{X}}(f) = i \rangle.$$

Then G is a **Gröbner basis** of I (with respect to $\succ$) if and only if $\forall i \in \mathbb{N}^n$ the ideal $\operatorname{lc}(i, I) \subseteq K[\bar{A}]$ is generated by

$$\{\operatorname{lc}_{\bar{A}}(g) \,|\, g \in G, i \in \deg_{\bar{X}}(g) + \mathbb{N}^n \}.$$

Example 3.2.7. Consider the ring $\mathbb{Q}[a,b][x,y]$ with the lexicographic order $x \succ_{lex} y$, and let $I = \langle f_1, f_2 \rangle = \langle ax + bx + y, bxy \rangle$. Since

$$\mathrm{lm}_{\{a,b\}}(\mathsf{Spoly2}(f_1, f_2)) = by^2 \notin \langle \mathrm{lm}_{\{a,b\}}(f_1), \mathrm{lm}_{\{a,b\}}(f_2) \rangle = \langle (a+b)x, bxy \rangle,$$

$\{f_1, f_2\}$ is not a Gröbner basis for I. Actually, a Gröbner basis for I is $\{f_1, f_2, by^2\}$.

There are a lot of applications of Gröbner bases in $K[\bar{A}][\bar{X}]$ which are well-known in commutative polynomial rings over a field. For instance, if G is a Gröbner basis for an ideal I in $K[\bar{A}][\bar{X}]$, then $\forall g \in I$, $g \xrightarrow{r2*}_G 0$. In this section, we do not describe the details of the properties of Gröbner bases in $K[\bar{A}][\bar{X}]$ (see [IP98] Proposition 3). The following algorithm is for computing Gröbner bases in $K[\bar{A}][\bar{X}]$.

Algorithm 3.2.8. $\mathsf{Insa\text{-}Pauer}(F, \succ)$

Input: F: a finite set of polynomials in $K[\bar{A}][\bar{X}]$,
    $\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output: G: a Gröbner basis of F with respect to $\succ$ in $K[\bar{A}][\bar{X}]$.
begin
  $G \leftarrow F$
  $B \leftarrow \{(f_{i_1}, f_{i2} \dots f_{ip}) \,|\, 1 \le i_1 < i_2 \cdots < i_p \le s, 2 \le p \le s\}$
   while $B \neq \emptyset$ **do**
    Take any element E from B; $B \leftarrow B \backslash \{E\}$
    $S_E \leftarrow$ Compute a basis of a syzygy module for $\mathrm{lc}_{\bar{A}}(E)$
     while $S_E \neq \emptyset$ **do**
      Take any element α from S_E; $S_E \leftarrow S_E \backslash \{\alpha\}$
      $h \leftarrow$Spoly2(E, α)
      $r \leftarrow$Reduction2(h, G)
       if $r \neq 0$ **then**
        $B \leftarrow B \cup \{(r, g_{j_1}, .., g_{j_q}) \,|\,$ distinct elements $g_{j_1}, \dots, g_{j_p} \in G, 1 \le p \le |G|\}$
        $G \leftarrow G \cup \{r\}$
       end-if
     end-while
   end-while
  `return`(G)
end

Remark: As we said earlier, we need the special S-polynomial Spoly2 and the special reduction Reduction2 in order to compute Gröbner bases in $K[\bar{A}][\bar{X}]$. In this point, this algorithm is more complicated than the Buchberger algorithm. There exist some criteria for computing Gröbner bases in $K[\bar{A}][\bar{X}]$. We can apply Buchberger's criteria [Buc79] and Zhou and Winkler's work [ZW06] for computing Gröbner bases.

The algorithm Insa-Pauer has been implemented by the author in the computer algebra system Risa/Asir. In following example, we give some outputs of the program.

Example 3.2.9. Let $F_1 = \{ax + by + 1, (b+1)y, ax^2 + bx + y\}$ and $F_2 = \{bxz + ay + a, yz + by + 3, ay^2z + bx + a\}$ be subsets of $\mathbb{Q}[a,b][x,y,z]$ and $\succ$ the lexicographic order such that $x \succ y \succ z$.
Then, the program outputs the following set as a Gröbner basis for $\langle F_1 \rangle$ with respect to $\succ$;

```
[-b+1,a*x+1,-2*y]
```

The program outputs the following set as a Gröbner basis for $\langle F_2\rangle$ with respect to $\succ$;

```
[((6*b-1)*a*y+a*z+8*a)*x+(-3*b+1)*a^2*y^3-8*a^2*y^2+3*a^2*y,(-a*z^2+(-b-8
)*a*z+(10*b-3)*a)*x+(-9*b+3)*a^2*y^2-24*a^2*y+9*a^2,b*x-b*a*y^2-3*a*y+a,(
z+b)*y+3,3*b^2*a*y-a*z^2+(-b-8)*a*z+(10*b-3)*a,-a*z^3+(-2*b-8)*a*z^2+(-b^
2+2*b-3)*a*z+(b^2-3*b)*a]
```

3.3 Approach via block orders

In this section, we introduce another computation method of Gröbner bases in $K[\bar{A}][\bar{X}]$. In fact, by computing Gröbner bases in $K[\bar{A},\bar{X}]$ with respect to a block order $\succ_{\bar{X},\bar{A}}$, we can obtain a Gröbner basis in $K[\bar{A}][\bar{X}]$. In this approach, we need a normal S-polynomial Spoly1 and reduction reduction1 which we saw in chapter 2. Before we describe an algorithm for computing Gröbner bases in $K[\bar{A}][\bar{X}]$, we need the following theorem.

Theorem 3.3.1. Let F be a finite set of polynomials in $K[\bar{A}][\bar{X}]$. F can be seen as a finite subset of polynomials in $K[\bar{A},\bar{X}]$ and we write the set as F again. Let $G=\{g_1,\dots,g_s\}$ be a Gröbner basis for $\langle F\rangle$ in $K[\bar{A},\bar{X}]$ with respect to a block order $\succ_{\bar{X},\bar{A}}:=(\succ_1,\succ_2)$(i.e., $\bar{X}\gg\bar{A}$). G can be seen as a set of $K[\bar{A}][\bar{X}]$ and we write the set as G again. Then, G is also a Gröbner basis for $\langle F\rangle$ with respect to $\succ_1$ in $K[\bar{A}][\bar{X}]$.

Proof. For all $h\in\langle F\rangle\subseteq K[\bar{A}][\bar{X}]$, we prove that $\mathrm{lm}_{\bar{A}}(h)$ is generated by $\{\mathrm{lm}_{\bar{A}}(g)|g\in G\}$. Since h can be seen as an element of $K[\bar{A},\bar{X}]$ and G is a Gröbner basis for $\langle F\rangle$ in $K[\bar{A},\bar{X}]$, h can be written as

$$h=h_1g_1+\cdots+h_sg_s$$

such that $\mathrm{lm}(h)\succ_{\bar{X},\bar{A}}\mathrm{lm}(h_1g_1)\succ_{\bar{X},\bar{A}}\cdots\succ_{\bar{X},\bar{A}}\mathrm{lm}(h_sg_s)$ where $h_1,\dots,h_s\in K[\bar{A},\bar{X}]$. As $\succ_{\{\bar{X},\bar{A}\}}$ is a block order on $K[\bar{A},\bar{X}]$, we have

$$\mathrm{lm}_{\bar{A}}(h)\succ_1\mathrm{lm}_{\bar{A}}(h_1g_1)\succ_1\cdots\succ_1\mathrm{lm}_{\bar{A}}(h_sg_s)$$

in $K[\bar{A}][\bar{X}]$. W.l.o.g., $h_1g_1,\dots,h_kg_k$ have the same leading power product "$\mathrm{lpp}_{\bar{A}}(h)$" where $k\le s$. That is,

$$\mathrm{lm}_{\bar{A}}(h)=\mathrm{lm}_{\bar{A}}(h_1g_1)+\cdots+\mathrm{lm}_{\bar{A}}(h_kg_k).$$

We have $\mathrm{lm}_{\bar{A}}(h_ig_i)=\mathrm{lm}_{\bar{A}}(h_i)\,\mathrm{lm}_{\bar{A}}(g_i)$, hence

$$\mathrm{lm}_{\bar{A}}(h)=\mathrm{lm}_{\bar{A}}(h_1)\,\mathrm{lm}_{\bar{A}}(g_1)+\cdots+\mathrm{lm}_{\bar{A}}(h_k)\,\mathrm{lm}_{\bar{A}}(g_k).$$

Therefore, $\mathrm{lm}_{\bar{A}}(h)\in\langle\mathrm{lm}_{\bar{A}}(g_1),\dots,\mathrm{lm}_{\bar{A}}(g_s)\rangle$. G is a Gröbner basis for $\langle F\rangle$ with respect to $\succ_1$ in $K[\bar{A}][\bar{X}]$. $\square$

By Theorem 3.3.1, we are able to construct an algorithm for computing Gröbner bases in $K[\bar{A}][\bar{X}]$.

Algorithm 3.3.2. GröbnerBasisB$(F,\succ)$

Input F: a finite set of polynomials in $K[\bar{A}][\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output G: a Gröbner basis of $\langle F\rangle$ in $K[\bar{A}][\bar{X}]$.

1. Consider F as a set of polynomials in $K[\bar{A}, \bar{X}]$.
2. Compute the reduced Gröbner basis G for $\langle F \rangle$ with respect to a block order $\succ_{\bar{X},\bar{A}} = (\succ, \succ_1)$ in $K[\bar{A}, \bar{X}]$ where $\succ_1$ is a term order on $\mathrm{pp}(\bar{X})$.
3. Consider G as a set of polynomials in $K[\bar{A}][\bar{X}]$. Then, by Theorem 3.3.1, G is a Gröbner basis for $\langle F \rangle$ with respect to $\succ_{\bar{X}}$ in $K[\bar{A}][\bar{X}]$.

Remark: We do not need to compute reduced Gröbner bases in $K[\bar{A}, \bar{X}]$. It suffices to compute a (normal) Gröbner basis. Since in chapter 4 and 5 we describe the algorithms Suzuki-Sato, NEW which has the algorithm GröbnerBasisB and need the properties of reduced Gröbner bases in $K[\bar{A}, \bar{X}]$, we built in reduced Gröbner bases computation in the algorithm (in order to avoid writing the same algorithm twice).

Since we do not need the special S-polynomial Spoly2 and the special reduction reduction2 in this algorithm, this algorithm is much more efficient than the algorithm Insa-Pauer.

Example 3.3.3. Let a, b, x, y be variables and $f_1 = (a-1)x + by^2, f_2 = ay + b$ in $\mathbb{Q}[a, b][x, y]$, $\succ'$ a lexicographic order such that $x \succ' y$. Moreover, let $\succ$ be a block order such that $x \succ_{lex} y \gg a \succ_{lex} b$. By the algorithm, first we compute the reduced Gröbner basis G for $\langle f_1, f_2 \rangle$ with respect to $\succ$,

$$G = \{ay + b, (a-1)x + by^2, -xy - bx + by^3\}.$$

Then, we can consider G as a subset of $\mathbb{Q}[a, b][x, y]$. G is a Gröbner basis for $\langle f_1, f_2 \rangle$ with respect to $\succ'$ in $\mathbb{Q}[a, b][x, y]$.

3.4 Problems

Here we discuss problems of the some algorithms Insa-Pauer and GröbnerBasisB (for computing reduced Gröbner bases).

3.4.1 Approach by Insa and Pauer

In this subsection, we consider a problem of the Insa-Pauser approach by the following example.

Example 3.4.1. Let $f_1 = a^2x - a$ and $f_2 = (a^3 - a)x - a^2 + 1$ be polynomials in $\mathbb{Q}[a][x]$. Then, a Gröbner basis of $\langle f_1, f_2 \rangle$ is $\{f_1, f_2\}$, because $\mathtt{Spoly1}(f_1, f_2) = 0$, $f_1 \xrightarrow{r2*}_{f_2} f_1$ and $f_2 \xrightarrow{r2*}_{f_1} f_2$ in $\mathbb{Q}[a][x]$. However, we have

$$f_3 = a \cdot f_1 - f_2 = ax - 1.$$

The polynomial f_3 is an element of $\langle f_1, f_2 \rangle$, and f_3 divides f_1 and f_2. This means $\langle f_3 \rangle = \langle f_1, f_2 \rangle$. That is, $\{f_3\}$ is a Gröbner basis for $\langle f_1, f_2 \rangle$, too. $\{f_3\}$ is simpler than $\{f_1, f_2\}$. However, $\{ax - 1\}$ cannot be computed by the Insa-Pauer algorithm.

3.4.2 Approach via block order

In this subsection, we give a problem of the approach of block orders $\succ_{\bar{X},\bar{A}}$ by the following example.

Example 3.4.2. Let $F = \{f_1 = ax + 1, f_2 = (b + 1)y, f_3 = az + bz + z\}$ be a set of polynomials in $\mathbb{Q}[a, b][x, y, z]$. F can be seen as a set of $\mathbb{Q}[a, b, x, y, z]$. We have a block order $\succ_{\{x,y,z\},\{a,b\}} = (\succ_{lex}, \succ_{grlex})$ such that $x \succ_{lex} y \succ_{lex} z$ and $a \succ_{grlex} b$ where $\succ_{lex}$ is the lexicographic order and $\succ_{grlex}$ is the graded reverse lexicographic order. Then the reduced Gröbner bases $G \subset \mathbb{Q}[a, b, x, y]$ for $\langle F\rangle$ with respect to a block order $\succ_{\{x,y,z\},\{a,b\}}$ is the following.

$$G = \{g_1 = (a + b + 1)z, g_2 = (b + 1)y, g_3 = yz, g_4 = ax + 1, g_5 = (b + 1)xz - z\}.$$

Since G is the reduced Gröbner basis of $\langle F\rangle$ in $\mathbb{Q}[a, b, x, y, z]$, $g \in G$ cannot be reduced by $G\backslash\{g\}$ with respect to the block order in $\mathbb{Q}[a, b, x, y, z]$. However, look at g_5. Then, we have

$$\mathrm{lm}_{\bar{A}}(g_5) \in \langle \mathrm{lm}_{\bar{A}}(G\backslash\{g_5\})\rangle \subset \mathbb{Q}[a, b][x, y, z].$$

That is, g_5 can be written

$$g_5 = x \cdot g_1 - z \cdot g_4.$$

This means that g_5 can be still reduced to 0 by g_1 and g_4 in $\mathbb{Q}[a, b][x, y, z]$. The polynomial g_5 is a redundant polynomial in $\mathbb{Q}[a, b][x, y, z]$. By the algorithm GröbnerBasisB, $G\backslash\{g_5\}$ cannot be computed.

Problem: Sometimes there exits a Gröbner basis which is simpler (or more minimal) than Gröbner bases computed by either of the two methods above.
This Gröbner basis cannot be computed by the two methods.

3.5 Reduced Gröbner bases

In this section, we define a reduced Gröbner basis for an ideal in $K[\bar{A}][\bar{X}]$ and give algorithms for computing a reduced Gröbner basis. First, we define weak reduced Gröbner bases in $K[\bar{A}][\bar{X}]$.

Definition 3.5.1 (Weak reduced Gröbner bases). Let $\succ_{\bar{X},\bar{A}} := (\succ_1, \succ_2)$ be a block order and I an ideal in $K[\bar{A}][\bar{X}]$. Then, a **weak reduced Gröbner basis** G for I with respect to $\succ_1$ and $\succ_{\bar{X},\bar{A}}$, is a Gröbner basis for I in $K[\bar{A}][\bar{X}]$ such that

1. for all $p \in G$, no monomial in $\mathrm{Mono}(p)$ lies in $\langle \mathrm{lm}(G\backslash \{p\})\rangle$ in $K[\bar{A}, \bar{X}]$ with respect to $\succ_{\bar{X},\bar{A}}$,
2. for all $p \in G$, no monomial in $\mathrm{Mono}_{\bar{A}}(p)$ lies in $\langle \mathrm{lm}_{\bar{A}}(G\backslash \{p\})\rangle$ in $K[\bar{A}][\bar{X}]$ with respect to $\succ_1$,
3. for all $p \in G$, $\mathrm{lc}(p) = 1$ with respect to $\succ_{\bar{X},\bar{A}}$.

As we said earlier, the algorithms Insa-Pauer and GröbnerBasisB cannot always compute reduced Gröbner bases in $K[\bar{A}][\bar{X}]$.

How do we compute (weak) reduced Gröbner bases in $K[\bar{A}][\bar{X}]$?

A polynomial ring $K[\bar{A}][\bar{X}]$ can be seen as a polynomial ring $K[\bar{A}, \bar{X}]$. This means that the polynomial ring $K[\bar{A}][\bar{X}]$ has properties of $K[\bar{A}, \bar{X}]$. In this sense, we have two reduction systems reduction1, reduction2 and two S-polynomial systems Spoly1, Spoly2 for computing weak reduced Gröbner bases in $K[\bar{A}][\bar{X}]$.

For instance, in Example 3.4.1. We have a Gröbner basis $\{f_1 = ax^2 - a, f_2 = (a^3 - a)x - a^2 + 1\}$. If we use reduction1 or Spoly1 to the Gröbner basis $\{f_1, f_2\}$, then we can obtain $ax + 1$ by the computation

$$f_2 \xrightarrow{r1*}_{\{f_1\}} ax + 1, \text{ or } \texttt{Spoly1}(f_1, f_2) = ax + 1.$$

Since $f_1 \xrightarrow{r1*}_{\{ax+1\}} 0$ and $f_2 \xrightarrow{r1*}_{\{ax+1\}} 0$, $\{ax + 1\}$ is a weak reduced Gröbner basis for $\langle f_1, f_2\rangle$.

In Example 3.4.2, we obtained a Gröbner basis $G = \{g_1, g_2, g_3, g_4, g_5\}$ by the algorithm GröbnerBasisB . Let's apply reduction2 to G. Then, since

$$\mathrm{lpp}_{\{a,b\}}(g_1) | \mathrm{lpp}_{\{a,b\}}(g_5), \quad \mathrm{lpp}_{\{a,b\}}(g_4) | \mathrm{lpp}_{\{a,b\}}(g_5)$$

and

$$\mathrm{lc}_{\{a,b\}}(g_5) = -\mathrm{lc}_{\{a,b\}}(g_1) + \mathrm{lc}_{\{a,b\}}(g_4) = -(a + b + 1) + a = -b - 1,$$

we have $g_5 \xrightarrow{r1*}_{\{g_1,g_2\}} 0$. Thus, g_5 is a redundant polynomial which is found by reduction1. By Definition 3.5.1, a weak reduced Gröbner basis for G is $\{g_1, g_2, g_3, g_4\}$ with respect to the lexicographic order with $x \succ y \succ z$.

By the observation above, we need reduction1, reduction2, Spoly1 and Spoly2 for computing weak reduced Gröbner bases in $K[\bar{A}][\bar{X}]$. Now, we can easily construct an algorithm for computing weak reduced Gröbner bases. We know that by the algorithms Insa-Pauer or GröbnerBasisB, we can obtain a Gröbner basis G in $K[\bar{A}][\bar{X}]$. This Gröbner basis G is not always a weak reduced Gröbner basis, hence we need reduction1 and reduction2 to reduce G to a reduced Gröbner basis. Actually, we need two reduction systems reduction1, reduction2 and one of two S-polynomial systems Spoly1 and Spoly2.
The following algorithm returns a weak reduced Gröbner basis. In the first step of the following algorithm, we apply Insa-Pauer or GröbnerBasisB for computing Gröbner bases in $K[\bar{A}][\bar{X}]$.

Algorithm 3.5.2. WRGB$(F, \succ_1, \succ_2)$ (Weak reduced Gröbner bases)

Input F: a finite set of polynomials in $K[\bar{A}][\bar{X}]$,
$\succ_1$: a term order on $\mathrm{pp}(\bar{X})$,
$\succ_2$: a term order on $\mathrm{pp}(\bar{A})$,
$(\succ_{\bar{X},\bar{A}} := (\succ_1, \succ_2)$: a block order,)
Output G: a weak reduced Gröbner basis of $\langle F\rangle$ with respect to $\succ_1$ and $\succ_{\bar{X},\bar{A}}$ in $K[\bar{A}][\bar{X}]$.
begin
$G \leftarrow$ Compute a Gröbner basis G for $\langle F\rangle$ by Insa-Pauer or GröbnerBasisB
$E1 \leftarrow 0$
while $E1 \neq 1$ **do**
if there exists $p \in G$ such that
$\left(p \xrightarrow{r1*}_{\{G\backslash\{p\}\}} p_1 \text{ and } p \neq p_1\right)$ or $\left(p \xrightarrow{r1*}_{\{G\backslash\{p\}\}} p_1 \text{ and } p \neq p_1, \text{ with respect to } \succ_{\bar{X},\bar{A}}\right)$
then
if $p_1 \neq 0$ **then**
$G \leftarrow \{G\backslash\{p\}\} \cup \{p_1\}$
else if
$G \leftarrow G\backslash\{p\}$

end-if
else-if
$E1 \leftarrow 1$
end-if
end-while
`return`(G)
end

Theorem 3.5.3. The algorithm WRGB terminates. The output forms a weak reduced Gröbner basis for $\langle F\rangle$ with respect to $\succ_1$ and $\succ_{\bar{X},\bar{A}}$ in $K[\bar{A}][\bar{X}]$.

Proof. In the first line of Algorithm 3.5.2 WRGB, if we apply Insa-Pauer for computing a Gröbner basis for $\langle F\rangle$ in $K[\bar{A}][\bar{X}]$, then Insa-Pauer terminates. (Since $K[\bar{A}]$ is a Noetherian ring, $K[\bar{A}][\bar{X}]$ is a Noetherian ring too. Thus, the termination of Insa-Pauer is guaranteed because we have a finite ascending chain condition of properly contained ideals over a Noetherian ring.) In the first step of Algorithm 3.5.2 WRGB, if we apply GröbnerBasisB for computing a Gröbner basis for $\langle F\rangle$ in $K[\bar{A}][\bar{X}]$, obviously GröbnerBasisB terminates. (see [Buc65]). Let G be a finite set of polynomials in $K[\bar{A}][\bar{X}]$. In the **while-loop** step, if there exists an element p of $\mathrm{Mono}(g)$ or $\mathrm{Mono}_{\bar{A}}(g)$ which can be reduced to p_1 by some polynomials of $G\backslash\{g\}$ in reduction1 or reduction2, then we always have $\mathrm{lm}(p) \succ_{\bar{X},\bar{A}} \mathrm{lm}(p_1)$ ($\mathrm{lm}(p_1)$ is smaller or equal than $\mathrm{lm}(p)$ with respect to the term order $\succ_{\bar{X},\bar{A}}$). That is, the result of applying reduction1 or reduction2 to any monomial $m \in \mathrm{Mono}(g)\cup\mathrm{Mono}_{\bar{A}}(g)$ has a leading monomial which cannot be greater than m with respect to $\succ_{\bar{X},\bar{A}}$. Therefore, iterated application of reduction1 and reduction2 to G will eventually terminate. This algorithm terminates and the outputs satisfy the properties of Definition 3.5.1. □

In Algorithm 3.5.2 WRGB, if we apply the algorithm Insa-Pauer for computing a Gröbner basis, then we need syzygy computations Spoly1 and "extended Gröbner bases algorithm Reduce1" (i.e., we have to apply the algorithms SYZ and EGA for computing them.). In general, syzygy computations and "extended Gröbner bases algorithm" are expensive. However, in Algorithm 3.5.2 WRGB, if we apply the algorithm GröbnerBasisB, then we do not need any syzygy computation. Actually, the algorithm GröbnerBasisB is a normal Gröbner bases computation in polynomial rings over a field with respect to a block order. At present, we have very powerful programs for computing Gröbner basis in $K[\bar{A},\bar{X}]$ in the computer algebra systems `Singular`[*1], `Risa/Asir`[*2] and `Magma`[*3]. We can apply these powerful programs for computing weak reduced Gröbner bases in $K[\bar{A}][\bar{X}]$. Thus, concerning implementation and speed, WRGB with GröbnerBasisB is better than WRGB with Insa-Pauer.

Before concluding this section, we consider a property of reduced Gröbner bases. Now we have a question.

"Is a weak reduced Gröbner basis uniquely determined by an ideal $I \subseteq K[\bar{A}][\bar{X}]$ and term orders?"

In fact, this answer is **"NO"**. A weak reduced Gröbner basis is not unique. We have the following easy example for this question.

[*1] `http://www.singular.uni-kl.de/`
[*2] `http://www.math.kobe-u.ac.jp/Asir/`
[*3] `http://magma.maths.usyd.edu.au/magma/`

Example 3.5.4. Let $F = \{(ab+1)xy, (ac+1)xy\}$ be a subset of $\mathbb{Q}[a,b,c][x,y]$ and $\succ_{\{x,y\},\{a,b,c\}} = (\succ_{lex}, \succ_{lex})$ a block order with $x \succ_{lex} y$ and $a \succ_{lex} b \succ_{lex} c$. Then, F is a weak reduced Gröbner basis for $\langle F \rangle$ in $\mathbb{Q}[a,b,c][x,y]$. In fact, F satisfies the properties 1,2 of Definition 3.5.1. However, we can say $\langle F \rangle = \langle (ac+1)xy, (-b+c)xy \rangle$. The set $G = \{(ac+1)xy, (-b+c)xy\}$ is a weak reduced Gröbner basis for $\langle F \rangle$ with respect to $\succ_{lex}$ and $\succ_{\{x,y\},\{a,b,c\}}$, too. Therefore, a weak reduced Gröbner basis is not uniquely determined.
Note that, actually, $\mathrm{lc}_{\bar{A}}(G) = \{ac+1, -b+c\}$ **is the reduced Gröbner basis for** $\langle \mathrm{lc}_{\bar{A}}(F) \rangle$ **with respect to** $\succ_{lex}$ **in** $\mathbb{Q}[a,b,c]$**.** Hence, in this case, by computing the reduced Gröbner basis for $\langle \mathrm{lc}_{\bar{A}}(F) \rangle$ in $\mathbb{Q}[a,b,c]$, we can obtain a unique reduced Gröbner basis.

We give one more example facilitate the understanding of the next definition. Let $F = \{(ac+b)x^2, (ac-c+bd^2)x^2, (-cd-bc+bd^3)x\} \subset \mathbb{Q}[a,b,c,d][x]$. We have the lexicographic order $\succ$ such that $a \succ b \succ c \succ d$. In fact, F is a weak Gröbner basis for $\langle F \rangle$. Suppose that $e \in \mathrm{lpp}_{\{a,b,c,d\}}(F)$, $F_e = \{f \mid \mathrm{lpp}_{\{a,b,c,d\}}(f) = e\}$. We have $\mathrm{lpp}_{\{a,b,c,d\}}(F) = \{x, x^2\}$, so $F_x = \{(-cd-bc+bd^3)x\}$ and $F_{x^2} = \{(ac+b)x^2, (ac-c+bd^2)x^2\}$. Let's consider all coefficients of F_x and F_{x^2}. Since $\mathrm{lc}_{\{a,b,c,d\}}(F_x) = \{-cd-bc+bd^3\}$ has only one element, $\{-cd-bc+bd^3\}$ is the reduced Gröbner basis for an ideal generated by itself. Next we consider $\mathrm{lc}_{\{a,b,c,d\}}(F_{x^2}) = \{ac+b, ac-c+bd^2\}$. Actually, $\{ac+b, ac-c+bd^2\}$ is **NOT** the reduced Gröbner basis for the ideal generated by itself $\{ac+b, ac-c+bd^2\}$ with respect to $\succ$ in $\mathbb{Q}[a,b,c,d]$. However, since the main variable x divides x^2, by definition of reduction2, $\mathrm{lc}_{\{a,b,c,d\}}(F_{x^2})$ is constrained by $\langle \mathrm{lc}_{\{a,b,c,d\}}(F_{x^2}) \rangle$. Therefore, we have to consider $\mathrm{lc}_{\{a,b,c,d\}}(F_{x^2})$ in $\mathbb{Q}[a,b,c,d]/\langle \mathrm{lc}_{\{a,b,c,d\}}(F_x) \rangle$. In this example, $\mathrm{lc}_{\{a,b,c,d\}}(F_{x^2})$ is the reduced Gröbner basis for the ideal generated by itself with respect to $\succ$ in $\mathbb{Q}[a,b,c,d]/\langle \mathrm{lc}_{\{a,b,c,d\}}(F_x) \rangle$.

We did not take care of all coefficients of weak reduced Gröbner bases in $K[\bar{A}]$, and thus a weak reduced Gröbner basis was not uniquely determined. Since the coefficient domain is a polynomial ring, we need some conditions to obtain a unique reduced Gröbner basis in $K[\bar{A}][\bar{X}]$. We call this reduced Gröbner basis **"strong reduced Gröbner basis"**.

Definition 3.5.5 (Strong reduced Gröbner bases). Let $\succ_{\bar{X},\bar{A}} := (\succ_1, \succ_2)$ be a block order and I an ideal in $K[\bar{A}][\bar{X}]$. Suppose that for $e \in \mathrm{lpp}_{\bar{A}}(G)$, $G_e = \{f \mid \mathrm{lpp}_{\bar{A}}(f) = e\}$. Then, a **strong reduced Gröbner basis** G for I with respect to $\succ_1$, $\succ_2$ and $\succ_{\bar{X},\bar{A}}$ is a Gröbner basis for I in $K[\bar{A}][\bar{X}]$ such that

1. for all $p \in G$, no monomial in $\mathrm{Mono}(p)$ lies in $\langle \mathrm{lm}(G \setminus \{p\}) \rangle$ in $K[\bar{A}, \bar{X}]$ with respect to $\succ_{\bar{X},\bar{A}}$,
2. for all $p \in G$, no monomial in $\mathrm{Mono}_{\bar{A}}(p)$ lies in $\langle \mathrm{lm}_{\bar{A}}(G \setminus \{p\}) \rangle$ in $K[\bar{A}][\bar{X}]$ with respect to $\succ_1$,
3. for all $e \in \mathrm{lpp}_{\bar{A}}(G)$, $\mathrm{lc}_{\bar{A}}(G_e)$ is the reduced Gröbner basis for an ideal generated by itself $\mathrm{lc}_{\bar{A}}(G_e)$ with respect to $\succ_2$ in the quotient ring $K[\bar{A}]/J_e$ where J_e is an ideal generated by $F = \{\mathrm{lc}_{\bar{A}}(g) \in K[\bar{A}] | g \in G \setminus G_e \text{ such that } \mathrm{lpp}_{\bar{A}}(g) | e\}$.
   (If $F = \emptyset$, $K[\bar{A}]/J_e = K[\bar{A}]$.)

Remark*: Let I be an ideal in $K[\bar{A}]$ and J an ideal in $K[\bar{A}]/I$. Then, it is possible to compute a reduced Gröbner basis for J with respect to a term order $\succ$ in $K[\bar{A}]/I$, and a reduced Gröbner basis in $K[\bar{A}]/I$ is uniquely determined; we can compute the reduced Gröbner basis G_J for J with respect to $\succ$ in $K[\bar{A}]$, and the reduced Gröbner basis G_I for I with respect to $\succ$ in $K[\bar{A}]$. Both G_J and G_I are unique. Next we compute all normal form of G_J by G_I. Then, $G_J \downarrow_{G_I}$ is the reduced Gröbner basis for J in $K[\bar{A}]/I$ and unique (where $G_J \downarrow_{G_I}$ is defined at the end of Algorithm 3.5.6 SRGB).

In order to consider the strong reduced Gröbner basis, let's see Example 3.4.1, again. In the example, we obtained a Gröbner basis $G = \{f_1 = a^2x - a, f_2 = (a^3 - a)x - a^2 + 1\}$ by Insa-Pauer, but G does not satisfy the property 3 of Definition 3.5.5. Since the set of all power products is $\mathrm{lpp}_{\bar{A}}(G) = \{x\}$, we have $G_x := \{f_1, f_2\}$ and $\mathrm{lc}_{\bar{A}}(G_x) := \{a^2, a^3 - a\}$. The reduced Gröbner basis for $\langle \mathrm{lc}_{\bar{A}}(G_x)\rangle$ is $\{a\}$. Since $\{a\} \neq \mathrm{lc}_{\bar{A}}(G_x)$, G is not a strong reduced Gröbner basis. However, we can construct the strong reduced Gröbner basis. Since $\langle a\rangle = \langle \mathrm{lc}_{\bar{A}}(G_x)\rangle$, a can be written as

$$a = c_1\,\mathrm{lc}_{\bar{A}}(f_1) + c_2\,\mathrm{lc}_{\bar{A}}(f_2),$$

where $c_1, c_2 \in \mathbb{Q}[a]$. In this case, $c_1 = a, c_2 = -1$. Now we can obtain a new polynomial g such that $\langle g\rangle = \langle G\rangle$, $\langle \mathrm{lm}_{\bar{A}}(g)\rangle = \langle \mathrm{lm}_{\bar{A}}(G)\rangle$ and $\{\mathrm{lc}_{\bar{A}}(g)\}$ is the reduced Gröbner basis for $\mathrm{lc}_{\bar{A}}(G_x)$.

$$g = c_1f_1 + c_2f_2 = af_1 - f_2 = ax + 1.$$

Therefore, $\{g\}$ is a strong reduced Gröbner basis.

The following algorithm returns a strong reduced Gröbner basis.

Algorithm 3.5.6. SRGB$(F, \succ_1, \succ_2)$ (Strong reduced Gröbner bases)

Input F: a finite set of polynomials in $K[\bar{A}][\bar{X}]$,
  $\succ_1$: a term order on $\mathrm{pp}(\bar{X})$,
  $\succ_2$: a term order on $\mathrm{pp}(\bar{A})$,
  $(\succ_{\bar{X},\bar{A}} := (\succ_1, \succ_2)$: a block order),
Output L: a strong reduced Gröbner basis of $\langle F\rangle$ with respect to the term orders $\succ_1, \succ_2$ and $\succ_{\{\bar{X},\bar{A}\}}$ in $K[\bar{A}][\bar{X}]$.
begin
$G \leftarrow$ Compute a weak reduced Gröbner basis for $\langle F\rangle$
$B \leftarrow \mathrm{lpp}_{\bar{A}}(G)$
$L \leftarrow \emptyset$
  while $B \neq \emptyset$ **do**
    Select **the lowest power product** p with respect to $\succ_1$ from B; $B \leftarrow B\backslash\{p\}$
    $G_p \leftarrow \{f \in F \mid \mathrm{lpp}_{\bar{A}}(f) = p\}$
    $G \leftarrow G\backslash G_p$
    $J_p \leftarrow \{\mathrm{lc}_{\bar{A}}(f) \mid f \in G \text{ s.t } \mathrm{lpp}_{\bar{A}}(f) | p\}$
    if $\mathrm{lc}_{\bar{A}}(G_p)$ is **NOT** the reduced Gröbner basis with respect to $\succ_2$ in $K[\bar{A}]/\langle J_p\rangle$ **then**
      $Q \leftarrow$ Compute Q such that $\langle Q\rangle = \langle G_p\rangle$, $\langle \mathrm{lm}_{\bar{A}}(Q)\rangle = \mathrm{lm}_{\bar{A}}(\langle G_p\rangle)$ and
       $\mathrm{lc}_{\bar{A}}(Q)$ is the reduced Gröbner basis for $\langle \mathrm{lc}_{\bar{A}}(G_p)\rangle$ with respect to $\succ_2$ in $K[\bar{A}]/\langle J_p\rangle$
      $L \leftarrow L \cup \{Q \downarrow_L\}$
      else-if
      $L \leftarrow L \cup \{G_p \downarrow_L\}$
      end-if
  end-while
   `return`(L)
end

In the algorithm, we used the notations $Q \downarrow_L$ and $G_p \downarrow_L$ where $Q, G_p, L \subset K[\bar{A}][\bar{X}]$. This meaning is the following.

$Q \downarrow_L :=$

begin
$S \leftarrow \emptyset$
while $Q \neq \emptyset$ **do**
Select q from Q; $Q \leftarrow Q \backslash \{q\}$; $q_1 \leftarrow q \downarrow_L$ (by Reduction1 and Reduction2)
  if $q_1 \neq 0$ **then**
    $S \leftarrow S \cup \{q_1\}$
  end-if
end-while
`return`(S)
end

Theorem 3.5.7. The algorithm SRGB terminates. The output forms a strong reduced Gröbner basis for $\langle F \rangle$ with respect to the term orders in $K[\bar{A}][\bar{X}]$.

Proof. We know that how to compute a weak reduced Gröbner basis G, and this step terminates. Since we have a Gröbner basis G, we have to check $\text{lc}_{\bar{A}}(G_p)$ where $p \in \text{lpp}_{\bar{A}}(G)$. If $\text{lc}_{\bar{A}}(G_p)$ is not a reduced Gröbner basis with respect to $\succ_2$ in $K[\bar{A}]/\langle J_p \rangle$, then, we have to compute the following;

$Q \leftarrow$ Compute Q such that $\langle Q \rangle = \langle G_p \rangle$, $\langle \text{lm}_{\bar{A}}(Q) \rangle = \text{lm}_{\bar{A}}(\langle G_p \rangle)$ and $\text{lc}_{\bar{A}}(Q)$ is the reduced Gröbner basis for $\langle \text{lc}_{\bar{A}}(G_p) \rangle$ with respect to $\succ_2$ in $K[\bar{A}]/\langle J_p \rangle$.

As we said in the **Remark***, it is possible to compute Q by "extended Gröbner bases algorithm Algorithm 2.3.9". This step clearly terminates. Since B is a finite set, the first **while-loop** terminates. Therefore, this algorithm terminates. In the **if**-part of the algorithm, if $\text{lc}_{\bar{A}}(G_p)$ is not the reduced Gröbner basis with respect to $\succ_2$ in $K[\bar{A}]/\langle J_p \rangle$, then the algorithm computes Q. Next the algorithm computes $Q \downarrow_L$. In fact, by reduction1, reduction2 and the weak reduced Gröbner basis, we have $\text{lm}_{\bar{A}}(Q) = \text{lm}_{\bar{A}}(Q \downarrow_L)$. That is, in this step, the algorithm reduces non-leading monomials of Q for the property 1,2 of Definition 3.5.5. By the same reasons, if $\text{lc}_{\bar{A}}(G_p)$ is the reduced Gröbner basis with respect to $\succ_2$ in $K[\bar{A}]/\langle J_p \rangle$, then we have $\text{lm}_{\bar{A}}(G_p) = \text{lm}_{\bar{A}}(G_p \downarrow_L)$, and $G_p \downarrow_L$ satisfies the property 1,2 of Definition 3.5.5. Therefore, this algorithm outputs a strong reduced Gröbner basis with respect to $\succ_1$, $\succ_2$ and $\succ_{\bar{X},\bar{A}}$. □

A strong reduced Gröbner bases have the following nice property.

Theorem 3.5.8. Let $\succ_{\bar{X},\bar{A}} := (\succ_1, \succ_2)$ be a block order on $\text{pp}(\bar{X}, \bar{A})$. Let I be an ideal in $K[\bar{A}][\bar{X}]$. Then, I has a unique strong reduced Gröbner basis.

Proof. Since the existence of strong reduced Gröbner bases was proved by Theorem 3.5.7, we prove the uniqueness. First we prove the following claim.

Claim 1 Let $\succ_{\bar{X},\bar{A}} := (\succ_1, \succ_2)$ be a block order on $\text{pp}(\bar{X}, \bar{A})$ and I an ideal in $K[\bar{A}][\bar{X}]$. Let G_1 and G_2 be strong reduced Gröbner bases for I with respect to $\succ_1$ and $\succ_{\bar{X},\bar{A}}$. Then, $\text{lm}(G_1) = \text{lm}(G_2)$. Namely, the set of all leading monomials of strong reduced Gröbner bases for I is unique.

(*Proof of Claim 1*) Assume that G_1 and G_2 are strong reduced Gröbner bases for I in $K[\bar{A}][\bar{X}]$. We set $G_1 = \{g_1, \ldots, g_s\}$, $G_2 = \{h_1, \ldots, h_p\}$. W.l.o.g., $\text{lpp}_{\bar{A}}(g_1) = \ldots = \text{lpp}_{\bar{A}}(g_k)$ is the lowest leading power product of G_1 with respect to $\succ_1$ for $1 \leq k \leq s$, and $\text{lpp}_{\bar{A}}(h_1) = \ldots = \text{lpp}_{\bar{A}}(h_l)$ is the lowest leading power product of G_2 with respect to $\succ_1$ for $1 \leq l \leq p$. If $\mathrm{lpp}_{\bar{A}}(g_1) \succ_1 \mathrm{lpp}_{\bar{A}}(h_1)$ ($\mathrm{lpp}_{\bar{A}}(g_1)$ is bigger than $\mathrm{lpp}_{\bar{A}}(h_1)$), h_1 can not be in $\langle G_1 \rangle$, because there is no element such that $\mathrm{lpp}_{\bar{A}}(g_i) | \mathrm{lpp}_{\bar{A}}(h_1)$ where $g_i \in G_1$. However, by $\langle G_1 \rangle = \langle G_2 \rangle$, we have $h_1 \in \langle G_1 \rangle$. Hence, $\mathrm{lpp}_{\bar{A}}(g_1) \succeq \mathrm{lpp}_{\bar{A}}(h_1)$ (by the order $\succ_1$). By the same reason, we have also $\mathrm{lpp}_{\bar{A}}(h_1) \succeq \mathrm{lpp}_{\bar{A}}(g_1)$ (by the order $\succ_1$). Therefore, $\mathrm{lpp}_{\bar{A}}(h_1) = \mathrm{lpp}_{\bar{A}}(g_1)$. We have two sets

$$\begin{aligned}\{\mathrm{lm}_{\bar{A}}(g_1), \ldots, \mathrm{lm}_{\bar{A}}(g_k)\} &= \{\mathrm{lc}_{\bar{A}}(g_1)\,\mathrm{lpp}_{\bar{A}}(g_1), \ldots, \mathrm{lc}_{\bar{A}}(g_k)\,\mathrm{lpp}_{\bar{A}}(g_1)\},\\ \{\mathrm{lm}_{\bar{A}}(h_1), \ldots, \mathrm{lm}_{\bar{A}}(h_p)\} &= \{\mathrm{lc}_{\bar{A}}(h_1)\,\mathrm{lpp}_{\bar{A}}(g_1), \ldots, \mathrm{lc}_{\bar{A}}(h_p)\,\mathrm{lpp}_{\bar{A}}(g_1)\}.\end{aligned}$$

Since G_1, G_2 are strong reduced Gröbner bases for I with respect to $\succ_1$, $\succ_2$ and $\succ_{\bar{X},\bar{A}}$ in $K[\bar{A}][\bar{X}]$, $\{\mathrm{lc}_{\bar{A}}(g_1), \ldots, \mathrm{lc}_{\bar{A}}(g_k)\}$ is the reduced Gröbner basis for an ideal generated by itself in $K[\bar{A}]$, and $\{\mathrm{lc}_{\bar{A}}(h_1), \ldots, \mathrm{lc}_{\bar{A}}(h_l)\}$ is also the reduced Gröbner basis for an ideal generated by itself in $K[\bar{A}]$. By the property of Gröbner bases G_1, G_2 and $\mathrm{lpp}_{\bar{A}}(h_1) = \mathrm{lpp}_{\bar{A}}(g_1)$, we have the following relations

$$\begin{aligned}\mathrm{lm}_{\bar{A}}(h_{j_1}) &= \alpha_1 \,\mathrm{lm}_{\bar{A}}(g_1) + \cdots + \alpha_k \,\mathrm{lm}_{\bar{A}}(g_k)\\ &= \alpha_1 \,\mathrm{lc}_{\bar{A}}(g_1)\,\mathrm{lpp}_{\bar{A}}(g_1) + \cdots + \alpha_k \,\mathrm{lc}_{\bar{A}}(g_k)\,\mathrm{lpp}_{\bar{A}}(g_1),\\ \mathrm{lm}_{\bar{A}}(g_{j_2}) &= \beta_1 \,\mathrm{lm}_{\bar{A}}(h_1) + \cdots + \beta_l \,\mathrm{lm}_{\bar{A}}(h_l)\\ &= \beta_1 \,\mathrm{lc}_{\bar{A}}(h_1)\,\mathrm{lpp}_{\bar{A}}(g_1) + \cdots + \beta_l \,\mathrm{lc}_{\bar{A}}(h_l)\,\mathrm{lpp}_{\bar{A}}(g_1),\end{aligned}$$

where $\alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_l \in K[\bar{A}]$, $1 \leq j_1 \leq k$ and $1 \leq j_2 \leq l$. Hence we can say $\langle \mathrm{lc}_{\bar{A}}(g_1), \ldots, \mathrm{lc}_{\bar{A}}(g_k)\rangle = \langle \mathrm{lc}_{\bar{A}}(h_1), \ldots, \mathrm{lc}_{\bar{A}}(h_l)\rangle$. Since the two sets $\{\mathrm{lc}_{\bar{A}}(g_1), \ldots, \mathrm{lc}_{\bar{A}}(g_k)\}$ and $\{\mathrm{lc}_{\bar{A}}(h_1), \ldots, \mathrm{lc}_{\bar{A}}(h_p)\}$ are the reduced Gröbner bases with respect to $\succ_2$ in $K[\bar{A}]$, we have $\{\mathrm{lc}_{\bar{A}}(g_1), \ldots, \mathrm{lc}_{\bar{A}}(g_k)\} = \{\mathrm{lc}_{\bar{A}}(h_1), \ldots, \mathrm{lc}_{\bar{A}}(h_l)\}$. Therefore we have

$$\{\mathrm{lm}_{\bar{A}}(g_1), \ldots, \mathrm{lm}_{\bar{A}}(g_k)\} = \{\mathrm{lm}_{\bar{A}}(h_1), \ldots, \mathrm{lm}_{\bar{A}}(h_l)\}.$$

Next we consider two sets $G_{11} := G_1 \backslash \{g_1, \ldots, g_k\}$ and $G_{21} := G_2 \backslash \{h_1, \ldots, h_l\}$. W.l.o.g., $\mathrm{lpp}_{\bar{A}}(g_{k+1}) = \ldots = \mathrm{lpp}_{\bar{A}}(g_{k_1})$ is the lowest leading power product of G_{11} with respect to $\succ_1$ for $2 \leq k_1 \leq s$. That is, $g_{k+1}, \ldots, g_{k_1} \in G_{11} \subseteq G_1$. Since G_1 is a strong reduced Gröbner basis, $\mathrm{lm}_{\bar{A}}(g_{k+1}), \ldots, \mathrm{lm}_{\bar{A}}(g_{k_1})$ can not be reduced by $\mathrm{lm}_{\bar{A}}(g_1), \ldots, \mathrm{lm}_{\bar{A}}(g_k)$. W.l.o.g., $\mathrm{lpp}_{\bar{A}}(h_{l+1}) = \ldots = \mathrm{lpp}_{\bar{A}}(h_{l_1})$ is the lowest leading power product of G_{21} with respect to $\succ_1$ for $2 \leq l_1 \leq p$. That is, $h_{l+1}, \ldots, h_{l_1} \in G_{21} \subseteq G_2$. By the same reason above, we have $\mathrm{lpp}_{\bar{A}}(g_{k+1}) = \mathrm{lpp}_{\bar{A}}(h_{l+1})$ and $\langle \mathrm{lc}_{\bar{A}}(g_{k+1}), \ldots, \mathrm{lc}_{\bar{A}}(g_{k_1})\rangle = \langle \mathrm{lc}_{\bar{A}}(h_{l+1}), \ldots, \mathrm{lc}_{\bar{A}}(h_{l_1})\rangle$. We know that $\mathrm{lpp}_{\bar{A}}(g_1)$ is the lowest leading power product of G_1 and G_2. If $\mathrm{lpp}_{\bar{A}}(g_1)$ does not divide $\mathrm{lpp}_{\bar{A}}(g_{k+1})$, then by the same reason above,

$$\{\mathrm{lm}_{\bar{A}}(g_{k+1}), \ldots, \mathrm{lm}_{\bar{A}}(g_{k_1})\} = \{\mathrm{lm}_{\bar{A}}(h_{l+1}), \ldots, \mathrm{lm}_{\bar{A}}(h_{l_1})\}.$$

If $\mathrm{lpp}_{\bar{A}}(g_1)$ divide $\mathrm{lpp}_{\bar{A}}(g_{k+1})$, then G_1 and G_2 have the same set

$$J = \{\mathrm{lc}_{\bar{A}}(g_1), \ldots, \mathrm{lc}_{\bar{A}}(g_k)\} = \{\mathrm{lc}_{\bar{A}}(h_1), \ldots, \mathrm{lc}_{\bar{A}}(h_l)\}.$$

By the **Remark***, the reduced Gröbner basis for $\langle \mathrm{lc}_{\bar{A}}(g_{k+1}), \ldots, \mathrm{lc}_{\bar{A}}(g_{k_1})\rangle$ is unique in $K[\bar{A}]/J$. Since G_1 and G_2 are strong reduced Gröbner bases, we have

$$\{\mathrm{lm}_{\bar{A}}(g_{k+1}), \ldots, \mathrm{lm}_{\bar{A}}(g_{k_1})\} = \{\mathrm{lm}_{\bar{A}}(h_{l+1}), \ldots, \mathrm{lm}_{\bar{A}}(h_{l_1})\}.$$

By the Hilbert basis theorem, G_1 and G_2 have finite many elements. Therefore, repeat the same procedure, then we have $\mathrm{lm}_{\bar{A}}(G_1) = \mathrm{lm}_{\bar{A}}(G_2)$. □

Suppose that G_1 and G_2 are strong reduced Gröbner bases for I. Then, by the claim 1, we have $\mathrm{lm}_{\bar{A}}(G_1) = \mathrm{lm}_{\bar{A}}(G_2)$. Thus, given $g_1 \in G_1$, there is $g_2 \in G_2$ such that

$\mathrm{lm}_{\bar{A}}(g_1) = \mathrm{lm}_{\bar{A}}(g_2)$. If we can show that $g_1 = g_2$, it will follow that $G_1 = G_2$, and uniqueness will be proved.
To show $g_1 = g_2$, consider $g_1 - g_2$. This is in I, and since G_1 is a Gröbner basis, it follows that $g_1 - g_2 \xrightarrow{r1*}_{G_1} \circ \xrightarrow{r2*}_{G_1} \circ \cdots \circ \xrightarrow{r1*}_{G_1} 0$ (by Reduction1 and Reduction2). However, we also know $\mathrm{lm}_{\bar{A}}(g_1) = \mathrm{lm}_{\bar{A}}(g_2)$. Hence, these monomials cancel in $g_1 - g_2$, and the remaining monomials are divisible by none of $\mathrm{lm}_{\bar{A}}(G_1) = \mathrm{lm}_{\bar{A}}(G_2)$ since G_1 and G_2 are reduced. This shows that $g_1 - g_2 \xrightarrow{r1*}_{G_1} \circ \xrightarrow{r2*}_{G_1} \circ \cdots \circ \xrightarrow{r1*}_{G_1} g_1 - g_2$, and then $g_1 - g_2 = 0$ follows. This completes the proof. □

3.6 Computation examples

The algorithms WRGB (with GröbnerBasisB) have been implemented for the case $K = \mathbb{Q}$ in the computer algebra system Risa/Asir by the author. In this section, we give three easy examples of reduced Gröbner bases .

Example 3.6.1. Let a, x, y be variables and $f_1 = (a-1)x + y^2, f_2 = ay + a$ polynomials in $\mathbb{Q}[a][x, y]$. We compute a reduced Gröbner basis for $\langle f_1, f_2\rangle$ with respect to the lexicographic order with $x \succ y$ in $\mathbb{Q}[a][x, y]$.
By the procedure of WRGB, first we compute the reduced Gröbner basis G for $\langle f_1, f_2\rangle$ with respect to a block order $\succ_{\{x,y\},\{a\}}$ in $\mathbb{Q}[a, x, y]$. The reduced Gröbner basis G in $\mathbb{Q}[a, x, y]$ is the following

$$G = \left\{g_1 = ay + a, g_2 = ax - x + y^2, g_3 = -xy - x + y^3 + y^2\right\}.$$

Second, we need to check whether there exists a polynomial $p \in G$ which can be reduced by $G\backslash\{p\}$ or not.
We have $\mathrm{lpp}_{\{a\}}(g_1)|\,\mathrm{lpp}_{\{a\}}(g_3)$, $\mathrm{lpp}_{\{a\}}(g_2)|\,\mathrm{lpp}_{\{a\}}(g_3)$ and $\mathrm{lc}_{\{a\}}(g_3) = \mathrm{lc}_{\{a\}}(g_1) - \mathrm{lc}_{\{a\}}(g_2) = -a + (a-1) = -1$, therefore g_3 can be reduced as follows

$$g_3 \xrightarrow{r1}_{\{g_1,g_2\}} g_3 - (-xg_1 + yg_2) = ax - x + y^2 = g_2.$$

The set $\{g_1, g_2\}$ is a weak reduced Gröbner basis for $\langle f_1, f_2\rangle$ in $\mathbb{Q}[a][x, y]$. Actually, $\{g_1, g_2\}$ is the strong reduced Gröbner basis, too.

In the following two examples, we compare three algorithms GröbnerBasisB, Insa-Pauer and WRGB. We used a PC with [CPU: Pentium M 1.73 GHZ, OS: Windows XP].

Example 3.6.2. Let a, b, x, y, z be variables and $F = \{bxz + ay + a, y + by + 3, ay^2z + bz + b, ay + a\}$ in $\mathbb{Q}[a, b][x, y, z]$. We have the graded lexicographic order with $x \succ y \succ z$. We compute a Gröbner basis for $\langle F\rangle$ in $\mathbb{Q}[a, b][x, y, z]$ by three algorithms, Insa-Pauer, GröbnerBasisB, and WRGB (with GröbnerBasisB).

1. By Insa-Pauer, we have the following Gröbner basis

```
[(b-2)*a,(a+b)*z+b,(b+1)*y+3,b*x].
(cputime: 0.07851sec)
```

   This list has four polynomials.
2. By GröbnerBasisB, we have the following Gröbner basis

```
[(b-2)*a,(a+b)*z+b,(-b^2+2*b)*z-b^2+2*b,(b+1)*y+3,a*y+a,b*x,
 a*x,(z+1)*y+(-b+3)*z-b+3,(y+3)*x].
(cputime: 0sec)
```

This list has eight polynomials.

3. By WRGB, we have the following weak reduced Gröbner basis

```
[(b-2)*a,(a+b)*z+b,(b+1)*y+3,b*x].
(cputime: 0.01563sec)
```

(Actually, this is the strong Gröbner basis, too.) This list is same as the first list which was obtained by Insa-Pauer.

Example 3.6.3. Let a, b, x, y, z be variables and $F = \{ax^2z+ay+a, axz+b, (a+1)xz+ab\}$ in $\mathbb{Q}[a, b][x, y, z]$. We have the lexicographic order with $x \succ y \succ z$. We compute a Gröbner basis for $\langle F \rangle$ in $\mathbb{Q}[a, b][x, y, z]$ by three algorithms Insa-Pauer, GröbnerBasisB, and WRGB (with GröbnerBasisB).

1. By Insa-Pauer, we have the following Gröbner basis

```
[b*a^2-b*a-b,-b*x+a*y+a,(a+1)*z*x+b*a,a*z*x+b,
 a*z*y+a*z+b^2*a-b^2,(-a^3+a^2+a)*y-a^3+a^2+a].
(cputime: 0.04688sec)
```

This list has six polynomials.

2. By GröbnerBasisB, we have the following Gröbner basis

```
[-b*a^2+b*a+b,(a^3-a^2-a)*y+a^3-a^2-a,b*z*y+b*z-b^3*a+2*b^3,
 a*z*y+a*z+b^2*a-b^2,-b*x+a*y+a,-z*x-b*a+b].
(cputime: 0sec)
```

This list has six polynomials.

3. By WRGB, we have the following reduced Gröbner basis

```
[-b*a^2+b*a+b,(a^3-a^2-a)*y+a^3-a^2-a,a*z*y+a*z+b^2*a-b^2,
 -b*x+a*y+a,-z*x-b*a+b].
(cputime: 0.01563sec)
```

This list has five polynomials.

Chapter 4

Comprehensive Gröbner bases and comprehensive Gröbner systems

In this chapter, we describe comprehensive Gröbner bases and comprehensive Gröbner systems. First, we describe the history and recent trend. Second, we give the definitions of comprehensive Gröbner bases and comprehensive Gröbner systems. Third, we treat the Suzuki-Sato algorithms for computing comprehensive Gröbner bases and comprehensive Gröbner systems. (In this thesis, our algorithms in various domains are based on the Suzuki-Sato algorithm.) Finally, we introduce the software which computes comprehensive Gröbner bases and comprehensive Gröbner systems.

4.1 Introduction

Comprehensive Gröbner bases for parametric polynomial ideals were introduced, constructed, and studied by Weispfenning in 1992. Since then comprehensive Gröbner bases have been studied and implemented in the several computer algebra systems.
A comprehensive Gröbner basis is a finite subsets G of a parametric polynomial ideal I such that $\sigma(I)$ constitutes a Gröbner basis of the ideal generated by $\sigma(I)$ under all specialization σ of the parameters in arbitrary fields.

Roughly speaking, a comprehensive Gröbner system is a parametric Gröbner basis with parameter spaces. If we take a parameter space $\mathbb{P}$ and its set of parametric polynomials G from a comprehensive Gröbner systems for a parametric polynomial ideal I, then $\sigma(G)$ constitutes a Gröbner basis of the ideal generated by $\sigma(I)$ under the specialization σ with respect to the parameter space $\mathbb{P}$ of the parameters. (We introduce a definition of comprehensive Gröbner bases and comprehensive Gröbner systems in the next section.) These concept has found numerous applications.

After Weispfenning's paper was published, Dolzman and Sturm have implemented and published the software [DS97]. However, there was no big development about comprehensive Gröbner bases and comprehensive Gröbner systems for ten years. Recently, the big developments were made by Montes, Sato, Suzuki and Weispfenning.

[Efficient computation]
• Montes published the new algorithm for computing comprehensive Gröbner systems and its software in 2002 and 2006 [Mon02, MM05, MM06].
• Suzuki and Sato publish the new algorithm for computing comprehensive Gröbner bases and comprehensive Gröbner systems in 2006 [SS06].

[Theory]

• Suzuki and Sato presented an alternative definition of comprehensive Gröbner bases in terms of Gröbner bases in polynomial rings over commutative von Neumann regular rings [SS02, SS03]. This Gröbner basis is called "alternative comprehensive Gröbner basis". Alternative comprehensive Gröbner bases have the following nice properties, which do not hold in standard comprehensive Gröbner bases ;

1. There is a canonical form of an alternative comprehensive Gröbner basis in a natural way.
2. We can use reductions of an alternative comprehensive Gröbner basis.

• Weispfenning presented a concept of canonical comprehensive Gröbner bases under very general assumptions on the parameter ring [Wei02a, Wei03]. After this paper was published, this result was applied for improving Montes' algorithm by Montes [MM06].

In this chapter we mainly describe the Suzuki-Sato algorithms [SS06] for computing comprehensive Gröbner bases and comprehensive Gröbner systems, because the Suzuki-sato algorithms are faster then other existing algorithms. In the final section of this chapter, we introduce the software for computing comprehensive Gröbner bases and systems; REDLOG in Reduce, Montes' program in Maple and Suzuki's program in Maple, Risa/Asir, Mathematica, singular and the author's package in Risa/Asir.

In chapter 5, we improve the Suzuki-Sato algorithms for computing comprehensive Gröbner systems. In chapter 8 and 9, we apply the Suzuki-Sato algorithm for computing comprehensive Gröbner bases and comprehensive Gröbner systems in rings of differential operators and $K[\bar{A}][\bar{X}]$-module.

4.2 Comprehensive Gröbner bases and comprehensive Gröbner systems

In this section, we give definitions and examples of comprehensive Gröbner bases and comprehensive Gröbner systems. In general, comprehensive Gröbner bases and comprehensive Gröbner systems are called "**parametric Gröbner bases**".

4.2.1 Comprehensive Gröbner systems

Here, we give a definition of comprehensive Gröbner systems and their examples. In this thesis, we use the following notations for the canonical specialization homomorphism.

For arbitrary $\bar{a} \in L^m$, we can define the canonical specialization homomorphism $\sigma_{\bar{a}} : K[\bar{A}] \to L$ induce by $\bar{a}$, and we can naturally extend it to $\sigma_{\bar{a}} : K[\bar{A}][\bar{X}] \to L[\bar{X}]$.

Definition 4.2.1 (Comprehensive Gröbner Systems). Let F be a subset of $K[\bar{A}][\bar{X}]$, $\mathcal{A}_1, \ldots, \mathcal{A}_l$ algebraically constructible subsets of L^m and $G_1, \ldots, G_l$ subsets of $K[\bar{A}][\bar{X}]$. Let $\mathcal{S}$ be a subset of L^m such that $\mathcal{S} \subseteq \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_l$. A finite set $\mathcal{G} = \{(\mathcal{A}_1, G_1), \ldots, (\mathcal{A}_l, G_l)\}$ of pairs is called a **comprehensive Gröbner system** on $\mathcal{S}$ for $\langle F \rangle$ if $\sigma_{\bar{a}}(G_i)$ is a Gröbner basis of the ideal $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}]$ for each $i = 1, \ldots, l$ and $\bar{a} \in \mathcal{A}_i$. Each $(\mathcal{A}_i, G_i)$ is called a **segment** of $\mathcal{G}$. We simply say $\mathcal{G}$ is a comprehensive Gröbner system for $\langle F \rangle$ if $\mathcal{S} = L^m$.

In this thesis, we use an algebraically constructible set that has a form

$$\mathbb{V}(f_1,\dots,f_k)\backslash\mathbb{V}(g_1,\dots,g_l)\subseteq L^m$$

where $f_1,\dots,f_k,g_1,\dots,g_l\in K[\bar{A}]$.

We give examples of comprehensive Gröbner systems in the following.

Example 4.2.2. Let $F=\{ax^2y+y, bx^2y^2+ax+y\}\subset\mathbb{Q}[a,b][x,y]$, a,b parameters, x,y variables and $\succ$ the lexicographic order such that $x\succ y$. Then, a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$ is

$$\mathcal{G}_1=\left\{\left(\mathbb{Q}^2\setminus\mathbb{V}(a,b),\{a^2x-by^2+ay,-b^2y^5+2bay^4-a^2y^3-a^3y\}\right),\left(\mathbb{V}(a,b),\{y\}\right)\right\}.$$

This meaning is the following;

$$\begin{cases}\{a^2x-by^2+ay,-b^2y^5+2bay^4-a^2y^3-a^3y\}, & \text{if } a\neq 0, b\neq 0,\\ \{y\}, & \text{if } a=b=0.\end{cases}$$

In the comprehensive Gröbner system $\mathcal{G}_1$, the parameter spaces $\mathbb{Q}^2\setminus\mathbb{V}(a,b)$ and $\mathbb{V}(a,b)$ are disjoint, i.e., $(\mathbb{Q}^2\setminus\mathbb{V}(a,b))\cap\mathbb{V}(a,b)=\emptyset$. The following set $\mathcal{G}_2$ is also a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$;

$$\mathcal{G}_2=\left\{\begin{array}{l}\left(\mathbb{Q}^2\setminus\mathbb{V}(ab),\{-b^2y^5+2aby^4-a^2y^3-a^3y,a^2x-by^2+ay\}\right),\\ \left(\mathbb{V}(b)\backslash\mathbb{V}(a),\{-y^2x+y,ax+y,y^3+ay\}\right),\left(\mathbb{V}(a),\{y\}\right)\end{array}\right\}.$$

We can understand the set as follows.

$$\begin{cases}\{-b^2y^5+2aby^4-a^2y^3-a^3y,a^2x-by^2+ay\}, & \text{if } ab\neq 0,\\ \{-y^2x+y,ax+y,y^3+ay\} & \text{if } b=0, a\neq 0,\\ \{y\}, & \text{if } a=0.\end{cases}$$

In the comprehensive Gröbner system $\mathcal{G}_2$, the parameter spaces are not disjoint, however $\mathcal{G}_2$ is a comprehensive Gröbner systems, too. Actually, we have

$$\mathbb{Q}^2=\Big(\mathbb{Q}^2\setminus\mathbb{V}(ab)\Big)\cup\Big(\mathbb{V}(b)\backslash\mathbb{V}(a)\Big)\cup\big(\mathbb{V}(a)\big).$$

Weispfenning introduced an algorithm for computing comprehensive Gröbner systems in [Wei92]. The idea of the algorithm is very natural and simple. For instance, let $F=\{f_1=(a-1)x+y^2, f_2=ay+a\}\subset\mathbb{Q}[a][x,y]$, x,y variables, a parameter and $\succ$ the lexicographic order such that $x\succ y$. The idea is the following; if the leading coefficient of a polynomial $f\in K[\bar{A}][\bar{X}]$ vanishes under the specialization $\sigma_{\bar{b}}$ where $\bar{b}\in\mathbb{C}^{|\bar{A}|}$, then $\mathrm{lpp}_{\bar{A}}(f)\neq\mathrm{lpp}(\sigma_{\bar{b}}(f))$. Therefore, we need the case distinctions, i.e., we need to compute the case $\bar{A}=\bar{b}$ and the case $\bar{A}\neq\bar{b}$. That is, we have to compute the S-polynomials in both cases.

In the example, first we need to consider the cases $a=0$ and $a\neq 0$. If $a=0$, then we have $\sigma_{a=0}(F)=\{-x+y^2\}$ which is always the Gröbner basis of $\sigma_{a=0}(F)$. If $a\neq 0$, we need again a case distinction "case $a=1$" and "case $a\neq 1$". When $a=1$,then $\sigma_{a=1}(f_1)=y^2$ and $\sigma_{a=1}(f_2)=y+1$. In this case, $\{1\}$ is the Gröbner basis for $\langle\sigma_{a=1}(F)\rangle=\langle 1\rangle$. Finally, we need to consider the case $a\neq 0,1$. In this case, by the first Buchberger's criterion, F is a Gröbner basis. Therefore, a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$ is

$$\Big\{(\mathbb{Q}\setminus\{0,1\},F),(\mathbb{V}(a-1),\{1\}),(\mathbb{V}(a),\{-x+y^2\})\Big\}.$$

In general, in this example, the case $(\mathbb{Q}\setminus\{0,1\}, F)$ is called "generic" case. The other cases are called "singular" cases. See Figure 4.1.

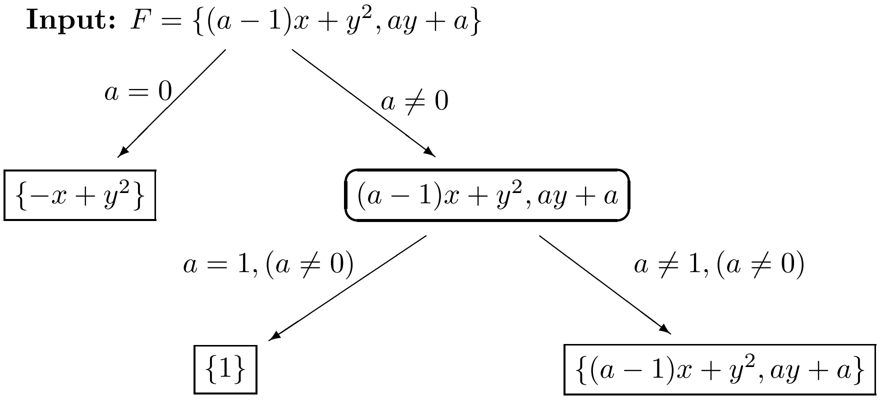

Figure 4.1

4.2.2 Comprehensive Gröbner bases

Here, we give a definition of comprehensive Gröbner bases and their examples. The existence of comprehensive Gröbner basis was shown in Weispfenning [Wei92]; the paper provided moreover a construction of a comprehensive Gröbner basis from a set of parametric polynomials and term order on $\mathrm{pp}(\bar{X})$. Actually, the construction of comprehensive Gröbner bases is basically computing comprehensive Gröbner systems. First we give the definition as follows.

Definition 4.2.3 (Comprehensive Gröbner bases). Let F and G be sets of polynomials in $K[\bar{A}][\bar{X}]$. $G \subset \langle F \rangle$ is called a **comprehensive Gröbner basis** for $\langle F \rangle$ if $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ for each $\bar{a} \in L^m$.

We already saw comprehensive Gröbner systems in previous section which have conditions of parameters (parameter spaces). However, comprehensive Gröbner bases do not have any condition of parameters. A comprehensive Gröbner basis is a set of polynomials. In this point, comprehensive Gröbner bases are different from comprehensive Gröbner systems. We give an example of comprehensive Gröbner bases.

Example 4.2.4. Let $F = \{ax^2y + y, bx^2y^2 + ax + y\} \subset \mathbb{Q}[a,b][x,y]$, a, b parameters, x, y variables and $\succ$ the lexicographic order such that $x \succ y$. Then, a comprehensive Gröbner basis for $\langle F \rangle$ with respect to $\succ$ is

$$\begin{aligned} G = \Big\{ & bxy^3 - axy^2 + ay, a^2x - by^2 + ay, -b^2y^5 + 2aby^4 - a^2y^3 - a^3y, -bx^3y^3 - xy^2 + y, \\ & bx^2y^2 + ax + y, bx^2y^4 + bxy^3 + y^3 + ay, ax^2y + y \Big\}. \end{aligned}$$

Even if we substitute arbitrary values for the parameter a, b of the set G, the set $\{\sigma(G)\}$ is always a Gröbner basis for $\langle \sigma(F) \rangle$ with respect to $\succ$.

Weispfenning introduced an algorithm for computing comprehensive Gröbner bases in [Wei92]. The idea of the algorithm is the same as the algorithm for computing comprehensive Gröbner systems. However, in order to construct comprehensive Gröbner bases we need the following concept.

Definition 4.2.5. Let F be a subset of $K[\bar{A}][\bar{X}]$, $\mathcal{A}_1, \dots, \mathcal{A}_l$ algebraically constructible subsets of L^m and $G_1, \dots, G_l$ subsets of $K[\bar{A}][\bar{X}]$. A finite set $\mathcal{G} = \{(\mathcal{A}_1, G_1), \dots, (\mathcal{A}_l, G_l)\}$ of pairs is called **faithful** for $\langle F \rangle$ if $G_i \subset \langle F \rangle$ for each $i = 1, \dots, l$.

If we have a faithful comprehensive Gröbner system $\{(\mathcal{A}_1, G_1), \dots, (\mathcal{A}_l, G_l)\}$ for $\langle F \rangle$ where the notations are from Definition 4.2.5, then by the definition of comprehensive Gröbner bases, $G_1 \cup \cdots \cup G_l$ is a comprehensive Gröbner basis for $\langle F \rangle$. The idea of Weispfenning's algorithm for computing comprehensive Gröbner bases is almost same as the algorithm for computing comprehensive Gröbner systems.

Let us consider the same example of section 4.2.1. We have $F = \{f_1 = (a-1)x+y^2, f_2 = ay + a\}$ in $\mathbb{Q}[a][x, y]$ and $\succ$ the lexicographic order such that $x \succ y$. First we have to compute the S-polynomial of the pair (f_1, f_2). As we said in section 4.2.1., we need case distinctions. If $a = 0$, then $\mathrm{lm}_{\{a\}}(f_1) = (a-1)x$ and $\mathrm{lm}_{\{a\}}(f_2) = 0$. Hence, in this case, F is a Gröbner basis. Next, if $a = 1$, then $\mathrm{lm}_{\{a\}}(f_1) = y^2$ and $\mathrm{lm}_{\{a\}}(f_2) = ay$. Therefore,

$$\begin{aligned}
\mathsf{Spoly1}(f_1, f_2) &= af_1 - yf_2 \\
&= a(a-1)x + ay^2 - ay^2 - ay \\
&= a(a-1)x - ay \\
&\xrightarrow{r1}_{ay+a} a(a-1)x - a = f_3
\end{aligned}$$

Note that we need the tail of f_1 "$(a-1)x$", because we need to compute a faithful comprehensive Gröbner systems. In this point, this computation method is different, and more complicated than the method of computing comprehensive Gröbner systems.

We have to continue the computation, and we need to compute the S-polynomials of pairs (f_1, f_3) and (f_2, f_3). However, we have $\mathrm{lm}_{\{a\}}(f_3) = a \neq 0$. Therefore, $\{f_1, f_2, f_3\}$ is a Gröbner basis for $\langle F \rangle$, because a is in the coefficient domain $\mathbb{Q}[a, b]$. (If we substitute an arbitrary non-zero value for the parameter a, then this Gröbner basis is $\{1\}$.) Finally, we have to consider the case $a \neq 0, 1$. In this case, by Buchberger's first criterion, F is a Gröbner basis for $\langle F \rangle$ with respect to $\succ$. Therefore, a comprehensive Gröbner basis for $\langle F \rangle$ with respect to $\succ$ is the following;

$$F \cup \{f_3\}.$$

See Figure 4.2, and compare Figure 4.1 and Figure 4.2.

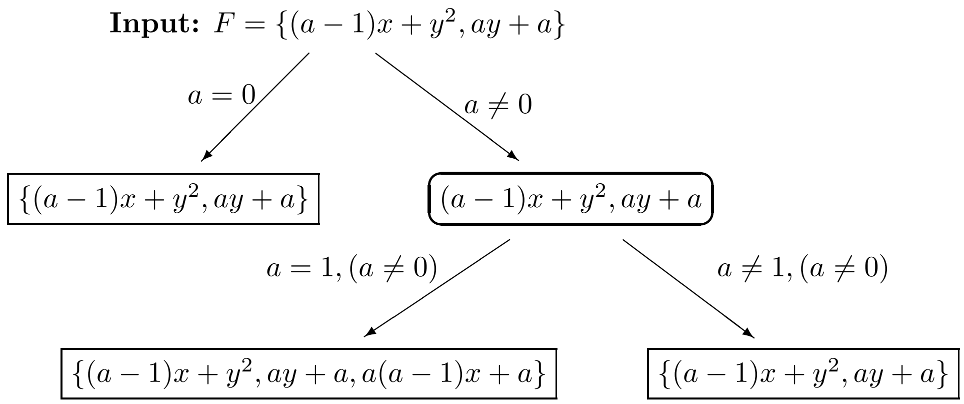

Figure 4.2

4.3 Stability of ideals

Here, we describe the stability of ideals under specialization in $K[\bar{A}][\bar{X}]$. Mainly, we introduce Kalkbrener's theory [Kal97].

As we said previous section, every ring homomorphism $\sigma : K[\bar{A}] \to L$ extends naturally to a homomorphism $\sigma : K[\bar{A}][\bar{X}] \to L[\bar{X}]$. The image under σ of an ideal $I \subseteq K[\bar{A}][\bar{X}]$ generates the extension $\sigma(I) := \{\sigma(f) : f \in I\} \subseteq L[\bar{X}]$.

Definition 4.3.1. We call an ideal $I \subseteq K[\bar{A}][\bar{X}]$ **stable** under the ring homomorphism σ and a term order $\succ$ if it satisfies

$$\sigma(\mathrm{lm}_{\bar{A}}(I)) = \mathrm{lm}(\sigma(I))$$

where $\sigma(\mathrm{lm}_{\bar{A}}(I)) := \{\sigma(\mathrm{lm}_{\bar{A}}(f)) : f \in I\}$ and $\mathrm{lm}(\sigma(I)) := \{\mathrm{lm}(f) : f \in \sigma(I)\}$.

In several papers [Bec94, Gia87, Kal97, EGT01, Sat05], the stability of ideals under specialization was studied. The following theorem is the key theorem for constructing the Suzuki-Sato algorithm (also our new algorithm) for computing comprehensive Gröbner systems and comprehensive Gröbner bases.

Theorem 4.3.2 (Kalkbrener (1997) [Kal97]). Let π be a ring homomorphism from $K[\bar{A}]$ to L, I an ideal in $K[\bar{A}][\bar{X}]$ and $G = \{g_1, \dots, g_s\}$ a Gröbner basis of I with respect to a term order $\succ$. We assume that the g_i's are ordered in such a way that there exists an $r \in \{1, \dots, s\}$ with $\pi(\mathrm{lc}_{\bar{A}}(g_i)) \neq 0$ for $i \in \{1, \dots, r\}$ and $\pi(\mathrm{lc}_{\bar{A}}(g_i)) = 0$ for $i \in \{r+1, \dots, s\}$. Then the following three conditions are equivalent.

1. I is stable under σ and $\succ$.
2. $\{\pi(g_1), \dots, \pi(g_r)\}$ is a Gröbner basis of $\sigma(I)$ with respect to the term order $\succ$.
3. For every $i \in \{r+1, \dots, s\}$, $\pi(g_i)$ is reducible to 0 modulo $\{\pi(g_1), \dots, \pi(g_r)\}$ in $L[\bar{X}]$.

Proof. We know that $\{\pi(g_1), \dots, \pi(g_q)\}$ is a Gröbner basis of $\pi(I)$ in $K[\bar{X}]$ if and only if

$$\{\pi(\mathrm{lm}(g)) | g \in G\} = \mathrm{lm}(\pi(I)).$$

Since

$$\{\pi(\mathrm{lm}(g)) | g \in G\} = \pi(\mathrm{lm}(I))$$

1. and 2. are equivalent.
If $\{\pi(g_1), \dots, \pi(g_q)\}$ is a Gröbner basis for $\pi(I)$ then the condition (3) holds. It remains to show that 3. implies 1..
Since $\{g_1, \dots, g_s\}$ is a Gröbner basis of I, we can write

$$\mathrm{lm}(I) = \langle \mathrm{lm}(g_1), \dots, \mathrm{lm}(g_s) \rangle.$$

We prove

$$\pi(\mathrm{lm}(I)) = \mathrm{lm}(\pi(I)).$$

Before we prove this equality, we prove the following claim.

Claim 1 Let $f \in I$ with $\pi(f) \neq 0$. Then, there exists $g \in I$ such that $\mathrm{lpp}_{\bar{A}}(g)$ divides $\mathrm{lpp}(\pi(f))$ and $\pi(\mathrm{lc}_{\bar{A}}(g)) \neq 0$.

Proof of Claim 1. This proof is the same as **"Kalkbrener [Kal97] Theorem 3.1"**.
We prove by induction on a monomial order $\succ$.
[Induction basis:] If $\mathrm{lpp}(f) = 1$, then $\pi(\mathrm{lc}(f)) \neq 0$ and $\mathrm{lpp}(f) = \mathrm{lpp}(\pi(f))$. Hence, the claim holds.
[Induction step:] Since the claim holds if $\pi(\mathrm{lc}(f)) \neq 0$, we assume that $\pi(\mathrm{lc}(f)) = 0$. If there exists an $i \in \{1, \ldots, r\}$ such that $\mathrm{lpp}(g_i)$ divides $\mathrm{lpp}(f)$, then we set

$$f' = \mathrm{lc}(g_i) \cdot f - \mathrm{lc}(f) \cdot \frac{\mathrm{lpp}(f)}{\mathrm{lpp}(g_i)} \cdot g_i.$$

Obviously, $\mathrm{lpp}(\pi(f')) = \mathrm{lpp}(\pi(f))$ and $\mathrm{lpp}(f) \succ \mathrm{lpp}(f')$, because of $\pi(\mathrm{lc}(f)) = 0$. Thus, the claim follows from the induction hypothesis.
Otherwise, there exist $j_1, \ldots, j_k \in \{q+1, \ldots, s\}$ and $c_{j_1}, \ldots, c_{j_k} \in R$ such that $\mathrm{lpp}(g_{j_l})$ divides $\mathrm{lpp}(f)$ for $l \in \{1, \ldots, k\}$ and

$$\mathrm{lm}(f) = \sum_{l=1}^{k} c_{j_l} \cdot \frac{\mathrm{lpp}(f)}{\mathrm{lpp}(g_{j_l})} \cdot \mathrm{lm}(g_{j_l}).$$

Let $i \in \{r+1, \ldots, s\}$. Since $\pi(g_i)$ is reducible to 0 module $\{\pi(g_1), \ldots, \pi(g_r)\}$ there exists an $h_i \in I$ and $b_i \in R \setminus \ker(\pi)$ with $\pi(b_i) \cdot \pi(g_i) = \pi(h_i)$ and $\mathrm{lpp}(f_i) \succ \mathrm{lpp}(\pi(g_i)) = \mathrm{lpp}(h_i)$. Set

$$f' := b \cdot f - \sum_{l=1}^{k} \frac{b}{b_{j_l}} \cdot c_{j_l} \cdot \frac{\mathrm{lpp}(f)}{\mathrm{lpp}(g_{j_l})} \cdot (b_{j_l} \cdot g_{j_l} - h_{j_l}),$$

where $b := \Pi_{l=1}^{k} b_{j_l}$. Obviously, $\mathrm{lpp}(\pi(f')) = \mathrm{lpp}(\pi(f))$ and $\mathrm{lpp}(f) \succ \mathrm{lpp}(f')$. The claim follows from the induction hypothesis. ☐

We prove $\pi(\mathrm{lm}(I)) = \mathrm{lm}(\pi(I))$.
($\subseteq$) Take an element $f \in \pi(\mathrm{lm}(I))$, then there exists $h \in I$ such that $f = \mathrm{lm}(\pi(h))$. By the claim 1, there exists $g \in I$ such that $\mathrm{lpp}(g)$ divides $\mathrm{lpp}(f)$ and $\pi(\mathrm{lc}(g)) \neq 0$. Hence,

$$\begin{aligned}
\mathrm{lpp}(f) &= d_1\,\mathrm{lpp}(g), && (d_1 \in K[\bar{X}]) \\
\mathrm{lm}(f) &= \mathrm{lc}(f) \cdot d_1 \cdot \mathrm{lpp}(g), && (\times\,\mathrm{lc}(f)) \\
\mathrm{lc}(g) \cdot \mathrm{lm}(f) &= \mathrm{lc}(f) \cdot d_1 \cdot \mathrm{lm}(g), && (\times\,\mathrm{lc}(g)) \\
\pi(\mathrm{lc}(g)) \cdot \pi(\mathrm{lm}(f)) &= \pi(\mathrm{lc}(f)) \cdot \pi(d_1) \cdot \pi(\mathrm{lm}(g)). && (\pi : \text{homo.})
\end{aligned}$$

Since $f = \mathrm{lm}(\pi(h))$, obviously $\pi(\mathrm{lm}(f)) = f$. Therefore, we have

$$f = \frac{\pi(\mathrm{lc}(f))}{\pi(\mathrm{lc}(g))} \cdot \pi(d_1) \cdot \mathrm{lm}(\pi(g)). \quad (\pi(\mathrm{lc}(g)) \neq 0)$$

Since $\mathrm{lm}(\pi(g)) \in \mathrm{lm}(\pi(I))$, $f \in \mathrm{lm}(\pi(I))$. Therefore, $\pi(\mathrm{lm}(I)) \subseteq \mathrm{lm}(\pi(I))$.

($\supseteq$) Take an element $f \in \mathrm{lm}(\pi(I))$. Then there exists $p \in I$ such that $f = \mathrm{lm}(\pi(p))$.By the claim 1, there exists $g \in I$ such that $\mathrm{lpp}(g)$ divides f and $\pi(\mathrm{lc}(g)) \neq 0$. Thus,

$$\begin{aligned}
\mathrm{lpp}(f) &= d\,\mathrm{lpp}(g), && (d \in K[\bar{X}]) \\
\mathrm{lm}(f) &= \mathrm{lc}(f) \cdot d \cdot \mathrm{lpp}(g), && (\times\,\mathrm{lc}(f)) \\
\mathrm{lc}(g) \cdot \mathrm{lm}(f) &= \mathrm{lc}(f) \cdot d \cdot \mathrm{lm}(g), && (\times\,\mathrm{lc}(g)) \\
\pi(\mathrm{lc}(g)) \cdot \pi(\mathrm{lm}(f)) &= \pi(\mathrm{lc}(f)) \cdot \pi(d) \cdot \pi(\mathrm{lm}(g)). && (\pi : \text{homo.}).
\end{aligned}$$

Since $f = \mathrm{lm}(\pi(p))$, obviously $\pi(\mathrm{lm}(f)) = f$. Therefore, we have

$$f = \frac{\pi(\mathrm{lc}(f))}{\pi(\mathrm{lc}(g))} \cdot \pi(d) \cdot \pi(\mathrm{lm}(g)).$$

Since $\pi(\mathrm{lm}(g)) \in \pi(\mathrm{lm}(I))$, $p_i \in \pi(\mathrm{lm}(I))$. Therefore, $\pi(\mathrm{lm}(I)) \supseteq \mathrm{lm}(\pi(I))$. □

4.4 Suzuki-Sato's algorithm

In this section we describe Suzuki-Sato's algorithm [SS06] for computing comprehensive Gröbner bases and comprehensive Gröbner systems. In general, the algorithms are faster than other existing algorithms. The algorithms are based on Kalkbrener's theory stability of ideals.
In this thesis, we assume the algorithm LCM. The algorithm $\mathsf{LCM}(h_1, \ldots, h_l)$ outputs the least common multiple of $h_1, \ldots, h_l$ in $K[\bar{A}]$ where $h_1, \ldots, h_l \in K[\bar{A}]$.

4.4.1 Comprehensive Gröbner systems

Here, we introduce the Suzuki-Sato algorithm [SS06] for computing comprehensive Gröbner systems.

The next two lemmas are the direct consequences of Theorem 4.3.2.

Lemma 4.4.1. Let F be a subset of $K[\bar{A}][\bar{X}]$. Let G be a Gröbner basis for $\langle F \rangle$ in $K[\bar{A}][\bar{X}]$ with respect to a term order $\succ$. Suppose that $\{h_1, \ldots, h_s\} := \{\mathrm{lc}_{\bar{A}}(g) : g \in G\}$ and $h := \mathsf{LCM}(h_1, \ldots, h_s)$.
Then, for any $\bar{a} \in L^m \backslash \mathbb{V}(h)$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ with respect to $\succ_1$ in $L[\bar{X}]$.

Proof. By Theorem 4.3.2 (3), this lemma holds. □

Lemma 4.4.2. Let F be a subset of $K[\bar{A}][\bar{X}]$ and S a subset of $K[\bar{A}]$. Let G be a Gröbner basis for $\langle F \cup S \rangle$ in $K[\bar{A}][\bar{X}]$ with respect to a term order $\succ$. Suppose that $B := \{b : b \in \langle S \rangle, b \in G\}$, $\{h_1, \ldots, h_s\} := \{\mathrm{lc}_{\bar{A}}(g) : g \in G \backslash B\}$ and $h := \mathsf{LCM}(h_1, \ldots, h_s)$.
Then, for any $\bar{a} \in \mathbb{V}(S) \backslash \mathbb{V}(h)$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ with respect to $\succ$ in $L[\bar{X}]$. Actually, we have $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G \backslash B)$.

Proof. If we take $g \in G \backslash B$, then for all $\bar{a} \in \mathbb{V}(S) \backslash \mathbb{V}(h)$ we have $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) \neq 0$. If we take $g \in G \cap B$, then we have $\sigma_{\bar{a}}(g) = 0$ and $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) = 0$. Of course, $\langle 0 \rangle$ is stable. Therefore, G is stable under the specialization $\sigma_{\bar{a}}$. By Theorem 4.3.2, $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G \backslash B)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$. □

By Lemma 4.4.1 and Lemma 4.4.2, we can construct an algorithm for computing comprehensive Gröbner systems which is from [SS06].

Algorithm 4.4.3. $\mathsf{Suzuki\text{-}Sato}(F, \succ)$ [SS06]

Input F: a finite subset of $K[\bar{A}][\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output G: a comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ$ on L^m.
begin
$G \leftarrow \mathsf{CGSMain}(F, \emptyset, \succ)$

return(G)
end

Algorithm 4.4.4. CGSMain($F, Z, \succ$)

Input F: a finite subset of $K[\bar{A}][\bar{X}]$,
Z: a finite set of polynomials in $K[\bar{A}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output H: a comprehensive Gröbner system for $\langle F \rangle$ on $\mathbb{V}(Z)$.
begin
$G \leftarrow$ GröbnerBasisB($F \cup Z, \succ$)
if $1 \in G$ **then**
$H \leftarrow \{(Z, \{1\}, \{1\})\}$
else
$G' \leftarrow G \backslash \{g : g \in G \cap K[\bar{A}], g \in \langle Z \rangle\}$
$S \leftarrow \{h_1, .., h_l\} := \{\mathrm{lc}_{\bar{A}}(f) : f \in G'\}$ $(**)$
if $S \neq \emptyset$ **then**
$h \leftarrow$ LCM($h_1, \ldots, h_l$)
$H \leftarrow \{(Z, \{h\}, G')\} \cup$ CGSMain($G, Z \cup \{h_1\}, \succ$) $\cup \cdots$
$\cdots \cup$ CGSMain($G, Z \cup \{h_l\}, \succ$)
else
$H \leftarrow \{(Z, \{1\}, G')\}$
end-if
end-if
return(H)
end

Remark : We are able to apply a lot of optimization techniques to obtain small and nice outputs comprehensive Gröbner systems. For instance in $(**)$, we can factorize all elements of S into irreducible factors, and compute a radical ideal of $\langle Z \rangle$ for getting small and nice outputs. Many researchers of parametric Gröbner bases have studied the optimization techniques to get small and nice outputs comprehensive Gröbner systems (and bases) [BW93, Kre92, Mon02, MM06, Nab05a, SSN03a, SS03, SSN03b, Wei02b, Wei03]. In the algorithms Suzuki-Sato and CGSMain, we can use these techniques for computing comprehensive Gröbner systems. Note that conditions of segments (of a comprehensive Gröbner system) produced by the algorithm `CGSM` may not be disjoint, i.e. $(\mathbb{V}(Z) \backslash \mathbb{V}(h)) \cap (\mathbb{V}(Z') \backslash \mathbb{V}(h'))$ could be non-empty for distinct elements $(Z, h, G), (Z', h', G') \in H$. Though this fact looks a serious disadvantage, it enables us to avoid producing unnecessary inequations (which is h in the algorithm CGSMain). In the algorithm we do not even check if $(\mathbb{V}(Z) \backslash \mathbb{V}(h)) = \emptyset$. Of course, we can check it after the algorithm terminates and omit it from the segments in case it is empty. We can also make the constructible sets of segments pairwise disjoints [SS06].
Since a leading coefficient of each polynomial of a segment does not vanish by the specialization, we can apply reductions of $K(\bar{A})[\bar{X}]$ where $K(\bar{A})$ is the field of rational functions. This is also one of optimization techniques.
In the algorithms Suzuki-Sato and CGSMain, we applied the algorithm GröbnerBasisB for computing Gröbner bases in $K[\bar{A}][\bar{X}]$. The Gröbner bases computed with respect to a block order with $\bar{X} \gg \bar{A}$ in $K[\bar{A}, \bar{X}]$ are not always reduced Gröbner bases in $K[\bar{A}][\bar{X}]$. As we saw in chapter 3, there sometimes exist some unnecessary polynomials in the outputs. Therefore, we can also apply the technique of reduced Gröbner bases in $K[\bar{A}][\bar{X}]$

for getting nice comprehensive Gröbner systems.

Theorem 4.4.5 ([SS06]). The algorithm NEW terminates for any input F of a finite subset of $K[\bar{A}][\bar{X}]$. The output forms a comprehensive Gröbner system for $\langle F\rangle$ on L^m.

Proof. Since we need the same proof in chapter 5,8 and 9, we introduce a proof of [SS06]. First, we show the termination. We suppose that CGSMain does not terminate, then there exists an infinite sequence $F_0, F_1, \ldots$ which are from the first line of CGSMain. Notice that $\langle F_{n+1}\rangle = \langle F_n \cup \{h_n\}\rangle$ for some $h_n \in K[\bar{A}]$ such that $h_n \notin \langle F_n\rangle$. Hence, we have $\langle F_n\rangle \subsetneq \langle F_n\rangle$ for each n, which contradicts to the fact $K[\bar{A},\bar{X}]$ is a Noetherian ring, and the algorithm GröbnerBasisB outputs a reduced Gröbner basis in $K[\bar{A},\bar{X}]$. This algorithm terminates.
We next show that, if $(Z,h,G) \in H$, then the triple (Z,h,G) forms a segment of a comprehensive Gröbner system for $\langle F\rangle$, i.e., $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle\sigma_{\bar{a}}(F)\rangle$ for each $\bar{a} \in \mathbb{V}(Z)\backslash\mathbb{V}(h)$.
Let G be a Gröbner basis of the ideal $\langle F'\rangle$ with respect to $\succ$ in $K[\bar{A}][\bar{X}]$, $B := \{g | g \in G \cap K[\bar{A}], g \in \langle Z\rangle\}$ and $\{h_1,\ldots,h_l\} := \{\mathrm{lc}_{\bar{A}}(f) | g \in G\backslash B\}$ and $h = \mathrm{lcm}(h_1,\ldots,h_l)$. Then by Lemma 4.4.2, $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle\sigma_{\bar{a}}(F')\rangle$ for $\bar{a} \in \mathbb{V}(Z)\backslash\mathbb{V}(h)$. Therefore, for $\bar{a} \in \mathbb{V}(Z)\backslash\mathbb{V}(h)$, $\sigma_{\bar{a}}(G\backslash B) = \sigma_{\bar{a}}(G)$ and $\sigma_{\bar{a}}(F') = \sigma_{\bar{a}}(F)$. This means that $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle\sigma_{\bar{a}}(F)\rangle$.
We have to finally prove that the conditions in H covers the entire L^m, i.e.,

$$L^m = \bigcup_{(P,q,G)\in H} \mathbb{V}(P)\backslash\mathbb{V}(q).$$

The following equation always holds.

$$\mathbb{V}(Z) = (\mathbb{V}(Z)\backslash\mathbb{V}(h')) \cup \bigcup_{i=1}^{l} \mathbb{V}(Z \cup h'_i).$$

The equation above follows by the induction on the well-founded tree of the algorithm. $\square$

The algorithm has been implemented in the computer algebra system Risa/Asir. In the following example, we see an output of the program.

Example 4.4.6. Let $F = \{ax^2y^2 + x^2 + 2, x^3 + bxy^2 + 2\}$ be a set of polynomials in $\mathbb{Q}[a,b][x,y]$, a,b parameters, x,y variables and $\succ$ the graded reverse lexicographic order such that $x \succ y$. Then, the program outputs the following as a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$.

```
[b]==0,  [a]!=0,
[x^3+2,x-a*y^2-1]

[b,a]==0,  [1]!=0,
[1]

[a]==0,  [b]!=0,
[x^2+2,x-b*y^2+2]

[0]==0,  [a*b]!=0,
[-x^2+(a*y^2+1)*x-b*y^2,a*x^3+b*x^2-b*x+b^2*y^2+2*a,-b*x^2-2*a*x+b^2*a*y^4
 +2*a^2*y^2+2*a-2*b]
```

This meaning is the following;

$$\begin{cases} \{x^3+2, x-ay^2-1\}, & \text{if } b=0, a\neq 0,\\ \{1\}, & \text{if } a=b=0,\\ \{x^2+2, x-by^2+2\}, & \text{if } a=0, b\neq 0,\\ \{-x^2+axy^2+x-by^2, ax^3+bx^2-bx+b^2y^2+2a, & \text{if } ab\neq 0.\\ \ -bx^2-2ax+b^2ay^4+2a^2y^2+2a-2b\} & \end{cases}$$

4.4.2 Comprehensive Gröbner bases

Here we introduce the Suzuki-Sato algorithm for computing comprehensive Gröbner bases. We already saw comprehensive Gröbner systems which have parameter spaces (conditions of parameters). However, comprehensive Gröbner bases do not have any parameter space. A comprehensive Gröbner basis is a set of polynomials. In order to construct an algorithm for computing comprehensive Gröbner bases, we need the concept "**faithful**" Definition 4.2.5.

Actually, we describe an algorithm for computing faithful comprehensive Gröbner systems. If $\{(\mathbb{V}(s_1)\backslash\mathbb{V}(t_1), G_1), \ldots, (\mathbb{V}(s_l)\backslash\mathbb{V}(t_l), G_l)\}$ is a faithful comprehensive Gröbner system for $\langle F\rangle$, then by the definition of comprehensive Gröbner basis, $G_1\cup\cdots\cup G_l$ is a comprehensive Gröbner basis for $\langle F\rangle$. Therefore, we modify the algorithm `CGSMain` to compute a faithful comprehensive Gröbner system. The key idea which is from [SS06], is to apply a new variable U.
In [SS06], they introduced a new auxiliary variable U besides $\bar{X}$ and $\bar{A}$ in order to compute comprehensive Gröbner bases.

We define homomorphisms σ^0 and σ^1 from $K[\bar{A}][U,\bar{X}]$ to $K[\bar{A}][\bar{X}]$ as a specialization of U with 0 and 1 respectively, i.e. $\sigma^0(f(U,\bar{A},\bar{X})) = f(0,\bar{A},\bar{X})$ and $\sigma^1(f(U,\bar{A},\bar{X})) = f(1,\bar{A},\bar{X})$.
Before we introduce the algorithm for computing comprehensive Gröbner bases, we need the following lemma which is also from [SS06].

Lemma 4.4.7 ([SS06]). Let F and S be subsets of $K[\bar{A}][\bar{X}]$. For any $g\in\langle (U\cdot F)\cup (U-1)\cdot S)\rangle\subseteq K[\bar{A}][U,\bar{X}]$, then $\sigma^0(g)\in\langle S\rangle\subseteq K[\bar{A}][\bar{X}]$ and $\sigma^1(g)\in\langle F\rangle\subseteq K[\bar{A}][\bar{X}]$.

Proof. See [SS06]. □

Theorem 4.4.8. Let F be a subset of $K[\bar{A}][\bar{X}]$, S a subset of $K[\bar{A}]$, and $\succ$ a term order on $\mathrm{pp}(\bar{X})$. Moreover, let G be a Gröbner basis of $\langle (U\cdot F)\cup(U-1)\cdot S)\rangle$ in $K[\bar{A}][U,\bar{X}]$ with respect to a block order $\succ' := (\succ_U,\succ)$ on $\mathrm{pp}(U,\bar{X})$ such that $U\gg\bar{X}$ where $\succ_U$ is a term order on $\mathrm{pp}(U)$. Suppose that $B_1 := \{g|g\in G\cap K[\bar{A}][U], \mathrm{lc}_{\bar{A}}(g)\in\langle S\rangle\}$, $B_2 := \{g|g\in G, \deg_U(\mathrm{lpp}_A(g)) = 0\}$, $G' := \{g|g\in G\backslash(B_1\cup B_2)\}$ and $\{h_1,\ldots,h_l\} := \{\mathrm{lc}_{\bar{A}}(g)|\,\mathrm{lc}_{\bar{A}}(g)\notin K, g\in G'\}\subseteq K[\bar{A}]$.
Then for each $\bar{a}\in\mathbb{V}(S)\backslash\mathbb{V}(h)$, $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis of $\langle\sigma_{\bar{a}}(F)\rangle$ in $L[\bar{X}]$ with respect to $\succ$ where $h=\mathrm{lcm}(h_1,\ldots,h_l)$. Actually, we have $\sigma_{\bar{a}}(\sigma^1(G)) = \sigma_{\bar{a}}(\sigma^1(G'))$.

Proof. Note that any polynomial of G' has a linear form of U, i.e., the degree of U is at most 1. It is clearly that $\sigma^1(G)$ is a basis of $\langle F\rangle$ by Lemma 4.4.7. We prove that $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis of $\langle\sigma_{\bar{a}}(F)\rangle$. For $\bar{a}\in\mathbb{V}(S)\backslash\mathbb{V}(h)$, we have $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g))\neq 0$ for each $g\in G'$.
By the sets G', B_1 and B_2, we have $G = G'\cup B_1\cup B_2$. For each $f\in B_1$, f can be written as $f = U\cdot f_1 + f_2$ where $f_1, f_2\in K[\bar{A}]$. By Lemma 4.4.7, $\sigma^0(f) = f_2\in\langle S\rangle$, thus $\sigma_{\bar{a}}(f_2) = 0$. By the definition of B_1, $\mathrm{lc}_{\bar{A}}(f) = f_1\in\langle S\rangle$, thus $\sigma_{\bar{a}}(f_1) = 0$. Hence,

$\sigma_{\bar{a}}(f) = 0$.
For each $q \in B_2$, by Lemma 4.4.7, $\sigma^0(q) = q \in \langle S \rangle$. Thus $\sigma_{\bar{a}}(q) = 0$. Even if we change a term order $\succ$ to a block order $\succ'$ with $U \gg \bar{X}$ in Lemma 4.4.2, the properties of Lemma 4.4.2 hold. Therefore, $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G \backslash (B_1 \cup B_2)) = \sigma_{\bar{a}}(G')$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(U \cdot F \cup (U-1) \cdot S) \rangle$ with respect to the $\succ'$ in $L[U, \bar{X}]$.
For $g \in G'$, g can be written as $g = U \cdot g_1 + g_2$ where $g_1, g_2 \in K[\bar{A}][\bar{X}]$. By Lemma 4.4.7, we have $\sigma^0(g) = g_2 \in \langle S \rangle$, thus $\sigma_{\bar{a}}(g_2) = 0$. Namely, we have $\sigma_{\bar{a}}(g) = \sigma_{\bar{a}}(U \cdot g_1)$. Since every power product of $\sigma_{\bar{a}}(G')$ has a variable U whose degree is 1 and $U \gg \bar{X}$, $\sigma^1(\sigma_{\bar{a}}(G'))$ is a Gröbner basis of $\langle \sigma^1(\sigma_{\bar{a}}(U \cdot F) \cup (U-1) \cdot S \rangle = \langle \sigma^1(\sigma_{\bar{a}}(U \cdot F)) \rangle = \langle \sigma_{\bar{a}}(F) \rangle$. Therefore, it follows that $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}]$. □

By Theorem 4.4.8, we can have an algorithm for computing faithful comprehensive Gröbner systems as follows.

Algorithm 4.4.9. $\mathsf{FCGS}(F, \succ)$ (Faithful Comprehensive Gröbner Systems) [SS06]

Input F: a subset of $K[\bar{A}][\bar{X}]$,
 $\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output G: a faithful comprehensive Gröbner system on L^m for $\langle F \rangle$ with respect to $\succ$.
begin
 $H \leftarrow \mathsf{GröbnerBasisB}(F)$
if $1 \in H$ **then**
 $G \leftarrow \{(\emptyset, \{1\}, H)\}$
end-if
 $S \leftarrow \{h_1, \ldots, h_l\} := \{q | q \in \mathsf{factorize}(\mathrm{lc}_{\bar{A}}(g)), \mathrm{lc}_{\bar{A}}(g) \notin K, g \in H\}$
if $S \neq \emptyset$ **then**
 $h \leftarrow \mathrm{lcm}(h_1, \ldots, h_l)$
 $G \leftarrow \{(\emptyset, h, H)\} \cup \mathsf{CGBMain}(H, \{h_1\}, \succ) \cup$
 $\cdots \cup \mathsf{CGBMain}(H, \{h_l\}, \succ)$
else
 $G \leftarrow \{(\emptyset, \{1\}, H)\}$
end-if
`return`(G)
end

Algorithm 4.4.10. $\mathsf{CGBMain}(F, Z, \succ)$

Input F: a subset of $K[\bar{A}][\bar{X}]$,
 Z: a finite set of polynomials in $K[\bar{A}]$,
 $\succ$: a term order on $\mathrm{pp}(U, \bar{X})$ such that $U \gg \bar{X}$,
Output G: a finite set of triples which forms a faithful comprehensive Gröbner system on $\mathbb{V}(Z)$ for $\langle F \rangle$.
begin
 $H \leftarrow \mathsf{GröbnerBasisB}(U \cdot F \cup ((U-1) \cdot Z, \succ))$
 $C \leftarrow$ the reduced Gröbner basis for $\langle Z \rangle$ in $K[\bar{A}]$
if $1 \in C$ **then**
 $G \leftarrow \emptyset$
end-if
 $B_1 \leftarrow \{g | g \in H \cap K[\bar{A}][U], \mathrm{lc}_{\bar{A}}(g) \in \langle Z \rangle\}$
 $B_2 \leftarrow \{g | g \in H, \deg_U(\mathrm{lpp}_A(g)) = 0\}$
 $H' \leftarrow H \backslash (B_1 \cup B_2)$

$L \leftarrow \{h_1, \ldots, h_l\} := \{q | q \in \mathsf{factorize}(\mathrm{lc}_{\bar{A}}(g)) \backslash K,\ g \in H'\}$
if $L \neq \emptyset$ **then**
$h \leftarrow \mathrm{lcm}(h_1, \ldots, h_l)$
$G \leftarrow \big\{(Z, h, \sigma^1(H'))\big\} \cup \mathsf{CGSMain}(F, Z \cup \{h_1\}, \succ) \cup \cdots \cup \mathsf{CGSMain}(F, Z \cup \{h_l\}, \succ)$
else
$G \leftarrow \big\{(Z, \{1\}, \sigma^1(H')\big\}$
end-if
`return`(G)
end

Remark : Like the remark of Algorithm 4.4.3, we are able to apply a lot of optimization techniques for getting small and nice outputs. See the remark.

Theorem 4.4.11 ([SS06]). Let F be a finite set of polynomials in $K[\bar{A}][\bar{X}]$. Then, the algorithm $\mathsf{FCGSM}(F, \succ)$ terminates. The output of FCGS is a faithful comprehensive Gröbner system on L^m for $\langle F \rangle$.

Proof. In order to show the termination of the algorithm, it suffices to show that each h_i is not in the ideal $\langle Z \rangle$ for each $i = 1, \ldots, l$ because this algorithm is almost same as algorithm CGS (see Theorem 4.4.3) (and we have $\sigma_{\bar{a}}(B_1) = \sigma_{\bar{a}}(B_2) = 0$ where $\bar{a} \in \mathbb{V}(Z) \backslash \mathbb{V}(h)$). All notations of this proof is from the algorithm FCGS.
By the construction of h_j, there exists $g \in H$ (which is from $\mathsf{GröbnerBasisB}(UF \cup (U-1)Z, \succ))$) such that $h_i \in \mathsf{factorize}\,\mathrm{lc}_{\bar{A}}(g)$, $\mathrm{lpp}_{\bar{A}}(g) \notin \mathrm{pp}(\bar{X})$. Therefore g can be written as

$$g = h_i p U T + g_1,$$

where $T \in \mathrm{pp}(\bar{X})$, $p \in \mathrm{pp}(\bar{A})$, $\mathrm{lpp}_{\bar{A}}(g) = U \cdot T$ and $g_1 \in K[\bar{A}][U, \bar{X}]$. If $h_i \in \langle Z \rangle$, then $h_i \cdot (U-1) \in \langle G \rangle$. Hence, $\mathrm{lm}_{\bar{A}}(h_i \cdot (U-1)) = \mathrm{lm}_{\bar{A}}(h_i \cdot U)$ must be reduced by G. In the algorithm GröbnerBasisB, we compute the reduced Gröbner basis for $\langle U \cdot F \cup (U-1) \cdot Z \rangle$ in $K[\bar{A}, U, \bar{X}]$ with respect to a block order with $U \gg \bar{X} \gg \bar{A}$. Since G is the reduced Gröbner basis in $K[\bar{A}, U, \bar{X}]$, this is the contradiction. Therefore, h_i is not in the ideal $\langle Z \rangle$.
It is an easy consequence of Theorem 4.4.8 and Lemma 4.4.7 that the output of FCGSM is a faithful comprehensive Gröbner system on L^m for $\langle F \rangle$. □

Now, it is clear that the following algorithm outputs a comprehensive Gröbner bases for $\langle F \rangle$.

Algorithm 4.4.12. $\mathsf{CGB}(F, \succ)$ [SS06]

Input F: a subset of $K[\bar{A}][\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output G: a comprehensive Gröbner basis on L^m for $\langle F \rangle$ with respect to $\succ$.
begin
$G \leftarrow \emptyset$
$H \leftarrow \mathsf{FCGS}(F, \succ)$
while $H \neq \emptyset$ **do**
Select a triple (D, Z_1, Z_2) from H
$H \leftarrow H \backslash \{(D, Z_1, Z_2))\}$
$G \leftarrow G \cup \{D\}$
end-while

return(D)
end

The algorithms FCGS and CGB have been implemented in the computer algebra system Risa/Asir by the author.

Example 4.4.13. Let $F = \{ax^2y^2 + x^2 + 2, x^3 + bxy^2 + 2\}$ be a set of polynomials in $\mathbb{Q}[a, b][x, y]$, a, b parameters, x, y variables and $\succ$ the graded reverse lexicographic order such that $x \succ y$. We saw a comprehensive Gröbner system for $\langle F\rangle$ in Example 4.4.6. Now, we consider a faithful comprehensive Gröbner system and comprehensive Gröbner basis. First, we consider a faithful comprehensive Gröbner system. Our program outputs the following as a faithful comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$.

```
[b]==0,  [a]!=0,
[-b*y^2*x^2+2*x-b^2*y^4-2*a*y^2-2,x^3+b*y^2*x+2]

[b,a]==0,  [1]!=0,
[(-2*a-b)*y^2*x^2-2*a*y^2*x-b^2*y^4+(-2*a+2*b)*y^2-6]

[a]==0,  [b]!=0,
[a*y^2*x^2+(a*y^2+1)*x-b*y^2+2,(a*y^2+1)*x^2+2]

[0]==0,  [a,b]!=0,
[x^3+b*y^2*x+2,-b*x^2-2*a*x+b^2*a*y^4+2*a^2*y^2+2*a-2*b,-b*y^2*x^2+2*x-b^2
*y^4-2*a*y^2-2,-x^2+(a*y^2+1)*x-b*y^2]
```

This meaning is the following;

$$\begin{cases} \{-by^2x^2 + 2x - b^2y^4 - 2ay^2 - 2, x^3 + by^2x + 2\}, & \text{if } b = 0, a \neq 0, \\ \{(-2a - b)y^2x^2 - 2ay^2x - b^2y^4 + (-2a + 2b)y^2 - 6\}, & \text{if } a = b = 0, \\ \{ay^2x^2 + (ay^2 + 1)x - by^2 + 2, (ay^2 + 1)x^2 + 2\}, & \text{if } a = 0, b \neq 0, \\ \{x^3 + by^2x + 2, -bx^2 - 2ax + b^2ay^4 + 2a^2y^2 + 2a - 2b & \text{if } ab \neq 0. \\ \ -by^2x^2 + 2x - b^2y^4 - 2ay^2 - 2, -x^2 + (ay^2 + 1)x - by^2\} & \end{cases}$$

If one compares the output with Example 4.4.6, then one can easily understand that this output has a lot of unnecessary monomials for a comprehensive Gröbner system. However, when we compute a comprehensive Gröbner basis, we need these monomials.

Next, we consider a comprehensive Gröbner basis for $\langle F\rangle$ with respect to $\succ$. The program outputs the following as a comprehensive Gröbner basis G:

```
[-b*x^2-2*a*x+b^2*a*y^4+2*a^2*y^2+2*a-2*b, -x^2+(a*y^2+1)*x-b*y^2,(a*y^2+1
)*x^2+2,a*y^2*x^2+(a*y^2+1)*x-b*y^2+2, (-2*a-b)*y^2*x^2-2*a*y^2*x-b^2*y^4+
(-2*a+2*b)*y^2-6, -b*y^2*x^2+2*x-b^2*y^4-2*a*y^2-2, x^3+b*y^2*x+2].
```

Note that if we substitute any values for parameters a, b of G then the set G computed is always a Gröbner basis for $\langle\sigma(F)\rangle$ with respect to $\succ$. However, the set G is **not always the reduced Gröbner basis** for $\langle\sigma(F)\rangle$ with respect to $\succ$.

4.5 Software

Here we introduce the software for computing comprehensive Gröbner bases and comprehensive Gröbner systems.

4.5.1 REDUCE package REDLOG

REDLOG is a package of the computer algebra system REDUCE[*1]. The Weispfenning algorithm [Wei92] for computing comprehensive Gröbner bases have been implemented within the framework of the REDUCE package REDLOG [DSS06]. REDLOG has the commands for computing comprehensive Gröbner bases and comprehensive Gröbner systems. In the following websites, one can see the manual and some examples.

http://www.fmi.uni-passau.de/~redlog/
http://www.fmi.uni-passau.de/~reduce/cgb/

The command gsys computes a faithful comprehensive Gröbner system. The command cgb computes a comprehensive Gröbner basis. In the following example, we give examples how the commands works. Let $\{ax^2y + bx + y, bx + y^2\}$ be a subset of $\mathbb{Q}[a,b][x,y]$, x, y variables and a, b parameters. We have the lexicographic order $\succ$ such that $x \succ y$. Then, the program works as follows.

```
REDUCE 3.8, 15-Apr-2004, patched to 22-Feb-2006 ...

1: load(cgb);

2: torder({x,y},lex)$

3: gsys {a*x^2*y+b*x*y,b*x+y^2};

{{a <> 0 and b <> 0,
      2
  {a*x *y + b*x*y,
          2
   b*x + y ,
      5    2  3
   a*y  - b *y }},
 {a <> 0 and b = 0,
      2                 2
  {a*x *y + b*x*y,b*x + y }},

 {b <> 0 and a = 0,
      2
  {a*x *y + b*x*y,
      2      3
   a*x *y - y ,
          2
   b*x + y }},

                         2
 {a = 0 and b = 0,{b*x + y }}}

4: cgb{a*x^2*y+b*x*y,b*x+y^2};
```

*1 http://www.reduce-algebra.com/index.htm

```
    2
{a*x *y + b*x*y,
    2      3
 a*x *y - y ,
        2
 b*x + y ,
    5    2  3
 a*y  - b *y }
```

4.5.2 Montes' programs in Maple

The Montes algorithm [Mon02, MM06] has been implemented in the computer algebra system Maple[*2] by Montes. One can get the program from the following website.

http://www-ma2.upc.edu/~montes/

In the following example, we give examples how the program works. Let $\{ax^2y + bx + y, bx + y^2\}$ be a subset of $\mathbb{Q}[a, b][x, y]$, x, y variables and a, b parameters. We have the lexicographic order $\succ$ such that $x \succ y$ (and $a > b$). Then, the program works as follows.

```
> with(dpgb):
> S:=[a*x^2*y+b*x*y,b*x+y^2];
```

$$S := [ax^2y + bxy, bx + y^2]$$

```
Xord:=plex(x,y); Pord:=plex(a,b);
```

$$\begin{aligned} Xord :=&\ \text{plex}(x, y) \\ Pord :=&\ \text{plex}(a, b) \end{aligned}$$

```
> T0:=dispgb(S,Xord,Pord,rebuild=0):
```

The output of the command dispgb has a lot of information. There exist two routines to obtain the fundamental information: "tplot" (to have a visual description of the discussion tree) and "finalcases" (to obtain the algebraic content of the tree.

```
> finalcases(T0);
```

$$[[[1,\ 1],\ [a\,y^5 - b^2\,y^3,\ b\,x + y^2],\ [],\ \{a,\ b\},\ [y^5,\ x]],\ [[1,\ 0],\ [y^2,\ y\,x^2],\ [b],\ \{a\},\ [y^2,\ y\,x^2]], \\ [[0,\ 1],\ [y^3,\ b\,x + y^2],\ [a],\ \{b\},\ [y^3,\ x]],\ [[0,\ 0],\ [y^2],\ [b,\ a],\ \{\},\ [y^2]]]$$

The output of "finalcases" is a list of terminal cases. Each case is given by a list whose elements are
1) the label,
2) the reduced Gröbner basis,
3) the null condition,
4) the non-null condition,
5) the set of the leading power products of the Gröbner basis.

In this thesis, we do not explain in the detail. If one is interested in the program, download it from the website and see the tutorial of the program DPGB.

[*2] http://www.maplesoft.com/

4.5.3 Suzuki's programs in the several computer algebra systems

The Suzuki-Sato algorithm [SS06] has been implemented in the computer algebra systems Risa/Asir*3, Mathematica*4, Maple and singular*5 by Suzuki. One can download the programs from the following website.

http://kurt.scitec.kobe-u.ac.jp/~sakira/CGBusingGB/

We give an example of the Maple's program. Let $\{ax^2y + bx + y, y^2 + x, bx + y^2\}$ be a subset of $\mathbb{Q}[a, b][x, y]$, x, y variables and a, b parameters. We have the lexicographic order plex such that $x \succ y$. Then, the program works to compute a comprehensive Gröbner system as follows.

```
> G := cgs([a*x^2*y+b*x+y,y^2+x,b*x+y^2],[x,y],[a,b],plex(x,y),plex(x,y,a,
b));
```

$$G := \big\{[1, [b - 1, a, -y + y^2, x + y]], [a, [b - 1, ay^5 - y^2 + y, y^2 + x]],$$
$$[a\,(b - 1)\,, [by - y, ay^5 - y^2 + y, y^2 + x]], [b - 1, [a, by - y, -y + y^2, x + y]]\big\}$$

```
> bases2exp(G);
```

$$\big\{[\{b - 1 = 0, a = 0\}\,, \{-y + y^2, x + y\}],$$
$$[\{b - 1 = 0, a! = 0\}\,, \{ay^5 - y^2 + y, y^2 + x\}],$$
$$[\{(a\,(b - 1))! = 0\}\,, \{ay^5 - y^2 + y, y^2 + x, by - y\}],$$
$$[\{a = 0, (b - 1)! = 0\}\,, \{-y + y^2, x + y, by - y\}]\big\}$$

4.5.4 Nabeshima's package

The author made a package in the compute algebra system Risa/Asir for computing comprehensive Gröbner bases and comprehensive Gröbner systems. In chapter 10, we introduce the package PGB.

*3 http://www.math.kobe-u.ac.jp/Asir/asir.html
*4 http://www.wolfram.com/
*5 http://www.singular.uni-kl.de/

Chapter 5

A new algorithm for computing comprehensive Gröbner systems

In this chapter, we treat a new algorithm for computing comprehensive Gröbner systems. In the previous chapter, we saw the Suzuki-Sato algorithm [SS06] for computing comprehensive Gröbner systems. Actually, in time complexity, the Suzuki-Sato algorithm is more efficient than other existing algorithms. In this chapter, we improve the Suzuki-Sato algorithm by using inequations ("$\neq 0$") and Gröbner bases computation in a polynomial ring over a polynomial ring.

If we compute a Gröbner basis for an ideal in a polynomial ring over a polynomial ring, then the Gröbner basis computed often has the special property $\diamondsuit 1$ (see section 5.1) which does not hold in a polynomial ring over a field. This property makes overmuch parameter spaces. Hence, the computation of comprehensive Gröbner systems becomes expensive. However, by using inequations ("$\neq 0$"), we can avoid this behavior. Therefore, we can compute a comprehensive Gröbner system much faster, and have a nice comprehensive Gröbner system. We implemented our new algorithm in the computer algebra system Risa/Asir [NT92]. Through our computation experiment, we checked that in many cases, our program is faster than the Suzuki-Sato algorithm. Especially, if the number of parameters is greater than the number of variables, our algorithm is much more efficient than Suzuki-Sato's one. Moreover, the outputs of our program are much nicer than the Suzuki-Sato algorithm. That is, the number of partitions of the whole parameter space is smaller than Suzuki-Sato's. This chapter is based on the author's paper [Nab07d].

First, we motivate the new algorithm in order to facilitate the understanding the algorithm.

5.1 Motivation

Let F be a subset of $K[\bar{A}][\bar{X}]$. Then, by the algorithm GröbnerBasisB we can compute a Gröbner basis $G = \{g_1, \ldots, g_l\}$ for $\langle F \rangle$ with respect to $\succ$ in $K[\bar{A}][\bar{X}]$. The Gröbner basis G in $K[\bar{A}][\bar{X}]$ often (not always) has the following property

($\diamondsuit 1$)

$$g_i, g_j \in G \text{ such that } \operatorname{lpp}_{\bar{A}}(g_i) \mid \operatorname{lpp}_{\bar{A}}(g_j) \text{ and } g_i \neq g_j.$$

If we consider a reduced Gröbner basis for a given ideal in $K[\bar{X}]$ (a polynomial ring over a field), then the reduced Gröbner basis does not hold this property. Actually, when we compute a comprehensive Gröbner system for a given ideal, this property often makes many small and unnecessary parameter spaces. Hence, our strategy for computing comprehensive Gröbner systems is **"avoiding this property by using inequations ($\neq 0$)"**. Before describing our algorithm, we consider the Suzuki-Sato algorithm and give our idea.

The first step of Suzuki-Sato$(F, \succ)$ works as Figure 5.1. That is, we have to consider l cases $\mathrm{lc}_{\bar{A}}(g_1) = 0, \ldots, \mathrm{lc}_{\bar{A}}(g_l) = 0$ for computing a comprehensive Gröbner system for $\langle F \rangle$.

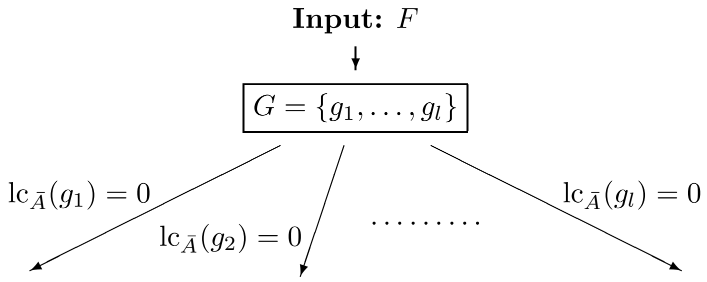

Figure 5.1

The Suzuki-Sato algorithm does not apply inequations ("$\neq 0$") for computing comprehensive Gröbner systems. In this point, this algorithm is extremely simple. However, as we said the above, Gröbner bases in $K[\bar{A}][\bar{X}]$ have the special property ($\Diamond 1$). Hence, the Suzuki-Sato algorithm provides overmuch parameter spaces. This condition (a lot of parameter spaces) is not nice when we compute a comprehensive Gröbner systems. Probably, by using inequations ("$\neq 0$"), we can obtain the number of parameter spaces which are smaller than Suzuki-Sato's outputs. This means that we may compute a comprehensive Gröbner systems more efficient than the Suzuki-Sato algorithm. (In the next subsection, we will discuss about this theory.) **When and how do we use inequations?**

If there exists $g_i \in G$ such that $\mathrm{lpp}_{\bar{A}}(g_i) = 1$, then we do not need to consider l cases. We need to consider only one case (branch) $\mathrm{lc}_{\bar{A}}(g_i) = 0$, because for $\bar{a} \in L^m \backslash \mathbb{V}(\mathrm{lc}_{\bar{A}}(g_i))$, the Gröbner basis of $\sigma_{\bar{a}}(F)$ is $\{1\}$. That is, if $\mathrm{lc}_{\bar{A}}(g_i) \neq 0$, then we can decide one segment without computing the cases $\mathrm{lc}_{\bar{A}}(g_j) = 0$ for $1 \leq j \neq i \leq l$. Therefore, we can remove the cases left by the inequations $\mathrm{lc}_{\bar{A}}(g_i) \neq 0$.

Suppose that for $p \in G$, $G_p := \{g \in G \backslash \{p\} : \mathrm{lpp}_{\bar{A}}(p) | \mathrm{lpp}_{\bar{A}}(g)\}$. If G_p is not an empty-set, for $\bar{a} \in L^m \backslash \mathbb{V}(\mathrm{lc}_{\bar{A}}(p))$, all elements of $\mathrm{lpp}_{\bar{A}}(G_p)$ can be reduced by $\sigma_{\bar{a}}(\mathrm{lpp}_{\bar{A}}(p))$. Therefore, we do not need to consider the cases $\mathrm{lc}_{\bar{A}}(g_{pi}) = 0$ where $g_{pi} \in G_p$. That is, the set $\mathrm{lc}_{\bar{A}}(G_p)$ can be removed by $\mathrm{lc}_{\bar{A}}(p) \neq 0$. If so, we can construct a new algorithm for computing comprehensive Gröbner systems which is more efficient than the Suzuki-Sato one. If G_p is an empty-set, then we can follow the Suzuki-Sato algorithm. This is our main strategy for computing comprehensive Gröbner systems.

Now we have a specific example for computing a comprehensive Gröbner system. Let $F = \{ax^3, bx^2, cx\} \in \mathbb{Q}[a, b, c][x]$ where a, b, c are parameters and x is a variable. First, we consider the Suzuki-Sato algorithm. The Suzuki-Sato algorithm works as Figure 5.2. A comprehensive Gröbner basis for $\langle F \rangle$ is $\left\{\boxed{1}, \boxed{2}, \ldots, \boxed{16}\right\}$ in Figure 5.2. Of course, we can use several optimization techniques for getting small and nice comprehensive Gröbner systems. However, basically, the Suzuki-Sato algorithm works as Figure 5.2.

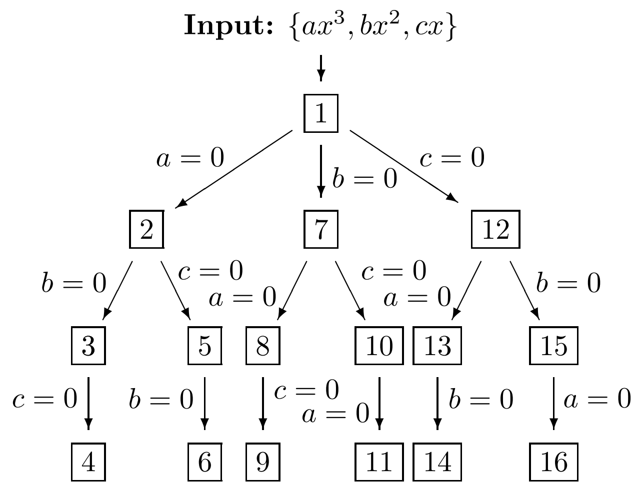

Figure 5.2

The algorithm (with several techniques) has been implemented in the computer algebra system Risa/Asir. The program outputs the following as a comprehensive Gröbner systems for $\langle F \rangle$.

$$\begin{cases} \{x\}, & \text{if } a=0, cb \neq 0, \\ \{x\}, & \text{if } a=b=0, c \neq 0 \\ \{x\} & \text{if } b=0, ac \neq 0 \\ \{x^2\} & \text{if } a=c=0, b \neq 0 \\ \{\emptyset\} & \text{if } a=b=c=0, \\ \{x^3\} & \text{if } c=b=0, a \neq 0, \\ \{x^2\}, & \text{if } c=0, ab \neq 0 \\ \{x\} & \text{if } abc \neq 0. \end{cases}$$

The program outputs 8 segments.

Next we try to apply our idea which uses inequations " $\neq 0$". First, we compute a Gröbner basis $S1$ for $\langle F \rangle$ in $\mathbb{Q}[a,b,c][x]$. Then, $S1 = \{ax^3, bx^2, cx\}$, and we know that $\mathrm{lpp}_{\{a,b,c\}}(cx) = x$, $\mathrm{lpp}_{\{a,b,c\}}(bx^2) = x^2$, $\mathrm{lpp}_{\{a,b,c\}}(ax^3) = x^3$. Clearly, $x|x^2$ and $x|x^3$. Hence, if $\mathrm{lc}_{\{a,b,c\}}(cx) = c \neq 0$, then a Gröbner basis for $\langle F \rangle$ is $\{x\}$. Since $L^3 \backslash \mathbb{V}(c)$ $(c \neq 0)$ cannot cover the whole space L^3, next we have to consider the case $c = 0$. If $c = 0$, then we have $S2 = \{ax^3, bx^2\}$ which is a Gröbner basis for $\langle S2 \rangle$ (itself). Clearly, $\mathrm{lpp}_{\{a,b,c\}}(bx^2) | \mathrm{lpp}_{\{a,b,c\}}(ax^3)$. Hence, if $c = 0$ and $b \neq 0$, then a Gröbner basis for $\langle F \rangle$ is $\{x^2\}$. Finally, we have to consider the cases $a \neq 0$ and $a = 0$. Therefore, our idea works as Figure 5.3.

Our idea returns the following comprehensive Gröbner systems for $\langle F \rangle$.

$$\begin{cases} \boxed{1} = [\{x\}, & \text{if } c \neq 0], \\ \boxed{2} = [\{x^2\}, & \text{if } c=0, b \neq 0], \\ \boxed{3} = [\{x^3\} & \text{if } b=c=0, a \neq 0], \\ \boxed{4} = [\{\emptyset\} & \text{if } a=b=c=0]. \end{cases}$$

This comprehensive Gröbner system has 4 segments.

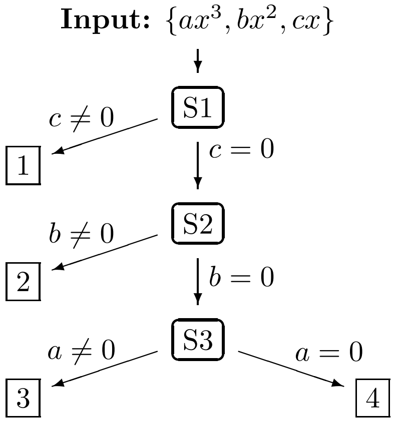

Figure 5.3

As we saw, our computation process Figure 5.3 is simpler than Suzuki-Sato's one Figure 5.2. Furthermore, the output of our approach has only 4 segments which is smaller than Suzuki-Sato's one. Therefore, our approach is much more efficient than the Suzuki-Sato algorithm.

In the next subsection, we will describe our approach strictly, and give a new algorithm for computing comprehensive Gröbner systems.

5.2 A new algorithm

In this section, we give a new algorithm for computing comprehensive Gröbner systems. The following theorem is the main idea for constructing the new algorithm.

Theorem 5.2.1. Let F be a subset of $K[\bar{A}][\bar{X}]$, $H = \{g, g_1, \dots, g_l\}$ a Gröbner basis for $\langle F\rangle$ with respect to $\succ$. Select g from H, and set $r := \frac{1}{\mathrm{lc}_{\bar{A}}(g)}$ (r is a new variable) and $g' := \mathrm{lpp}_{\bar{A}}(g) + r \cdot (g - \mathrm{lm}_{\bar{A}}(g))$. Suppose that $H' := (H \backslash \{g\}) \cup \{g'\} = \{g', g_1, \dots, g_l\} \subseteq K[r, \bar{A}][\bar{X}]$, and G' is a Gröbner basis of H' with respect to $\succ$ in $K[r, \bar{A}][\bar{X}]$. Furthermore, $G := \left\{ f \neq 0, f \in K[\bar{A}][\bar{X}] \,\middle|\, f = \mathrm{lc}_{\bar{A}}(g)^k \cdot \sigma_{r=\frac{1}{\mathrm{lc}_{\bar{A}}(g)}}(q),\ \deg_r(q) = k \in \mathbb{N}, q \in G' \right\}$ and $\{h_{01}, \dots, h_{0e}\} := \{\mathrm{lc}_{\bar{A}}(f) \in K[\bar{A}] : f \in G\}$.
Then, for any $\bar{a} \in L^m \backslash (\mathbb{V}(\mathrm{lc}_{\bar{A}}(g)) \cup \mathbb{V}(h))$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F)\rangle$ with respect to $\succ$ in $L[\bar{X}]$ where $h = \mathsf{LCM}(h_{01}, \cdots, h_{0e})$. ($\sigma_{r=\frac{1}{\mathrm{lc}_{\bar{A}}(g)}}(q)$ means substituting $\frac{1}{\mathrm{lc}_{\bar{A}}(g)}$ for the variable r of q.)

Proof. For all $\bar{a} = (a_1, \dots, a_m) \in L^m \backslash (\mathbb{V}(\mathrm{lc}_{\bar{A}}(g)) \cup \mathbb{V}(h))$, we have $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) \neq 0$, and $\sigma_{\bar{a}}(h) \neq 0$. Set $\bar{b} := (a_1, \dots, a_n, \frac{1}{\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g))}) \in L^{m+1}$. By the definition of h, for all $p \in G'$, we have $\sigma_{\bar{b}}(\mathrm{lm}_{\bar{A}}(p)) \neq 0$. Hence, by Theorem 4.4.1, $\sigma_{\bar{b}}(G')$ is a Gröbner basis for $\langle \sigma_{\bar{b}}(H')\rangle$ with respect to $\succ$. Actually, $\langle \sigma_{\bar{b}}(G')\rangle = \langle \sigma_{\bar{a}}(G)\rangle$. Therefore, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{b}}(H')\rangle$ with respect to $\succ$, too. Since $\sigma_{\bar{a}}(g_i) = \sigma_{\bar{b}}(g_i)$ and $\langle \sigma_{\bar{a}}(g)\rangle = \langle \sigma_{\bar{b}}(g')\rangle$ for $1 \leq i \leq l$, we have $\langle \sigma_{\bar{b}}(H')\rangle = \langle \sigma_{\bar{a}}(H)\rangle$. As $\sigma_{\bar{a}}$ is a ring homomorphism, obviously we have $\langle \sigma_{\bar{a}}(H)\rangle = \langle \sigma_{\bar{a}}(F)\rangle$. Therefore, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F)\rangle$ with respect to $\succ$. ∎

The following two corollaries are the direct consequence of Theorem 4.3.2, Theorem 5.2.1 and Lemma 4.4.1.

Corollary 5.2.2. With the same notations and conditions in Theorem 5.2.1, select s from G, and set $r := \frac{1}{\mathrm{lc}_{\bar{A}}(s)}$ (r is a new variable) and $s' := \mathrm{lpp}_{\bar{A}}(s) + r \cdot (s - \mathrm{lm}_{\bar{A}}(s))$. Suppose that $D_1 := (G \backslash \{s\}) \cup \{s'\} \subseteq K[r, \bar{A}][\bar{X}]$, and D_2 is a Gröbner basis of $\langle D_1 \rangle$ with respect to $\succ$ in $K[r, \bar{A}][\bar{X}]$. Furthermore,
$D_3 := \left\{ f \in K[\bar{A}][\bar{X}] \,\middle|\, f \neq 0, f = \mathrm{lc}_{\bar{A}}(s)^k \cdot \sigma_{r=\frac{1}{\mathrm{lc}_{\bar{A}}(g)}}(q),\ \deg_r(q) = k \in \mathbb{N}, q \in D_2 \right\}$ and $\{h_{11}, \ldots, h_{1e}\} := \{\mathrm{lc}_{\bar{A}}(f) \in K[\bar{A}] : f \in D_3\}$.
Then, for any $\bar{a} \in L^m \backslash (\mathbb{V}(\mathrm{lc}_{\bar{A}}(g)) \cup \mathbb{V}(\mathrm{lc}_{\bar{A}}(s)) \cup \mathbb{V}(h_1))$, $\sigma_{\bar{a}}(D_3)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ with respect to $\succ$ in $L[\bar{X}]$ where $h_1 = \mathsf{LCM}(h_{11}, \cdots, h_{1e})$.

Corollary 5.2.3. With the same notations and conditions in Theorem 5.2.1, let S be a subset of $K[\bar{A}]$. Then, we compute a Gröbner basis D for $\langle G \cup S \rangle$ with respect to $\succ$ by the algorithm GröbnerBasisB in $K[\bar{A}][\bar{X}]$. Suppose that $B := \{b : b \in D, b \in \langle S \rangle\}$, $\{d_1, \ldots, d_u\} := \{\mathrm{lc}_{\bar{A}}(f) : f \in D \backslash B\}$ and $d := \mathsf{LCM}(d_1, \ldots, d_u)$.
Then, for any $\bar{a} \in \mathbb{V}(S) \backslash (\mathbb{V}(\mathrm{lc}_{\bar{A}}(g)) \cup \mathbb{V}(d))$, $\sigma_{\bar{a}}(D)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ with respect to $\succ$ in $L[\bar{X}]$.

Now, we can construct a new algorithm by using Lemma 4.4.1, Lemma 4.4.2, Theorem 5.2.1, Corollary 5.2.2 and Corollary 5.2.3. Before describing the algorithm, we give one example for computing a comprehensive Gröbner system by using our strategy.

Example 5.2.4. Let $F = \{xy + x, ax^2 + y + 2, bxy + y\}$ be a subset of $\mathbb{Q}[a, b][x, y]$, a, b parameters and x, y variables. We have the lexicographic order $\succ$ such that $x \succ y$. Let's compute a comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ$.

1. First, we compute a Gröbner basis for $\langle F \rangle$ in $\mathbb{Q}[a, b][x, y]$ by the algorithm GröbnerBasisB. Then, the algorithm GröbnerBasisB$(F, \succ)$ outputs $\{a + b^2, y + 1, bx + 1, ax - b\}$. Clearly, for $\alpha \in \mathbb{C}^2 \setminus \mathbb{V}(a + b^2)$, $\{1\}$ is the Gröbner basis for $\langle \sigma_\alpha(F) \rangle$ with respect to $\succ$. That is, one of segments of a comprehensive Gröbner system for $\langle F \rangle$ is $(\mathbb{C}^2 \setminus \mathbb{V}(a + b^2), \{1\})$.

2. Next, we have to consider the case $\{a + b^2 = 0\}$. By Lemma 4.4.2, we can obtain one segment $(\mathbb{V}(a + b^2) \setminus \mathbb{V}(ab), \{y + 1, bx + 1, ax - b\})$. However, this procedure is the same as Suzuki-Sato's one. As we are considering a new algorithm by using Theorem 5.2.1, we do not apply this procedure. Since we use Theorem 5.2.1, we have to select one polynomial from $\{y + 1, bx + 1, ax - b\}$. Now we have a question "**Which polynomial had we better select to compute a comprehensive Gröbner system efficiently?**"
This answer is very important for our new algorithm. In this example, we know that $\mathrm{lpp}_{\{a,b\}}(bx+1)$ divides $\mathrm{lpp}_{\{a,b\}}(ax-b)$, and $\mathrm{lpp}_{\{a,b\}}(ax-b)$ divides $\mathrm{lpp}_{\{a,b\}}(bx+1)$. Hence, if we select $bx + 1$ (or $ax - b$), then $ax - b$ (or $bx + 1$) can be reduced by $\mathrm{lpp}_{\{a,b\}}(bx + 1)$ (or $\mathrm{lpp}_{\{a,b\}}(ax - b)$). Therefore, we had better select one of $bx + 1, ax - b$. Let's select $ax - b$. In order to follow Theorem 5.2.1, we replace $ax - b$ as $x - br$ where r is a new variable and $r := \frac{1}{a}$.

3. Now we are considering the case $\{a + b^2 = 0, a \neq 0\}$. We compute a Gröbner basis for $\langle a+b^2, y+1, bx+1, x-br \rangle$ in $\mathbb{Q}[a, b, r][x, y]$ with respect to $\succ$. Then the algorithm GröbnerBasisB outputs $\{-ar + 1, a + b^2, y + 1, x - br\}$. Since we are considering the case $\{a + b^2 = 0, a \neq 0\}$ (and $r = \frac{1}{a}$), we do not need $-ar + 1, -a - b^2$ from the set and we can replace $x - br$ as $ax - b$. By Theorem 5.2.1, for $\alpha \in \mathbb{V}(a + b^2) \setminus \mathbb{V}(a)$,

$\{ax-b, y+1\}$ is a Gröbner basis for $\langle\sigma_\alpha(F)\rangle$ in $\mathbb{C}[x,y]$. That is, one of the segments is $(\mathbb{V}(a+b^2)\setminus\mathbb{V}(a), \{ax-b, y+1\})$.

4. Finally, we have to consider the case $\{a+b^2=0, a=0\}$. We can simplify the case into $\{a=0, b=0\}$. In this case, clearly, the Gröbner basis is $\{1\}$. Therefore, a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$ is $\big\{(\mathbb{C}^2\setminus\mathbb{V}(a+b^2), \{1\}), (\mathbb{V}(a+b^2)\setminus\mathbb{V}(a), \{ax-b, y+1\}), (\mathbb{V}(a,b), \{1\})\big\}$. That is,

$$\begin{cases} \{1\}, & \text{if } a+b^2\neq 0, \\ \{ax-b, y+1\}, & \text{if } a+b^2=0, a\neq 0, \\ \{1\} & \text{if } a=b=0. \end{cases}$$

See Figure 5.4.

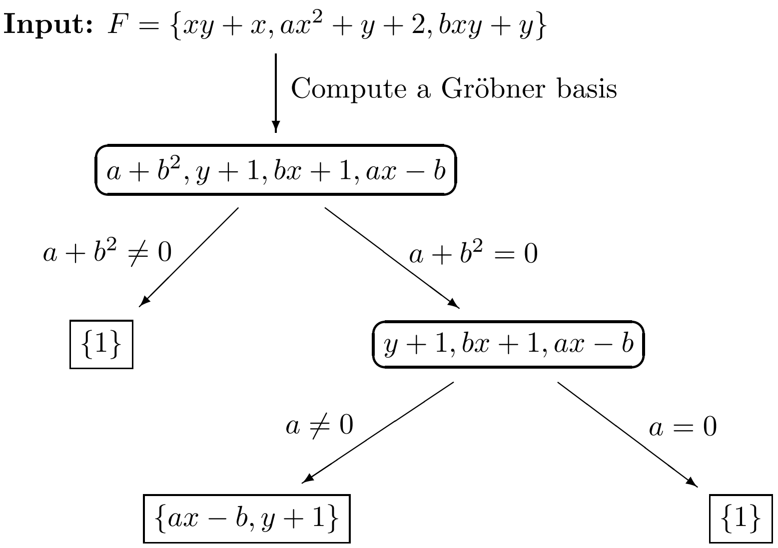

Figure 5.4

Let G be a Gröbner basis for an ideal I in $K[\bar{A}][\bar{X}]$. Suppose that

$$E := \{f\in G : \exists g\in G\setminus\{f\} s.t. \operatorname{lpp}_{\bar{A}}(f) | \operatorname{lpp}_{\bar{A}}(g)\}.$$

When we apply Theorem 5.2.1 for computing comprehensive Gröbner systems, we have to select one polynomial from G. Then, we have the following question.

Which polynomial should we select in order to compute comprehensive Gröbner systems efficiently?

In Example 5.2.4 (2), we selected $ax-b$, because $\operatorname{lpp}_{\{a,b\}}(ax-b)$ divides $\operatorname{lpp}_{\{a,b\}}(bx+1)$. In this case, we have $E=\{ax-b, bx+1\}$. If E is an empty set, then in Theorem 5.2.4, we have always $\operatorname{lpp}(\sigma_{\bar{a}}(H)) = \operatorname{lpp}(\sigma_{\bar{a}}(H'))$ for any $\bar{a}\in L^m\setminus(\mathbb{V}(\operatorname{lc}_{\bar{A}}(g))\cup\mathbb{V}(h)) = L^m\setminus\mathbb{V}(h)$. (By the theorem, clearly $\sigma_{\bar{a}}(H)$ and $\sigma_{\bar{a}}(H')$ is a Gröbner basis for $\langle F\rangle$ with respect to $\succ$ in $L[\bar{X}]$.) Namely, in this case, we should not apply Theorem 5.2.4, because we have $\operatorname{lc}_{\bar{A}}(H)\setminus K = (\operatorname{lc}_{\bar{A}}(H')\cup\{\operatorname{lc}_{\bar{A}}(g)\})\setminus K$. That is, the parameter spaces cannot be changed by the selected polynomial. In this case, we apply Suzuki-Sato's approach. **If E is not an**

empty set, then our approach (Theorem 5.2.1) works powerfully for computing comprehensive Gröbner systems. In fact, it is often happened that E is not an empty set. Our answer of the question is that "selecting one element from E". In the new algorithm which is the following, like normal strategy of Gröbner bases computation we select one polynomial from E which have the **lowest** leading power product in $\mathrm{lpp}_{\bar{A}}(E)$ with respect to a term order.

Since the following algorithm makes the branches by inequations "$\neq 0$", we need the Corollary 5.2.2 and Corollary 5.2.3. That is, these corollaries are for the branches "$\neq 0$". In the new algorithm NEW, we assume the algorithm factorize. The algorithm factorize(h) outputs a set of all irreducible factors of h in $K[\bar{A}]$ where $h \in K[\bar{A}]$.

In the remark of the algorithm NEW, we describe why we input a natural number U in the algorithm NEW.

Algorithm 5.2.5. NEW$(F, U, \succ)$

Input F: a finite subset of $K[\bar{A}][\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
U: a natural number $(< \infty)$,
Output G: a comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ$ on L^m.
begin
$G \leftarrow$ NewCGSMain$(F, \emptyset, \emptyset, 1, 0, \succ, U)$
return(G)
end

Algorithm 5.2.6. NewCGSMain$(F, L_1, L_2, D, N, \succ, U)$

Input F : a finite subset of $K[\bar{A}][\bar{X}]$,
L_1 : a finite set of polynomials in $K[\bar{A}]$ "$(= 0)$"
L_2 : a finite set of polynomials in $K[\bar{A}]$ "$(\neq 0)$",
D: a polynomial in $K[\bar{A}]$,
U: a natural number $(< \infty)$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
N: a natural number $(< U)$,
Output H: a comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ$ on $\mathbb{V}(L_1) \backslash \mathbb{V}(L_2)$.
begin
$G \leftarrow$ GröbnerBasisB$(F \cup L_1, \succ)$ in $K[r, \bar{A}][\bar{X}]$
$G^* \leftarrow$ Transform(G, D)
$G_1 \leftarrow G^* \backslash \{g : g \in G^* \cap K[\bar{A}], g \in \langle L_1 \rangle\}$
$E \leftarrow \{f \in G_1 : \exists g \in G_1 \backslash \{f\} \text{ s.t. } \mathrm{lpp}_{\bar{A}}(f) | \mathrm{lpp}_{\bar{A}}(g)\}$
if $E \neq \emptyset$ and $N \leq U$ **then**
Select q from E s.t. $\mathrm{lpp}_{\bar{A}}(q)$ is the lowest element in $\mathrm{lpp}_{\bar{A}}(E)$ with respect to $\succ$ $(r := \mathrm{lc}_{\bar{A}}(q)^{-1},$ i.e., r is the new variable.)
$q^* \leftarrow \mathrm{lpp}_{\bar{A}}(q) + r \cdot (q - \mathrm{lm}_{\bar{A}}(q))$ (i.e., $\mathrm{lc}_{\bar{A}}(q^*) = 1$)
$F^* \leftarrow (G_1 \backslash \{q\}) \cup \{q^*\}$
$\{t_1, \ldots, t_k\} \leftarrow$ factorize$(\mathrm{lc}_{\bar{A}}(q))$
$t \leftarrow t_1 \cdot t_2 \cdots t_k$
if $\mathbb{V}(L_1) \backslash \left(\mathbb{V}(t) \cup \bigcup_{s \in L_2} \mathbb{V}(s) \right) \neq \emptyset$ **then** (♣1)

$N \leftarrow N+1$
$H_1 \leftarrow \mathsf{NewCGSMain}(F^*, L_1, L_2 \cup \{t\}, \mathrm{lc}_{\bar{A}}(q), N, \succ, U)$
end-if
$H_2 \leftarrow \mathsf{NewCGSMain}(G_1, L_1 \cup \{t_1\}, L_2, \emptyset, 0, \succ, U) \cup \cdots$
$\cdots \cup \mathsf{NewCGSMain}(G_1, L_1 \cup \{t_k\}, L_2, \emptyset, 0, \succ, U)$
$H \leftarrow H_1 \cup H_2$
else

$$S \leftarrow \{h_1, .., h_l\} := \left\{ f : \mathbb{V}(f) \not\subset \bigcup_{s \in L_2} \mathbb{V}(s), f \in \mathsf{factorize}(\mathrm{lc}_{\bar{A}}(g)), \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G_1 \right\} \quad (\clubsuit 2)$$

$h \leftarrow \mathsf{LCM}(h_1, \ldots, h_l)$
$H \leftarrow \{(L_1, \{h\}, G_1)\}$
if $S \neq \emptyset$ **then**
while $S \neq \emptyset$ **do**
Select p from S; $S \leftarrow S \backslash \{p\}$
$H \leftarrow H \cup$ $\mathsf{NewCGSMain}(G_1, L_1 \cup \{p\}, L_2, \emptyset, 0, \succ, U)$
end-while
else
$H \leftarrow \{(L_1, L_2, G_1)\}$
end-if
end-if
return(H)
end

Algorithm 5.2.7. $\mathsf{Transform}(F, D)$

Input F : a finite subset of $K[r, \bar{A}][\bar{X}]$,
D: a polynomial in $K[\bar{A}]$
Output G: a finite subset of $K[\bar{A}][\bar{X}]$
begin
if $D = \emptyset$ **then**
return(F)
end-if
$G \leftarrow \emptyset$
while $F \neq \emptyset$ **do**
Select f from F; $F \leftarrow F \backslash \{f\}$
$N \leftarrow \deg_r(f)$
$L \leftarrow \mathrm{Mono}(f)$; $H \leftarrow 0$
while $L \neq \emptyset$ **do**
Select p from L ; $L \leftarrow L \backslash \{p\}$
$N_1 \leftarrow N - \deg_r(p)$
$p_1 \leftarrow$ Substitute $\frac{1}{\mathrm{lc}_{\bar{A}}(g)}$ for the variable r of p
$q_1 \leftarrow p_1 \cdot D^{N_1}$
$H \leftarrow H + q_1$
end-while
if $H \neq 0$ **then**
$G \leftarrow G \cup \{H\}$
end-if
end-while
return(G)

end

Remark : In (♣1) and (♣2), we applied the notation $\bigcup$ (union) for the algorithm. Since obviously $\mathbb{V}(h_1) \cup \mathbb{V}(h_2) = \mathbb{V}(\mathsf{LCM}(h_1, h_2))$ where $h_1, h_2 \in K[\bar{A}]$, we can apply $\mathbb{V}(\mathsf{LCM}(s_1, \ldots, s_l))$ instead of $\bigcup_{s \in L_2} \mathbb{V}(s)$ where $L_2 = \{s_1, \ldots, s_l\}$. As we used the notation "$\cup$ (union)" in Theorem 5.2.1, we followed Theorem 5.2.1 in Algorithm 5.2.6.

In Theorem 5.2.1, Corollary 5.2.2 and Corollary 5.2.3, we need to transform the set F as follows;

(1) computing a Gröbner basis H for $\langle F \rangle$, (line 1)
(2) transforming H into H' by the new variable r, (line 7)
(3) computing a Gröbner basis G' for $\langle H' \rangle$, (line 1)
(4) transforming G' into G by the algorithm Transform. (line 2)

If we do not use the natural number U in the algorithm, then by these transformations (1) – (4), we **rarely** obtain the infinite loop from 1 to 14 (recurrently) on paths of a tree structure. When we compute a Gröbner basis in $K[\bar{A}][\bar{X}]$, we apply the algorithm GröbnerBasisB. Since the algorithm GröbnerBasisB which uses a block order, regards parameters as variables, if we iterate the procedure line 1 – line 14, then we rarely see that line 1 always outputs the same Gröbner basis. In order to avoiding this infinite loop, we introduced the natural number U. This is very technical step for always terminating the algorithm.
We can apply a lot of optimization techniques to obtain small and nice outputs comprehensive Gröbner systems (like the Suzuki-Sato algorithm [SS06]). Theoretically, like Algorithm 4.4.4, we do not need factorize in order to compute comprehensive Gröbner systems. However, since factorize is very effective as one of the optimization techniques to obtain small and nice outputs, we add the algorithm factorize to the algorithm. We can also compute a radical ideal of $\langle L_1 \rangle$ to get nice outputs.

Theorem 5.2.8. The algorithm $\mathsf{NEW}(F, \succ)$ terminates. The output forms a comprehensive Gröbner system for $\langle F \rangle$ on L^m.

Proof. First we show the termination. It suffices to show the termination of $\mathsf{NewCGSMain}(F, L_1, L_2, D, N, \succ, U)$. The key part is line 5 of Algorithm 5.2.6.

($*1$) If $E = \emptyset$ and $N \leq U$, then we have to consider lines 18–29 where is Suzuki-Sato's approach. In this case, the algorithm provides one segment (see line 19).
($*2$) If $E \neq \emptyset$ and $N \leq U$, then we have to consider lines 6–16. In this case, the algorithm does not provide any segment.

NewCGSMain is a recurrence algorithm and makes the tree structure. Take an arbitrary path of the tree structure. We prove that the algorithm executes lines 17–28 ($*1$) and lines 6–15 ($*2$) a finite number of times in the path. By the same reason of the proof of Suzuki-Sato [SS06], the algorithm executes ($*1$) a finite number of times (see [SS06]). We need to prove that the algorithm executes ($*2$) a finite number of times. As we said in the remark, if we do not have the number U and N, then the algorithm does not always terminate. However, the algorithm has U which is a finite number, and thus the algorithm executes ($*2$) at most U times. Hence, this algorithm terminates.
Next we have to show the correctness. This proof is almost same as the proof of the Suzuki-Sato algorithm. See Theorem 4.4.5. We remark that in this proof we need Theorem 5.2.1,

Corollary 5.2.2 and Corollary 5.2.3.
In line 13 and 15, the algorithm computes the cases $t = \mathsf{LCM}(t_1, \ldots, t_k) \neq 0$ and $t_1 = 0, \ldots, t_k = 0$, i.e, $\bigcup_{i=1}^{k} \mathbb{V}(t_i) \cup (L^m \setminus \mathbb{V}(t)) = L^m$. By this fact and the proof of Suzuki-Sato [SS06], the output of the algorithm covers the whole parameter space. □

The algorithm NEW has been implemented in the computer algebra system Risa/Asir. In the following examples, we give outputs of the program. Note that in the program the natural number U of the algorithm NEW is fixed $U = 5$.

Example 5.2.9. Let $F = \{ax^4y + xy^2 + bx, x^3 + 2xy, bx^2 + x^2y\}$ be a subset of $\mathbb{Q}[a, b][x, y]$, a, b parameters and x, y variables. We have the lexicographic order $\succ$ such that $x \succ y$. Then, the program outputs a comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ$ as follows.

```
[b]==0,  [[1]]!=0,
[x^3+2*y*x,y*x^2,y^2*x]

[a,b+1]==0,  [[1]]!=0,
[-x^3-2*x,(-y+1)*x]

[8*b^3*a^2-b^2-2*b-1]==0,  [[b+1]]!=0,
[(-b-1)*x^2+4*b^2*a*x,(-y-b)*x]

[0]==0,  [[8*b^4*a^2-b^3-2*b^2-b]]!=0,
[x]
```

This meaning is the following;

$$\begin{cases} \{x^3 + 2xy, x^2y, xy^2\}, & \text{if } b = 0, \\ \{x^3 + 2x, (y-1)x\}, & \text{if } a = b + 1 = 0, \\ \{x\}, & \text{if } 8b^3a^2 - b^2 - 2b - 1 = 0, b + 1 \neq 0, \\ \{x\} & \text{if } 8b^4a^2 - b^3 - 2b^2 - b = 0. \end{cases}$$

Example 5.2.10. Let $F = \{ax^2 + by^2, cx^2 + y^2, 2ax - 2cy\}$ be a subset of $\mathbb{Q}[a, b, c][x, y]$, a, b, c parameters and x, y variables. We have the lexicographic order $\succ$ such that $x \succ y$. Then, the program outputs a comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ$ as follows.

```
[a,b,c]==0,  [[1]]!=0,
[y^2]

[b^2+c,a-c*b,b*a+c^2]==0,  [[a]]!=0,
[a*x-c*y]

[a,c]==0,  [[b^2+c]]!=0,
[y^2]

[a,c*b]==0,  [[c],[b^2+c]]!=0,
[c*x^2,y]

[a-c*b]==0,  [[a],[b^2+c]]!=0,
[a*x-c*y,y^2]

[a]==0,  [[c],[a-c*b]]!=0,
```

```
[c*x^2,y]

[0]==0,  [[a],[a-c*b]]!=0,
[a*x-c*y,y^2]
```

This meaning is the following;

$$\begin{cases} \{y^2\}, & \text{if } \mathbb{V}(a,b,c), \\ \{ax-cy\}, & \text{if } \mathbb{V}(b^2+c, a-cb, ba+c^2)\setminus\mathbb{V}(a) \\ \{y^2\}, & \text{if } \mathbb{V}(a,c)\setminus\mathbb{V}(b^2+c), \\ \{cx^2, y\} & \text{if } \mathbb{V}(a,cb)\setminus(\mathbb{V}(c)\cup\mathbb{V}(b^2+c)), \\ \{ax-cy, y^2\} & \text{if } \mathbb{V}(a-cb)\setminus(\mathbb{V}(a)\cup\mathbb{V}(b^2+c)), \\ \{cx^2, y\} & \text{if } \mathbb{V}(a)\setminus(\mathbb{V}(c)\cup\mathbb{V}(a-cb)), \\ \{ax-cy, y^2\} & \text{if } \mathbb{C}^3\setminus(\mathbb{V}(a)\cup\mathbb{V}(a-cb)). \end{cases}$$

This output has 7 segments. (Note that $\mathbb{V}(h_1)\cup\mathbb{V}(h_2)=\mathbb{V}(\mathsf{LCM}(h_1,h_2))$ where $h_1,h_2\in K[\bar{A}]$.) By the way, the program of the Suzuki-Sato algorithm outputs 17 segments.

As we saw chapter 4, when we compute a comprehensive Gröbner basis, we need to compute a faithful comprehensive Gröbner system. It is clear that, by the definition of faithful, the algorithm NEW can not output a faithful comprehensive Gröbner systems.

5.3 Optimization techniques

In this subsection, we introduce one optimization technique for computing a nice and small comprehensive Gröbner system. In the author's experience, we sometimes see the condition of the following lemma when we compute a Gröbner basis in $K[\bar{A}][\bar{X}]$. We can apply the lemma as one of optimization techniques.

Lemma 5.3.1. Let F be a subset of $K[\bar{A}][\bar{X}]$, $\succ$ a term order on $\mathrm{pp}(\bar{X})$ and G a Gröbner basis for $\langle F\rangle$ with respect to $\succ$ in $K[\bar{A}][\bar{X}]$. We have $g_1,\dots,g_l\in G$ such that $\mathrm{Mono}_{\bar{A}}(g_i)\subset\langle \mathrm{lpp}_{\bar{A}}(G\backslash\{g_1,\dots,g_l\})\rangle$ where $1\le i\le l$. Suppose that $\{h_1,\dots,h_s\}:=\{\mathrm{lc}_{\bar{A}}(f)\in K[\bar{A}]: f\in G\backslash\{g_1,\dots,g_l\}\}$ and $h:=\mathsf{LCM}(h_1,\dots,h_s)$. Then, for any $\bar{a}\in L^m\setminus\mathbb{V}(h)$,

(1) $\langle G\rangle$ is stable under $\sigma_{\bar{a}}$ and $\succ$,
(2) $\sigma_{\bar{a}}(G\backslash\{g_1,\dots,g_l\})$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(F)\rangle$ with respect to $\succ$ in $L[\bar{X}]$.

Proof. Since $\mathrm{Mono}_{\bar{A}}(g_i)\subset\langle \mathrm{lpp}_{\bar{A}}(G\backslash\{g_1,\dots,g_l\})\rangle$, $\mathrm{lm}_{\bar{A}}(\sigma_{\bar{a}}(g_i))\in\langle \mathrm{lpp}_{\bar{A}}(G\backslash\{g_1,\dots,g_l\})\rangle$. Hence, $\langle\sigma_{\bar{a}}(\mathrm{lm}_{\bar{A}}(G))\rangle = \langle\sigma_{\bar{a}}(\mathrm{lm}_{\bar{A}}(G\backslash\{g_1,\dots,g_l\}))\rangle$. $\sigma_{\bar{a}}(g_i)$ can be reduced to 0 by $\sigma_{\bar{a}}(G\backslash\{g_1,\dots,g_l\})$. By Theorem 4.3.2, G is a stable under $\sigma_{\bar{a}}$ and $\sigma_{\bar{a}}(G\backslash\{g_1,\dots,g_l\})$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(F)\rangle$ with respect to $\succ$ in $L[\bar{X}]$. ☐

We have already seen an easy example of this lemma in Example 5.2.4. If there exists $p\in G$ such that $\mathrm{lpp}_{\bar{A}}(p)=1$ (notations are from Lemma 5.3.1), then we do not need to consider the other polynomials $G\backslash\{p\}$ for getting a segment. In the next example, we give a more general example of Lemma 5.3.1.

Example 5.3.2. Let a,b be parameters, x,y,z variables, and $F=\{axz+bxz+a, bz+a, (a^2+a)xy\}$ in $\mathbb{Q}[a,b][x,y,z]$. By the algorithm GröbnerBasisB, we have a Gröbner basis G for $\langle F\rangle$ with respect to the lexicographic order $\succ$ such that $x\succ y\succ z$. This Gröbner basis G is the following

$$G=\left\{\begin{array}{l} g_1=bz+a,\ g_2=(-a^2-a)y,\ g_3=(-a^2-ab)x+ab, \\ g_4=(az-a)x+a,\ g_5=(b-1)axy-aby \end{array}\right\}.$$

Then, we have $g_5 \in \langle \mathrm{lpp}_{\{a,b\}}(g_1), \ldots, \mathrm{lpp}_{\{a,b\}}(g_4)\rangle$. Therefore, by Lemma 5.3.1, we can say that for $\alpha \in C^2 \backslash \mathbb{V}(ab(a+1)(a+b))$, $\{\sigma_\alpha(g_1), \ldots, \sigma_\alpha(g_4)\}$ is a Gröbner basis for $\langle \sigma_\alpha(F)\rangle$ with respect to $\succ$ in $\mathbb{C}[x,y,z]$. That is, by using Lemma 5.3.1, we can remove g_5. Therefore, we do not need to consider the case $\{\mathrm{lc}_{\{a,b\}}(g_5) = b-1 = 0\}$.

5.4 Benchmark tests and improvements

The algorithms Suzuki-Sato and NEW which contain several optimization techniques, have been implemented in Risa/Asir by the author. In this section, we compare both programs Suzuki-Sato and NEW, and notice both problems. Moreover, in the second part of this section, we improve our algorithm NEW. Note that the natural number U of the algorithm NEW is fixed $U = 5$. (We used a PC [CPU: Pentium M 1.73 GHZ, Memory 512 MB RAM, OS: Windows XP].)

Let a, b, c, d be parameters and x, y, z, w variables and $\succ$ the lexicographic order such that $x \succ y \succ z \succ w$. We have the following subsets of $\mathbb{C}[a,b,c,d][x,y,z,w]$;

$F_1 = \{ax^4y + xy^2 + bx, x^3 + 2xy, bx^2 + x^2y\}$,
$F_2 = \{ax^2y^3 + by + y, x^2y^2 + xy + 2, ax^2 + by + 2\}$,
$F_3 = \{ax^4 + cx^2 + b, bx^3 + x^2 + 2, cx^2 + dx\}$,
$F_4 = \{ax^3y + cxy^2, x^4y + 3dy, cx^2 + bxy, x^2y^2 + ax^2, x^5 + y^5\}$.

The following table includes timing date of the programs in each problems.

Problem	**Algorithm**	**Segments**	**time (sec.)**
F_1	Suzuki-Sato	7	0.079
	NEW	4	0.031
F_2	Suzuki-Sato	4	0.047
	NEW	6	0.093
F_3	Suzuki-Sato	31	2.421
	NEW	22	2.203
F_4	Suzuki-Sato	39	1.391
	NEW	15	0.234

In the table above, we can see that our algorithm NEW runs faster than the algorithm Suzuki-Sato in the problems F_1, F_3, F_4. Furthermore, the numbers of segments are smaller than Suzuki-Sato's outputs. We remark that NEW does not always run faster than Suzuki-Sato. See the problem F_2. However, in many cases, NEW runs faster than Suzuki-Sato. Especially, if the number of parameters is greater than the number of variable, i.e., $|\bar{A}| > |\bar{X}|$, then NEW is much more efficient than Suzuki-Sato for computing comprehensive Gröbner systems.

Next, we consider more difficult problems. We use the lexicographic order $\succ$ such that $x \succ y \succ z \succ w$. We have the following subsets of $\mathbb{C}[a,b,c,d][x,y,z,w]$;

$F_5 = \{ax^2y + bx + y^3, ax^2y + bxy, y^2 + bx^2y + cxy\}$,
$F_6 = \{ax^2y^2 + bxz^2, bxy^2 + cx^2 + 2, cx^2 + by^2z\}$,
$F_7 = \{x^4 + ax^3 + bx^2 + cx + d, 4x^3 + 3ax^2 + 2bx + c\}$,
$F_8 = \{x^3 - a, y^4 - b, x + y - az\}$,

$F_9 = \{ax^2 + by, cw^2 + z, (x - z)^2 + (y - w)^2, 2dxw - 2by\}$.

Problem	Algorithm	Segments	time (sec.)
F_5	Suzuki-Sato	14	0.219
	NEW	6	0.109
F_6	Suzuki-Sato	11	0.125
	NEW	7	0.094
F_7	Suzuki-Sato	875	92.88
	NEW	17	0.312
F_8	Suzuki-Sato	7	0.282
	NEW	$--$	> 30 m
F_9	Suzuki-Sato	$--$	> 30 m
	NEW	$--$	> 30 m

In the problems F_5, F_6, F_7, NEW runs faster than Suzuki-Sato. Why does the program NEW run faster? Because NEW compute segments whose number is smaller than that of Suzuki-Sato's. Look at "**Segments**" of the table. In the problem F_8, NEW cannot return between 30 minutes. **Why?** Because the program of the algorithm GröbnerBasisB need a lot of time for computing a Gröbner basis in $\mathbb{C}[r, a, b][x, y, z]$ where r is the new variable. In Algorithm 5.2.6 line 6, we have to make the new variable r. This is dangerous when we compute a Gröbner basis in polynomial rings. The problems of the algorithms Suzuki-Sato and NEW, are the following.

- Suzuki-Sato creates a lot of segments (parameter spaces).
- NEW (sometimes) needs expensive Gröbner bases computations (however, the number of segments is not big).

Now we improve the algorithm NEW. Look at line 6 of Algorithm 5.2.6. In the line, we must select one polynomial. Actually, in the problem F_8, the program NEW selected a bad polynomial. Therefore, the program could not return. We should select a good polynomial from E for computing a comprehensive Gröbner system. In Algorithm 5.2.6, we define the following set as E

$$E := \{f \in G : \exists g \in G \backslash \{f\} \text{ s.t. } \operatorname{lpp}_{\bar{A}}(f) | \operatorname{lpp}_{\bar{A}}(g)\}.$$

In fact, in the problem F_8, the program NEW selects a polynomial f from E which has 12 monomials, i.e., the cardinality of $\mathrm{Mono}_{\{a,b\}}(f)$ is 12. Since this polynomial f is very long (and we multiply f by the new variable r), the Gröbner bases computation become very expensive. The author has computed a lot of comprehensive Gröbner systems by the program. By these computational experiments, the author was noticed that "**we should not select a long polynomial from** E." That is, in concerning speed, we need to consider **how many monomials the selected polynomial has**.

Now, in Algorithm 5.2.6, we can replace E to the following set

$$E_s := \{f \in G : \sharp(\mathrm{Mono}_{\bar{A}}(f)) \leq s, \exists g \in G \backslash \{f\} \text{ s.t. } \operatorname{lpp}_{\bar{A}}(f) | \operatorname{lpp}_{\bar{A}}(g)\}.$$

where $s \in \mathbb{N}$ and $\sharp(\mathrm{Mono}_{\bar{A}}(f))$ is the cardinality of the set $\mathrm{Mono}_{\bar{A}}(f)$. Clearly, we have $E_s \subseteq E$. In this case, we rename the algorithm NEW as NEW$[s]$. In the following table we also consider F_{10} and F_{11} which are the following;

$F_{10} = \{ax^2y + bcx + y^3, cx^2y + bxy, ay^2 + bx^2y + cxy\}$,

$F_{11} = \{(a+bc)x^4 + y^3, cx^2y + bxy, (b+c)x^2y + ay^2\}$.

Problem	**Algorithm**	**Segments**	**time (sec.)**
F_7	NEW[1]	621	91.39
	NEW[2]	53	1.141
	NEW[3]	17	0.359
F_8	Suzuki-Sato	7	0.328
	NEW[1]	7	0.375
	NEW[2]	7	0.375
F_9	Suzuki-Sato	$--$	> 30 m
	NEW[1]	458	133.2
F_{10}	Suzuki-Sato	37	1.172
	NEW	$--$	> 30 m
	NEW[1]	29	0.671
F_{11}	Suzuki-Sato	$--$	> 30 m
	NEW	$--$	> 30 m
	NEW[1]	239	7.828

Remark: If we apply $s = 1$ for NEW[s], then we do not need the new variable r. However, like the problem F_7, sometimes the algorithm creates a lot of segments. In our algorithm, selecting a good polynomial from E (or E_r) is very important to compute a comprehensive Gröbner system, efficiently.
The method of selecting a polynomial from E (or choosing s) for computing comprehensive Gröbner systems efficiently, is a open problem.

We introduced a new algorithm for computing comprehensive Gröbner systems. In many cases, our algorithm is more efficient than the Suzuki-Sato algorithm. That is, our algorithm creates smaller outputs, and runs faster than the Suzuki-Sato algorithm. In general, if the number of parameters is greater than the number of variables, then the Suzuki-Sato algorithm is slower than other existing algorithm. This is because the Suzuki-Sato algorithm creates overmuch segments. (If the number of parameters is smaller than the number of variables, then the algorithm is very fast.) However, in this case, our algorithm is even faster. This is the main advantage of our algorithm compared to the Suzuki-Sato algorithm.

Chapter 6

Comprehensive Gröbner bases and von Neumann regular rings

In this chapter, we describe the relations between comprehensive Gröbner bases and non-parametric Gröbner bases over commutative von Neumann regular rings. Commutative von Neumann regular rings can be viewed as certain subdirect products of fields. So in some sense they can code arbitrary sets of fields. In 1987, Weispfenning studied and constructed the theory of Gröbner bases in polynomial rings over a commutative von Neumann regular ring. In 1992, Weispfening also introduced, constructed and studied comprehensive Gröbner bases for parametric polynomial ideals. Here, we show that there is a surprisingly close relationship between his two works. Thus, we show that Gröbner bases over commutative von Neumann regular rings do in fact cover parametric Gröbner bases over commutative von Neumann rings. We call the parametric Gröbner bases "alternative comprehensive Gröbner bases (ACGB)". In papers [SS02, SS03, SSN02, SSN03b, SSN03a, SS04, Wei02b, Wei06], these results are shown.

In the second part of this chapter, we present the special type of comprehensive Gröbner bases. In construction of parametric Gröbner bases, we usually assume that parameters can take arbitrary values. In case, however, there exist some constraints among parameters, it is more natural to construct comprehensive Gröbner bases for only parameters satisfying such constraints. Using this idea, we formalized comprehensive Gröbner bases in terms of ACGB. In the author's papers [SSN03a, SSN03b, Nab05a, Nab05b], these results were presented.

6.1 Von Neumann regular rings and Boolean algebra

In this section we describe relations between commutative von Neumann regular rings and Boolean algebra. Some of the facts of this section are presented in Saracino and Weispfenning [SW75, Lou79], and some books of "Boolean algebra", for instance [BS80]. First, we give a definition of "commutative von Neumann regular rings".

Definition 6.1.1 (commutative von Neumann regular rings [SW75, Wei87]). A commutative ring R with identity 1 is called a commutative **von Neumann regular ring** if it has the following property:

$$\forall a \in R \ \ \exists b \in R \text{ such that } \quad a^2 b = a.$$

For such b, $a^* := ab$ and $a^{-1} := ab^2$ are uniquely determined and satisfy $aa^* = a$, $aa^{-1} = a^*$ and $(a^*)^2 = a^*$ is idempotent of a, a^{-1} the quasi inverse of a.

Note that every direct product of fields is a commutative von Neumann regular ring. Conversely, any commutative von Neumann regular ring is known to be isomorphic to a subring of a direct product of fields [SW75].
In this chapter, we assume that R is a commutative von Neumann regular ring.

Example 6.1.2. Take $R = \mathbb{Q}^3$ and define for $a = (a_1, a_2, a_3) \in \mathbb{Q}^3$, $a^{-1} := (y_1, y_2, y_3)$ where for $i \in \{1, 2, 3\}$

$$y_i = \begin{cases} 0, & \text{if } a_i = 0, \\ \frac{1}{a_i}, & \text{otherwise.} \end{cases}$$

We see that for all $a \in \mathbb{Q}^3$ there exists $b \in \mathbb{Q}^3$, namely

$$b := a^{-1} \text{ such that } a^2 b = a.$$

Therefore, $\mathbb{Q}^3$ is a von Neumann regular ring. (We consider $0^{-1} := 0$.)

A definition of Boolean algebra is the following.

Definition 6.1.3. $\mathbb{B} := \langle B, \wedge, \vee, \neg, 0, 1\rangle$ is called a **Boolean algebra** if $\mathbb{B}$ satisfies the following property;

1. $x \vee x = x \wedge x = x$, for $x \in B$,
2. $x \vee y = y \vee x$, $x \wedge y = y \wedge x$, for $x, y \in B$,
3. $(x \vee y) \vee z = x \vee (y \vee z)$, $(x \wedge y) \wedge z) = x \wedge (y \wedge z)$, for $x, y, z \in B$,
4. $(x \vee y) \wedge x = x$, $(x \wedge y) \vee x = x$, for $x, y \in B$,
5. $(x \wedge y) \vee z = (x \vee z) \wedge (y \vee z)$, $(x \vee y) \wedge z = (x \wedge z) \vee (y \wedge z)$, for $x, y, z \in B$,
6. $x \vee \neg x = 1$, $x \wedge \neg x = 0$, for $x \in B$.

Definition 6.1.4. Let R be a commutative von Neumann regular ring. Let $A = \{x \in R | x^2 = x\}$ be a set of idempotents of R. Define on A the operations $\neg, \wedge, \vee$ by $\neg a = 1 - a$, $a \wedge b = ab$ and $a \vee b = a + b - ab$. Then $B(R) := \langle A, \neg, \wedge, \vee, 1, 0\rangle$ is called the **Boolean algebra** of R. The set A is called the **carrier set** of $B(R)$.

Note that the carrier set of $B(\mathbb{Q}^3)$ is $\{(x_1, x_2, x_3) \,|\, x_1, x_2, x_3 \in \{0, 1\}\}$.

Definition 6.1.5 ([BS80]). Let $\mathbb{B} = \langle B, \wedge, \vee, \neg, 0, 1\rangle$ be a Boolean algebra. A subset I of B is called an **ideal** of $\mathbb{B}$ if

1. $0 \in I$,
2. $a, b \in I \implies a \vee b \in I$,
3. $(a \in I \text{ and } b = a \wedge b) \implies b \in I$.

Note that definition 6.1.5 property (3) is equivalent to ;

$$a \in I \text{ and } \ b \in B \Rightarrow a \wedge b \in I.$$

It is important to consider prime ideals of Boolean algebra in order to construct an algorithm for computing Gröbner bases in polynomial rings over a commutative von Neumann regular ring. We give a definition of prime ideals of Boolean algebra and the examples.

Definition 6.1.6 ([BS80]). An ideal I of a Boolean algebra is called a **prime ideal** if $1 \notin I$ and $a \wedge b \in I$ implies $a \in I$ or $b \in I$.

Example 6.1.7. Let $B(\mathbb{Q}^3) := \langle B, \wedge, \vee, \neg, 0, 1\rangle$. Then, prime ideals of $B(\mathbb{Q}^3)$ are

$$P_1 := \{(0, x_2, x_3) \,|\, x_2, x_3 \in \{0, 1\}\},$$
$$P_2 := \{(x_1, 0, x_3) \,|\, x_1, x_3 \in \{0, 1\}\},$$
$$P_3 := \{(x_1, x_2, 0) \,|\, x_1, x_2 \in \{0, 1\}\}.$$

The set $Q_i := \{(x_1, 0, 0) \,|\, x_1 \in \{0, 1\}\}$ is not a prime ideal in $B(\mathbb{Q}^3)$, because $(1, 0, 1) \wedge (1, 1, 0) = (1, 0, 0) \in Q_1$, but $(1, 0, 1), (1, 1, 0) \neq Q_1$.

Proposition 6.1.8. For $i = 1, \dots, n$ let $P_i = \{(x_1, \dots, x_n) \in \{0, 1\}^n \,|\, x_i = 0\}$. The set of all prime ideals of $B(\mathbb{Q}^n) = \langle B, \wedge, \vee, \neg, 0, 1\rangle$ is $\{P_1, \dots, P_n\}$ where $B = \{(x_1, \dots, x_n) \,|\, x_i \in \{0, 1\}\}$.

Proof. Let S be an arbitrary non-empty subset of $\{1, \cdots, n\}$. We denote for all $j \in S$,

$$I_j := \{(x_1, \dots, x_n) \in \{0, 1\}^n \,|\, x_j = 1\}.$$

Then, I_j is not an ideal in $B(\mathbb{Q}^n)$, because I_j does not satisfy definition 5 (3).
Let L be an arbitrary subset of $\{1, \cdots, n\}$. Then we denote

$$I_L := \{(x_1, \dots, x_n) \in \{0, 1\}^n \,|\, x_j = 0, j \in L\}.$$

First we prove that if $|L| = 1$ ($|L|$ is the cardinality of L), then I_L is a prime ideal.
Let $L = \{i\} \subseteq \{1, \dots, n\}$, then a_i and b_i are the ith coordinate of $a, b \in B$. Take $a \wedge b \in I_L$, then $a_i \wedge b_i = 0$. Hence a_i or b_i must be 0. Therefore $a \in I_L$ or $b \in I_L$ and thus I_L is a prime ideal. Second, we prove that if $|L| > 1$, then I_L is not a prime ideal. Take $j_1, j_2 \in L$, $j_1 \neq j_2$. Let a_j be the jth coordinate of $a \in B$. Take $f \in \{(x_1, \dots, x_n) \in \{0, 1\} \,|\, x_{j_1} = 1, x_{j_2} = 0\}$ and $g = \{(x_1, \dots, x_n) \in \{0, 1\}^n \,|\, x_{j_1} = 0, x_{j_2} = 1\}$. Then $f \wedge g \in I_L$, but $f_{j_1} \wedge g_{j_1} = f_{j_2} \wedge g_{j_2} = 0$. Hence $f, g \notin I_L$. Therefore I_L is a prime ideal of $B(\mathbb{Q}^n)$. The set of all prime ideals of $B(\mathbb{Q}^n)$ is $\{P_1, \dots, P_n\}$. ☐

Definition 6.1.9. An ideal I of a Boolean algebra B is called a **maximal** if there exists no ideal J with $I \subsetneq J \subsetneq B$.

Theorem 6.1.10 (Theorem 3.12 [BS80]). Let I be an ideal of $\mathbb{B} := \langle B, \wedge, \vee, \neg, 0, 1\rangle$. Then, I is a maximal ideal of $\mathbb{B}$ if and only if for any $a \in B$, exactly one of $a, \neg a$ belongs to I.

Lemma 6.1.11 (Corollary 3.13 [BS80]). Let I is an prime ideal of Boolean algebra $\langle B, \wedge, \vee, \neg, 0, 1\rangle$ if and only if I is a maximal ideal.

Proof. ($\Leftarrow$) Suppose I is a maximal ideal with

$$a \wedge b \in I, \text{ for } a, b \in B.$$

As

$$(a \wedge b) \vee (\neg a \vee \neg b) = 1 \notin I,$$

we have

$$\neg a \vee \neg b \notin I,$$

hence,

$$\neg a \notin I \text{ or } \neg b \in I.$$

By theorem 6.1.10 either

$$a \in I \text{ or } b \in I.$$

($\Rightarrow$) Since $0 \in I$, given $a \in B$ we have

$$a \wedge \neg a \in I.$$

Since I is a prime ideal,

$$a \in I \text{ or } \neg a \in I.$$

As

$$a \vee \neg a = 1 \notin I,$$

one of $a, \neg a$ belong to I. By theorem 6.1.10, I is a maximal ideal. □

For a Boolean algebra B, $\mathrm{Spec}(B)$ denotes the prime spectrum of B, i.e., the set of all prime ideals of B. The set of all maximal ideals of B is denoted by $\mathrm{St}(B)$. (Actually, in this case, by Lemma 6.1.11 we can say $\mathrm{Spec}(B) = \mathrm{St}(B)$.)

Theorem 6.1.12 (Saracino-Weispfenning[SW75]). For a maximal ideal I of $B(R)$, $I_R = \{xy \mid x \in R, y \in I\}$ (then I_R is a maximal ideal of R). If we define a map Φ from R into $\prod_{I \in St(B(R))} R/I_R$ by $\Phi(x) = \prod_{I \in \mathrm{St}(B(R))} [x]_{I_R}$, then Φ is a ring isomorphism.

An example of the theorem is the following.

Example 6.1.13. Let's consider $\mathbb{Q}^3$. From Proposition 6.1.8 and Lemma 6.1.11, we know

$$St(B(\mathbb{Q}^3)) = \mathrm{Spec}(B(\mathbb{Q}^3)) = \{P_1, P_2, P_3\}.$$

Let $S_i := \{xy | x \in \mathbb{Q}^3, y \in P_i\}$ for each $i = 1, 2, 3$. Then, by Theorem 6.1.12,

$$\mathbb{Q}^3 \cong \mathbb{Q}^3/S_1 \times \mathbb{Q}^3/S_2 \times \mathbb{Q}^3/S_3.$$

Obviously, $\mathbb{Q}^3/S_3$ is isomorphic to $\mathbb{Q}$. By this identification Φ can be seen as the identity map on $\mathbb{Q}^3$.

Let $R_p := R/(p_R)$ where $p \in \mathrm{Spec}(B(R))$. Then for a subset Q of R and $p \in \mathrm{Spec}(B(R))$ we let $\Phi_p : R \longrightarrow R_p$ be the canonical homomorphism, and Q_p the image of Q under Φ_p.

Remark: Let P_1 be as in example 6.1.7. Obviously ${\mathbb{Q}^3}_{P_1}$ is isomorphic to $\mathbb{Q}$ and thus Φ_{P_1} can be seen as the projection map

$$\begin{aligned} \Phi_{P_1} : \mathbb{Q}^3 &\to \mathbb{Q}, \\ (a, b, c) &\mapsto (c), \end{aligned}$$

where $a, b, c \in \mathbb{Q}$.

6.2 Gröbner bases over von Neumann regular rings

Here, we describe the theory of Gröbner bases in polynomial rings over a commutative von Neumann regular ring. The theory has been studied by Weispfenning [Wei87]. Before describing the theory, we define the notations for $R[\bar{X}]$.

Definition 6.2.1. Let f be a non zero polynomial in $R[\bar{X}]$ and $\succ$ be an arbitrary order on the set of power products.

1. The **set of power products** of f that appear with a non-zero coefficient, is written $\mathrm{pp}(f)$.
2. The biggest power product of $\mathrm{pp}(f)$ with respect to $\succ$ is denoted by $\mathrm{lpp}(f)$ and is called the **leading power product** of g with respect to $\succ$.
3. The coefficient corresponding to $\mathrm{lpp}(f)$ is called **leading coefficient** of f with respect to $\succ$.
4. The product $\mathrm{lc}(f)\,\mathrm{lpp}(f)$ is called the **leading monomial** of f with respect to $\succ$.

Example 6.2.2. To illustrate, let $f = (2,0,\frac{2}{3})xy^2z + (-1,-3,0)z^2 + (\frac{1}{2},0,5)x^3 + (3,4,0)x^2z^2$ in $\mathbb{Q}^3[x,y,z]$ and let $\succ$ denote the lexicographic order with $x \succ y \succ z$. Then

$$\begin{aligned}
\mathrm{pp}(f) &= \{xy^2z, z^2, x^3, x^2z^2\},\\
\mathrm{lc}(f) &= \left(\frac{1}{2}, 0, 5\right),\\
\mathrm{lpp}(f) &= x^3,\\
\mathrm{lm}(f) &= \left(\frac{1}{2}, 0, 5\right) x^3.
\end{aligned}$$

In this section and the next section, Greek letters α, β, γ are used for power products, Roman letters a, b, c for elements of R, f, g, h for polynomials over a commutative von Neumann regular ring R. In order to describe the theory, we need a reduction system as the normal polynomial ring $K[\bar{X}]$. Note that R is not an integral domain.

Definition 6.2.3 (reduction [Wei87]). For a polynomial $f = a\alpha + g$ with $\mathrm{lm}(f) = a\alpha$, a monomial **reduction** $\rightarrow_f$ is defined as follows:

$$b\alpha\beta + h \rightarrow_f b\alpha\beta + h - ba^{-1}\beta(a\alpha + g) = (b\alpha\beta - ba^{*}\alpha\beta) + h - g$$

where $ab \neq 0$ (and $b\alpha\beta$ need not be the leading monomial of $b\alpha\beta+h$). (See Definition 6.1.1 for the notation a^*.)
Actually, we can repeat this reduction step until we have a polynomial which can not reduced by f. In this case, we use the notation $\xrightarrow{*}_f$.

An example of the reduction is the following.

Example 6.2.4. Let $f = (2,0)x^2y + (2,1)y$, $g = (3,2)x^2y^2 \in \mathbb{Q}^2[x,y]$ with the lex-order $x \succ y$.

$$\begin{aligned}
g \longrightarrow_f g &- (3,2) \cdot \left(\frac{1}{2}, 0\right) \cdot y \cdot f\\
&= (3,2)x^2y^2 - \big((3,0)x^2y^2 + (3,0)y^2\big)\\
&= (0,2)x^2y^2 + (-3,0)y^2
\end{aligned}$$

In the above example, we have $\mathrm{lpp}(g) = x^2y^2$, but after reduction by f, we have still $\mathrm{lpp}((0,2)x^2y^2 + (-3,0)y^2) = x^2y^2$. (Note that the first coordinate was reduced by f.) This property is not good for computing Gröbner bases in the ring, and thus we need the following definition.

Definition 6.2.5 (boolean closed [Wei87]). A polynomial f is called **boolean closed** if $(\mathrm{lc}(f))^*f = f$.

Example 6.2.6. Let $f = (0,-1)x^2y + (0,3)xy + (0,2)$ in $\mathbb{Q}^2[x,y]$. Then $(\mathrm{lc}(f))^* = (0,1)$, and $(\mathrm{lc}(f))^*f = f$. Hence f is a boolean closed polynomial.

A reduction $\rightarrow_F$ by a set F of polynomials in also naturally defined.

Remark: If polynomial g is not a boolean closed polynomial, then we have a problem which we have already seen in Example 6.2.4. Therefore, we need only boolean closed polynomials to compute reductions. If we have a non-boolean closed polynomial g, then we have to classify g into a set of boolean closed polynomials for computing reductions.

We are able to construct a set of boolean closed polynomials H from a given finite set of polynomials $F \subset R[\bar{X}]$ such that ideal $\langle F \rangle = \langle H \rangle$. Though H is not determined uniquely, we use the notation $\mathrm{BC}(F)$ (boolean closure of F) to denote one of such H. We need a set of boolean closed polynomials for reductions. The following algorithm provides a set $\mathrm{BC}(F)$ of boolean closed polynomials for a given subset F of $R[\bar{X}]$ such that $\langle F \rangle = \langle \mathrm{BC}(F) \rangle$.

Let q be a polynomial in $R[\bar{X}]$. We denote by $q - (\mathrm{lc}(q)^*)q$ the **boolean remainder** $\mathrm{br}(q)$ of q, and by $\mathrm{lc}(q)^* q$ the **boolean closure** $\mathrm{bc}(q)$ of q. So for $q \neq 0$, $\deg_{\bar{X}}(\mathrm{br}(q)) \leq \deg_{\bar{X}}(q)$ and $q = \mathrm{bc}(q) + \mathrm{br}(q)$.

Algorithm 6.2.7. $\mathsf{BC}(F, \succ)$ (Boolean Closure[Wei87])

Input: F: a finite set of polynomials in $R[\bar{X}]$
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output: Q: a finite set of boolean closed polynomials in $R[\bar{X}]$ with $\langle F \rangle = \langle Q \rangle$.
begin
$Q \longleftarrow \emptyset$; $H \longleftarrow F$
  while $H \neq \emptyset$ **do**
    Select g from H; $H \leftarrow H \backslash \{g\}$
    $Q \longleftarrow Q \cup \{\mathrm{bc}(g)\}$
      if $\mathrm{br}(g) \neq 0$ **then**
        $H \longleftarrow H \cup \{\mathrm{br}(g)\}$
      end-if
  end-while
return(Q)
end

Example 6.2.8. Let $f := (1,3,0)x^2y + (3,1,1)xy + (0,0,1)y + (-1,3,-2)$ and $g := (-1,0,2)x^2 + (-1,2,2)xy + (3,2,0)x + (2,0,0)$ in $\mathbb{Q}^3[x,y]$ and $\succ$ is the lexicographic order such that $x \succ y$. Then by the algorithm, we have

$$\mathrm{BC}(\{f\}) = \{(1,3,0)x^2y + (3,1,0)xy + (-1,3,0), (0,0,1)xy + (0,0,1)y + (0,0,-2)\},$$
$$\mathrm{BC}(\{g\}) = \{(-1,0,2)x^2 + (-1,0,2)xy + (3,0,2)x + (2,0,0), (0,2,0)xy + (0,2,0)x\}.$$

Hence,

$$\mathrm{BC}(\{f,g\}) = \mathrm{BC}(\{f\}) \cup \mathrm{BC}(\{g\}).$$

Theorem 6.2.9 ([Wei87]). For any finite set F of polynomials, we can construct a finite set H of boolean closed polynomials such that ideal $\langle F \rangle = \langle H \rangle$.

We can naturally define Gröbner bases in $R[\bar{X}]$, like the case polynomial rings over a field $K[\bar{X}]$, as follows.

Definition 6.2.10 (Gröbner bases). Fix a term order on $\mathrm{pp}(\bar{X})$. A finite set $G = \{g_1, \ldots, g_s\}$ of an ideal I is said to be a **Gröbner basis** for I with respect to $\succ$ if

$$\langle \mathrm{lm}(g_1), \ldots, \mathrm{lm}(g_l) \rangle = \mathrm{lm}(I).$$

Definition 6.2.11 (S-polynomial [Wei87]). For each pair of polynomials $f = a\alpha + f'$ and $g = b\beta + g'$ where $\mathrm{lm}(f) = a\alpha$, $\mathrm{lm}(g) = b\beta$. An **S-polynomial** of f and g (written : $\mathrm{SP}(f,g)$) is defined as follows:

$$\begin{aligned}\mathrm{SP}(f,g) &= b\frac{\mathrm{lcm}(\alpha,\beta)}{\beta} \cdot f - a\frac{\mathrm{lcm}(\alpha,\beta)}{\alpha} \cdot g \\ &= b\frac{\mathrm{lcm}(\alpha,\beta)}{\beta} \cdot f' - a\frac{\mathrm{lcm}(\alpha,\beta)}{\alpha} \cdot g'.\end{aligned}$$

Example 6.2.12. Let $f = (1,0,2)x^2y + (2,3,-1)xy + (2,1,0)y$, $g = (2,1,1)x^2 + (2,3,0)x + (1,1,1)y$ in $\mathbb{Q}^3[x,y]$. Then the S-polynomial of f and g are is:

$$\begin{aligned}\mathrm{SP}(f,g) &= (2,1,1) \cdot f - (1,0,2) \cdot g \\ &= (4,3,-1)xy + (4,1,0)y + (1,0,0)xy + (1,0,2)y^2 \\ &= (5,3,-1)xy + (1,0,2)y^2 + (4,1,0)y.\end{aligned}$$

Theorem 6.2.13 ([Wei87]). Let $G \subset R[\bar{X}]$ be a finite set of boolean closed polynomials. Then G is a Gröbner basis if and only if $\mathrm{SP}(f,g) \longrightarrow_G 0$ for any pair f and g of polynomials in G.

Now, we can construct an algorithm for computing Gröbner bases in $R[\bar{X}]$. This algorithm is essentially same as the Buchberger algorithm [Buc65]. We remark again that R is not an integral domain.

Algorithm 6.2.14. GBovN$(F, \succ)$ (Gröbner basis over a von Neumann regular ring)

Input: F: a finite list of polynomials in $R[\bar{X}]$
  $\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output: G: Gröbner basis of $\langle F \rangle$ with respect to $\succ$ in $R[\bar{X}]$ with $\langle F \rangle = \langle G \rangle$
begin
$G \longleftarrow \mathrm{BC}(F, \succ)$; $C \longleftarrow \{\{g,f\} | g, f \in G\}$
  while $C \neq \emptyset$ **do**
    Select $\{h_1, h_2\}$ from C; $C \leftarrow C \backslash \{\{h_1, h_2\}\}$
      if $\mathrm{SP}(h_1,h_2)\downarrow_G \neq 0$ **then**
        $G \longleftarrow G \cup \mathrm{BC}(\mathrm{SP}(h_1,h_2)\downarrow_G, \succ)$ (see below $(*)$)
        $B \longleftarrow \{(f,k) | k \in \mathrm{BC}(\mathrm{SP}(h_1,h_2)\downarrow_G, \succ), f \in G\}$
        $C \longleftarrow C \cup B$
      end-if
  end-while
return(G)
end
($(*)$ $h\downarrow_F$ denotes a normal form of h modulo $\rightarrow_F$, i.e., $h\downarrow_F$ is irreducible modulo $\rightarrow_F$.)

If a finite set G is a Gröbner basis and reduced, then G is called **reduced Gröbner basis** (i.e., $\forall p \in G$, p cannot be reduced by $G \backslash \{p\}$). We have the following property.

Theorem 6.2.15 ([Wei87]). Let G be a reduced Gröbner basis, then any element of G is boolean closed.

In $K[\bar{X}]$, reduced Gröbner bases serve us as canonical forms of Gröbner bases, however we have to be careful in $R[\bar{X}]$.

Definition 6.2.16 ([Sat98]). A polynomial f is called **monic** if it satisfies $\mathrm{lc}(f) = (\mathrm{lc}(f))^*$.

Definition 6.2.17 ([Sat98, Wei87]). A reduced Gröbner basis $G \subset R[\bar{X}]$ is called a **stratified** Gröbner basis, when it satisfies the following two properties.

1. Every element of G is monic.
2. $\mathrm{lpp}(f) \neq \mathrm{lpp}(g)$ for any distinct elements f and g of G.

Theorem 6.2.18 ([Wei87]). A stratified Gröbner basis is determined uniquely, i.e., two stratified Gröbner bases $G, G' \subset R[\bar{X}]$ with $\langle G\rangle = \langle G'\rangle$ must be identical.

Remember that in Theorem 6.1.12, I_R depends on I. So let $R_p := R/(p_R)$ where $p \in \mathrm{Spec}(B(R))$. Then for a subset Q of R and $p \in \mathrm{Spec}(B(R))$ we let $\Phi_p : R \to R_p$ be the canonical homomorphism, and Q_p the image of Q under Φ_p. The following theorem is quite important for constructing (alternative) comprehensive Gröbner bases. In the next theorem, we use the notations R_p and $G_p := \Phi(G)$.

Theorem 6.2.19 (Weispfenning [Wei87]). Let G be a finite set of non-zero polynomials in $R[\bar{X}]$.

1. If G is a set of boolean closed polynomials, then G is a Gröbner basis if and only if for all $p \in \mathrm{Spec}(B(R))$, G_p in a Gröbner basis in $R_p[\bar{X}]$.
2. G is a reduced Gröbner basis if and only if G is a set of boolean closed polynomials and for all $p \in \mathrm{Spec}(B(R))$, G_p is a reduced Gröbner basis in $R_p[\bar{X}]$.
3. If for all $h \in R[\bar{X}]$, G is a Gröbner basis, then $(h{\downarrow_G})_p = h_p \downarrow_{G_p}$.

6.3 Criteria for computing Gröbner bases

In [Buc79, Buc70], Buchberger has introduced criteria for computing Gröbner bases in $K[\bar{X}]$. It is possible to generalize the criteria to $R[\bar{X}]$. In this section, we present criteria for computing Gröbner bases in $R[\bar{X}]$. That is, we describe techniques for removing unnecessary critical pairs. By the criteria, we can improve Algorithm 6.2.14 for computing Gröbner bases efficiently. The facts of this section is from the author's paper [Nab05b].

Theorem 6.3.1 (First Criterion). Let f and g be non-zero boolean closed polynomials in $R[\bar{X}]$. If f and g have $\mathrm{lc}(f)\,\mathrm{lc}(g) = 0$ or disjoint leading power products, then $\mathrm{SP}(f, g) \xrightarrow{*}_{\{f,g\}} 0$.

Proof. If $\mathrm{lc}(f)\,\mathrm{lc}(g) = 0$, then by Definition 6.2.11, $\mathrm{SP}(f, g) = 0$. Assume that f and g have disjoint leading power products. For all $p \in \mathrm{Spec}(B(R))$, then we have f_p, g_p in $R_p[\bar{X}]$. If one of f_p and g_p is 0, then $\mathrm{SP}(f, g)_p = 0$ in $R_p[\bar{X}]$. (R is not an integral domain.) If $f_p \neq 0$ and $g_p \neq 0$, then we can apply the original Buchberger's criterion [Buc79]. Hence, we have $\mathrm{SP}(f_p, g_p)\xrightarrow{*}_{\{f_p,g_p\}}0$ in $R_p[\bar{X}]$. Therefore, $\{f_p, g_p\}$ is a Gröbner basis for ideal $\langle f_p, g_p\rangle$ in $R_p[\bar{X}]$. By Theorem 6.2.19, $\{f, g\}$ is a Gröbner basis in $R[\bar{X}]$. Therefore, $\mathrm{SP}(f, g) \xrightarrow{*}_{\{f,g\}} 0$. □

Theorem 6.3.2 (Second Criterion). Let p, g_1 and g_2 be non-zero boolean closed polynomials in $R[\bar{X}]$ such that the following hold:

1. $\mathrm{lpp}(p)|\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2))$, and
2. $\mathrm{lc}(g_1)^* \, \mathrm{lc}(p)^* \, \mathrm{lc}(g_2)^* = \mathrm{lc}(g_1)^* \, \mathrm{lc}(g_2)^*$.

Then, $\mathrm{SP}(g_1, g_2)$ is generated by an ideal $\langle \mathrm{SP}(g_1, p), \mathrm{SP}(g_2, p)\rangle$ in $R[\bar{X}]$. That is, $\mathrm{SP}(g_1, g_2) \in \langle \mathrm{SP}(g_1, p), \mathrm{SP}(g_2, p)\rangle$.

Proof. We have the following equation (which is an easy exercise).

$$\begin{aligned}\mathrm{lc}(p)^* \frac{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p))}{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2))} &\mathrm{SP}(g_1, g_2) \\ + \mathrm{lc}(g_2)^* \frac{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p))}{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(p))} &\mathrm{SP}(g_1, p) \\ + \mathrm{lc}(g_1)^* \frac{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p))}{\mathsf{LCM}(\mathrm{lpp}(g_2), \mathrm{lpp}(p)} &\mathrm{SP}(g_2, p) = 0.\end{aligned}$$

Since $\mathrm{lpp}(p)|\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2))$, we have

$$\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p)) = \mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2)).$$

By the above equations, we have

$$\begin{aligned}\mathrm{lc}(p)^* \, \mathrm{SP}(g_1, g_2) + \mathrm{lc}(g_2)^* \frac{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p))}{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(p))} &\mathrm{SP}(g_1, p) \\ + \mathrm{lc}(g_1)^* \frac{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p))}{\mathsf{LCM}(\mathrm{lpp}(g_2), \mathrm{lpp}(p)} &\mathrm{SP}(g_2, p) = 0. \qquad (*)\end{aligned}$$

By the definition of S-polynomial and the assumption 2, we have

$$\mathrm{lc}(p)^* \, \mathrm{SP}(g_1, g_2) = \mathrm{SP}(g_1, g_2).$$

Hence, the equation $(*)$ can be transformed as follows :

$$\begin{aligned}\mathrm{SP}(g_1, g_2) = &- \mathrm{lc}(g_2)^* \frac{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p))}{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(p))} \mathrm{SP}(g_1, p) \\ &- \mathrm{lc}(g_1)^* \frac{\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2), \mathrm{lpp}(p))}{\mathsf{LCM}(\mathrm{lpp}(g_2), \mathrm{lpp}(p)} \mathrm{SP}(g_2, p) \\ &\in \langle \mathrm{SP}(g_1, p), \mathrm{SP}(g_2, p)\rangle.\end{aligned}$$

□

We have the next corollary which directly follows from theorem 6.3.2

Corollary 6.3.3. Let g_1, p, g_2 and p_i be polynomials in $R[\bar{X}]$ for each $i = 1, 2, \ldots l$ such that the following holds:

1. $\mathrm{lpp}(p_i)|\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2))$ for each $i = 1, 2, \ldots l$, and
2. $(\mathrm{lc}(p_1)^* \vee \mathrm{lc}(p_2)^* \vee \ldots \vee \mathrm{lc}(p_l)^*) \, \mathrm{lc}(g_1)^* \, \mathrm{lc}(g_2)^* = \mathrm{lc}(g_1)^* \, \mathrm{lc}(g_2)^*$.

Then, $\mathrm{SP}(g_1, g_2)$ is generated by an ideal $\langle \mathrm{SP}(g_1, p_1), \ldots, \mathrm{SP}(g_1, g_l), \mathrm{SP}(g_2, p_1), \ldots, \mathrm{SP}(g_2, p_l)\rangle$ in $R[\bar{X}]$. Note that notation $\vee$ is sum of boolean algebra. i.e. $a \vee b = a + b - ab$.

The next algorithm is required by Algorithm 6.3.5 ImprovedGB, and contains two criteria above. The following algorithm removes unnecessary critical pairs in $R[\bar{X}]$. The foundation algorithm of Algorithm 6.3.4 is UPDATE of [BW93](pp.230). The original algorithm UPDATE [BW93] is improved for the polynomial ring $R[\bar{X}]$ by the two criteria. The new algorithm which is the following, is called UPDATE, again. The termination argument follows the original UPDATE.

Algorithm 6.3.4. UPDATE$(G_{old}, B_{old}, h, \succ)$ (Update of a set of critical pairs and basis)

Input: G_{old}: a finite subset in $R[\bar{X}]$,
B_{old}: a finite set of critical pairs in $R[\bar{X}]$,
h : a non-zero polynomial $\in R[\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output: G_{new}: updates of G_{old}, B_{new}: update of B_{old}.
begin
$C \longleftarrow \{\{h,g\} | g \in G_{old}\}$; $D \longleftarrow \emptyset$; $E \longleftarrow \emptyset$
while $C \neq \emptyset$ **do**
Select $\{h,g\}$ from C
  if $\mathrm{lc}(h)^* \mathrm{lc}(g)^* = 0$ **then**
    $C \leftarrow C \backslash \{\{h,g\}\}$
  else
    $E \leftarrow E \cup \{\{h,g\}\}$; $C \leftarrow C \backslash \{\{h,g\}\}$
  end-if
end-while
while $E \neq \emptyset$ **do**
Select $\{h,g_1\}$ from C; $E \leftarrow E \backslash \{\{h,g_1\}\}$
  if $\mathrm{lpp}(h)$ and $\mathrm{lpp}(g_1)$ are disjoint **or**
  $\big(\mathsf{LCM}(\mathrm{lpp}(h), \mathrm{lpp}(g_2)) \nmid \mathsf{LCM}(\mathrm{lpp}(h), \mathrm{lpp}(g_1)) \forall \{h, g_2\} \in E$ **and**
      $\mathsf{LCM}(\mathrm{lpp}(h), \mathrm{lpp}(g_2)) \nmid \mathsf{LCM}(\mathrm{lpp}(h), \mathrm{lpp}(g_1)) \forall \{h, g_2\} \in D\big)$ **then**
  $D \leftarrow D \cup \{\{h,g_1\}\}$
  end-if
end-while
$F \longleftarrow \emptyset$
while $D \neq \emptyset$ **do**
select $\{h,g\}$ from D ; $D \leftarrow D \backslash \{\{h,g\}\}$
  if $\mathrm{lpp}(h)$ and $\mathrm{lpp}(g)$ are not disjoint **then**
  $F \leftarrow F \cup \{\{h,g\}\}$
  end-if
end-while
$B_{new} \leftarrow \emptyset$
while $B_{old} \neq \emptyset$ **do**
Select $\{g_1, g_2\}$ from B_{old}; $B_{old} \leftarrow B_{old} \backslash \{\{g_1, g_2\}\}$
  if $\mathrm{lpp}(h) \nmid \mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2))$ **or** $\mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(h)) = \mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2))$
    or $\mathsf{LCM}(\mathrm{lpp}(h), \mathrm{lpp}(g_2)) = \mathsf{LCM}(\mathrm{lpp}(g_1), \mathrm{lpp}(g_2))$ **then**
  $B_{new} \leftarrow B_{new} \cup \{\{g_1, g_2\}\}$
  elif $\mathrm{lc}(h)^* \mathrm{lc}(g_1)^* \mathrm{lc}(g_2)^* \neq \mathrm{lc}(g_1)^* \mathrm{lc}(g_2)^*$ **then**
  $B_{new} \leftarrow B_{new} \cup \{\{g_1, g_2\}\}$
  end-elif
  end-if
$B_{new} \leftarrow B_{new} \cup F$; $G_{new} \leftarrow \emptyset$
  while $G_{old} \neq \emptyset$ **do**
  select g from G_{old}; $G_{old} \leftarrow G_{old} \backslash \{g\}$
    if $\mathrm{lpp}(h) \nmid \mathrm{lpp}(g)$ **then**
      $G_{new} \leftarrow G_{new} \cup \{g\}$
    end-if
  end-while
$G_{new} \leftarrow G_{new} \cup \{h\}$

end-while
return(G_{new}, B_{new})
end

Finally, we construct an algorithm which is more efficient than Algorithm 6.2.14. Let F be a finite subset of $R[\bar{X}]$ with a order and $\succ$ a term order on $\mathrm{pp}(\bar{X})$. Then, the following algorithm outputs a reduced Gröbner basis G in $R[\bar{X}]$ such that $\langle F\rangle = \langle G\rangle$, and eliminates superfluous critical pairs according to the first and second criterion.

Algorithm 6.3.5. ImprovedGB$(F, \succ)$

Input: F : a finite list of polynomials in $R[\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output: G : Gröbner bases of F in $R[\bar{X}]$ with $\langle F\rangle = \langle G\rangle$.
begin
$L \leftarrow \mathrm{BC}(F, \succ)$
$G \leftarrow \emptyset$
$B \leftarrow \emptyset$
while $L \neq \emptyset$ **do**
Select g from L
$L \leftarrow L \backslash \{g\}$
$(G, B) \leftarrow \mathrm{UPDATE}(G, B, g, \succ)$
end-while
while $B \neq \emptyset$ **do**
Select $\{g_1, g_2\}$ from B
$B \leftarrow L \backslash \{\{g_1, g_2\}\}$
$h \leftarrow \mathrm{SP}(g_1, g_2)\downarrow_G$
  if $h \neq 0$ **then**
  $H \leftarrow \mathrm{BC}(\{h\}, \succ)$
    while $H \neq \emptyset$ **do**
    Select h_1 from H
    $H \leftarrow H \backslash \{h_1\}$
    $(G, B) \leftarrow \mathrm{UPDATE}(G, B, h_1, \succ)$
    end-while
  end-if **end-while**
return(G)
end

6.4 Alternative comprehensive Gröbner bases

Here we describe alternative comprehensive Gröbner bases (ACGB). Alternative comprehensive Gröbner bases are based on the theory of polynomial rings over commutative von Neumann regular rings. The idea is the following.

In section 6.4, 6.5 and 6.6, we assume that K **is always an infinite field**. Let $f_1(\bar{A}, \bar{X}), \ldots, f_k(\bar{A}, \bar{X})$ be polynomials in $K[\bar{A}, \bar{X}]$ with parameters $\bar{A} = \{A_1, \ldots, A_m\}$ and variables $\bar{X} = \{X_1, \ldots, X_n\}$. Consider each polynomial $f(\bar{A})$ in $K[\bar{A}]$ as function from K^m to K, then $f_1(\bar{A}, \bar{X}), \ldots, f_k(\bar{A}, \bar{X})$ become polynomials in the polynomial ring $K^{K^m}[\bar{X}]$ over a von Neumann regular ring K^{K^m}.

This idea leads us to define an alternative Comprehensive Gröbner basis (ACGB).

Definition 6.4.1 ([SS03]). Let F be a finite set of polynomials in a polynomial ring $K[\bar{A}, \bar{X}]$ over K with variables $\bar{A} = \{A_1, \ldots, A_m\}$ and $\bar{X} = \{X_1, \ldots, X_n\}$. Let G be a Gröbner basis of $\langle F \rangle$ in a polynomial ring $\bar{K}^{\bar{K}^m}[\bar{X}]$. G is called an **alternative comprehensive Gröbner basis (ACGB)** of F with parameters $\bar{A}$ in $\bar{K}[\bar{X}]$. ($\bar{K}$ is an algebraic closure of K.)

Remark: In order to enable the above Gröbner bases computation, it suffices to establish a way to handle the smallest commutative von Neumann regular ring extending the canonical image of $K[\bar{A}]$. If the rational field $K(\bar{A})$ would correspond to it, the situation would be very nice. Unfortunately, it does not work. Consider the inverse ${A_1}^{-1}$ of A_1 in the commutative von Neumann regular ring K^{K^m}. Since $A_1(a_1, \ldots, a_m) = a_1$ for any $a_1, \ldots, a_m \in K$, ${A_1}^{-1}$ should be the function ϕ from $K^m \to K$ such that $\phi(0, a_2, \ldots, a_m) = 0$ and $\phi(a_1, \ldots, a_m) = \frac{1}{a_1}$ if $a_1 \neq 0$. Certainly ϕ is not a member of $K(\bar{A})$.

Example 6.4.2 ([SS03]). Let t be a function $\mathbb{C}^2$ to $\mathbb{C}$ defined by

$$t(a,b) = \begin{cases} a-b, & \text{if } (a,b) \in \mathbb{C}^2 \setminus \mathbb{V}(a-b), \\ 0, & \text{otherwise.} \end{cases}$$

The inverse is

$$t(a,b) = \begin{cases} \frac{1}{a-b}, & \text{if } (a,b) \in \mathbb{C}^2 \setminus \mathbb{V}(a-b), \\ 0, & \text{otherwise.} \end{cases}$$

The addition of t and t^{-1} is

$$(t+t^{-1})(a,b) = \begin{cases} \frac{a^2-2ab+b^2+1}{a-b}, & \text{if } (a,b) \in \mathbb{C}^2 \setminus \mathbb{V}(a-b), \\ 0, & \text{otherwise.} \end{cases}$$

The multiplication

$$(t \cdot t^{-1})(a,b) = \begin{cases} 1, & \text{if } (a,b) \in \mathbb{C}^2 \setminus \mathbb{V}(a-b), \\ 0, & \text{otherwise.} \end{cases}$$

Actually, we would like to apply Theorem 6.2.19 for constructing an algorithm for computing ACGB in $\bar{K}^{\bar{K}^m}[\bar{X}]$. What is a prime ideal in $B(\bar{K}^{\bar{K}^m})$?

Proposition 6.4.3. The form of all prime ideals is the following:

$$\forall \alpha \in \bar{K}^m, T_\alpha := \left\{ t \in \bar{K}^{\bar{K}^m} \,\middle|\, t(\alpha) = 0, t(\beta) \in \{0,1\}, \forall \beta \in K^m \setminus \{\alpha\} \right\} \quad (*).$$

Proof. **(Ideal)** First, we prove that T_α is an ideal in $B(\bar{K}^{\bar{K}^m})$. It is obviously $0 \in T_\alpha$. Take f, g from T_α, then $f(\alpha) = 0$ and $g(\alpha) = 0$. Hence, we have $(f \vee g)(\alpha) = (f+g-fg)(\alpha) = f(\alpha) + g(\alpha) - f(\alpha)g(\alpha) = 0$. This fact implies $f \vee g \in T_\alpha$. (Note that K is a infinite field, and the function $K^m \to K$ is a injection.) Take f from T_α and $h \in B(\bar{K}^{\bar{K}^m})$. We have $(h \wedge f)(\alpha) = 0$ since $f(\alpha) = 0$. Therefore, T_α is an ideal in $B(\bar{K}^{\bar{K}^m})$.
(Prime) We prove that T_α is a prime ideal. Take $f \wedge g \in T_\alpha$, then $(f \wedge g)(\alpha) = f(\alpha) \wedge g(\alpha) = 0$. Therefore, $f(\alpha) = 0$ or $g(\alpha) = 0$, this means $f \in T_\alpha$ or $g \in T_\alpha$. T_α is a prime ideal.
Next we prove that all prime ideals of $B(\bar{K}^{\bar{K}^m})$ has the form $(*)$. This means that any

function q from T_α is always zero at only **one** point α. This prove is essentially same as Example 6.1.8. Let's consider the following set;
$\forall \alpha_1, \ldots, \alpha_i \in K^m$ with $\alpha_j \neq \alpha_l$, $j, l \in \{1, \ldots, i\}$ and $j \neq l$,

$$T_{(\alpha_1,\ldots,\alpha_i)} := \left\{ t \in \bar{K}^{\bar{K}^m} \,\middle|\, t(\alpha_1) = 0, \ldots, t(\alpha_i) = 0, t(\beta) \in \{0,1\}, \forall \beta \in \bar{K}^m \backslash \{\alpha_1, \ldots, \alpha_i\} \right\}.$$

That is, any function q from T_i is always zero at only i points.
Let $i = 2$. Take $f \wedge g \in T_2$, then $f(\alpha_1) \wedge g(\alpha_1) = 0$ and $f(\alpha_2) \wedge g(\alpha_2) = 0$. Let us consider the following function;

$$F := \left\{ h \in \bar{K}^{\bar{K}^m} \,\middle|\, h(\alpha_1) = 0, h(\alpha_2) = 1, t(\beta) \in \{0,1\}, \forall \beta \in \bar{K}^m \backslash \{\alpha_1, \alpha_2\} \right\},$$

and

$$G := \left\{ h \in \bar{K}^{\bar{K}^m} \,\middle|\, h(\alpha_1) = 1, h(\alpha_2) = 0, t(\beta) \in \{0,1\}, \forall \beta \in \bar{K}^m \backslash \{\alpha_1, \alpha_2\} \right\}.$$

Take $f_1 \in F$ and $g_1 \in G$. Then $f_1 \wedge g_1 \in T_2$, but $f_1, g_1 \notin T_2$. Hence, $T_{(\alpha_1,\alpha_2)}$ is not a prime ideal. Even if $i > 2$, $T_{(\alpha_1,\ldots,\alpha_i)}$ is not a prime ideal in $B(\bar{K}^{\bar{K}^m})$. This proof is the same as the case $i = 2$. Therefore, the form T_α is only the prime ideal in $B(\bar{K}^{\bar{K}^m})$. □

We define a computable ring T and operations on T which witness that T forms a von Neumann regular ring. For an arbitrary polynomial $f \in K[\bar{A}]$, we can consider it as a mapping $f : \bar{K}^m \to \bar{K}$, i.e., $f \in \bar{K}^{\bar{K}^m}$. Therefore, we can define the canonical embedding

$$\varphi : K[\bar{A}] \to \bar{K}^{\bar{K}^m}.$$

Let T be the closure of the image $\varphi(K[\bar{A}])$ under addition, multiplication, and inverse in the von Neumann regular ring $\bar{K}^{\bar{K}^m}$, thus T becomes a von Neumann regular ring.
Let's define the following map

$$\begin{aligned} \mathrm{ter}_T : K[\bar{A}][\bar{X}] &\to T[\bar{X}], \\ c_1\alpha_1 + \cdots + c_l\alpha_l &\mapsto \mathrm{ter}_T(c_1)\alpha_1 + \cdots + \mathrm{ter}_T(c_l)\alpha_l, \end{aligned}$$

where $c_1, \ldots, c_l \in K[\bar{A}]$ and $\alpha_1, \ldots, \alpha_l \in \mathrm{pp}(\bar{X})$.

We know the form of all prime ideals. Therefore, if we have $B(T)$ in Theorem 6.2.19, then the theorem means the following.

Theorem 6.4.4. Let $F = \{f_1(\bar{A}, \bar{X}), \ldots, f_s(\bar{A}, \bar{X})\}$ be a set of polynomials in $K[\bar{A}][\bar{X}]$ (where $\bar{A}$ are parameters and $\bar{X}$ are variables). Furthermore, let $G = \{g_1, \ldots, g_l\}$ be the reduced Gröbner basis of $\mathrm{ter}_T(F)$ in $T[\bar{X}]$. Then, for each m-tuple $\bar{a} = (a_1, \ldots, a_m) \in \bar{K}^m$, $G_{\bar{a}}$ becomes the reduced Gröbner basis of the ideal $\langle f_1(\bar{a}, \bar{X}), \ldots, f_s(\bar{a}, \bar{X})\rangle$ in $\bar{K}[\bar{X}]$. Here $G_{\bar{a}}$ denotes the set $\{g_{1\bar{a}}, \ldots, g_{l\bar{a}}\}$ of polynomials $g_{1\bar{a}}, \ldots, g_{l\bar{a}}$ in $\bar{K}[\bar{X}]$ given from $g_1, \ldots, g_l$ by replacing each coefficient c with $c(\bar{a})$. (Remember that c is an element of T).

By the above theorem, the Gröbner bases satisfy the main property of comprehensive Gröbner bases. Therefore, they are called "alternative comprehensive Gröbner bases".
In fact, we need to define the algebraic structure of T to compute a Gröbner bases in $T[\bar{X}]$. That is, we need addition, multiplication, and inverse in the von Neumann regular ring T. In [SS02, SS03], they are introduced and defined. In this thesis, we do not describe them. If one is interested in the detail, the author strongly recommends to see [SS03]. In the algebraic structure of T, the following definitions are required for computing Gröbner

bases in $T[\bar{X}]$. In the next section, the notations of the following definition will be applied for describing the special types of comprehensive Gröbner bases. Therefore, we give the following two definitions.

Definition 6.4.5 (preterrace [SS03]). A triple (s, t, r) is called a preterrace on $K[\bar{A}]$ if s and t are finite sets of polynomials in $K[\bar{A}]$ and $r = \frac{g}{h}$ for some $g, h \in K[\bar{A}]$ which satisfy

1. $\mathbb{V}(s) \subseteq \mathbb{V}(t)$,
2. $(\mathbb{V}(\{g\}) \cup \mathbb{V}(\{h\})) \cap (\mathbb{V}(t) \setminus \mathbb{V}(s)) = \emptyset$, i.e., $g(\bar{a}) \neq 0$ and $h(\bar{a}) \neq 0$ for any $\bar{a} \in \mathbb{V}(t) \setminus \mathbb{V}(s)$.

For a given preterrace $p = (s, t, r)$, the **support** of p (written: $\mathrm{supp}(p)$) is the set of $\mathbb{V}(t) \setminus \mathbb{V}(s) \subseteq K^m$. For a preterrace $p = (s, t, \frac{g}{h})$ on $K[\bar{A}]$ and $\bar{a} \in K^m$, we define $p(\bar{a}) \in K$ by

$$p(\bar{a}) = \begin{cases} \frac{g(\bar{a})}{h(\bar{a})}, & \text{if } \bar{a} \in \mathrm{supp}(p) = \mathbb{V}(t) \setminus \mathbb{V}(s), \\ 0, & \text{otherwise.} \end{cases}$$

Therefore, p can be consider as a member of T.

Definition 6.4.6 (terrace [SS03]). A finite set $\{p_1, \ldots, p_l\}$ is called a **terrace** on $K[\bar{A}]$ of each p_i (i=1,...,l)is a preterrace on $K[\bar{A}]$ such that $\succ (p_i) \neq \emptyset$ and $\mathrm{supp}(p_i) \cap \mathrm{supp}(p_j) = \emptyset$ for any distinct $i, j \in \{1, \ldots, l\}$. The support of a terrace t is defined by

$$\mathrm{supp}(t) = \bigcup_{p \in t} \mathrm{supp}(p) \subseteq K^m.$$

Example 6.4.7. Let $f = abx^2y + x + by$, $g = y^2 + ax + b$ be polynomials in $\mathbb{C}[a, b][x, y]$. We consider the map $\mathrm{ter}_t : \mathbb{C}[a, b][x, y] \to T_{(a,b)}[x, y]$ where $T_{(a,b)}$ is the von Neumann regular ring of equivalence class on terrace on $\mathbb{C}[a, b]$. Then,

$$\begin{aligned} \mathrm{ter}_T(f) &= [(\mathbb{C}^2 - \mathbb{V}(ab), ab)]x^2y + [(\mathbb{C}^2, 1)]x + [(\mathbb{C}^2 - \mathbb{V}(b), b)]y, \\ \mathrm{ter}_T(g) &= [(\mathbb{C}^2, 1)]y^2 + [(\mathbb{C}^2 - \mathbb{V}(a), a)]x + [(\mathbb{C}^2 - \mathbb{V}(b), b)]1. \end{aligned}$$

The one of coefficients $[(\mathbb{C}^2 - \mathbb{V}(ab), ab)]$ means

$$\begin{cases} ab, & \text{if } ab \neq 0, \\ 0, & \text{otherwise.} \end{cases}$$

One can notice that every coefficients has the parametric spaces and elements of $\mathbb{C}[a, b]$. That is, every coefficients has preterraces (see Definition 6.4.5).

Now, since we know Algorithm 6.3.5, Theorem 6.4.4 and the structure of $T[\bar{X}]$ (which is from [SS03]), we can construct an algorithm for computing alternative comprehensive Gröbner bases.

Algorithm 6.4.8. ACGB$(F, \succ)$ (Alternative Comprehensive Gröbner Bases)

Input F : a subset of $K[\bar{A}][\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output G : an alternative comprehensive Gröbner basis for $\langle F \rangle$ with respect to $\succ$.

1. Compute $\mathrm{ter}_T(F)$.
2. Compute a Gröbner basis G for $\langle \mathrm{ter}_T(F) \rangle$ with respect to $\succ$ in $T[\bar{X}]$ by the Algorithm 6.3.5 where T is a commutative von Neumann regular ring.

3. G is an alternative comprehensive Gröbner basis for $\langle F \rangle$ with respect to $\succ$.

In Figure 6.1, we give the rough procedure of Algorithm 6.4.8.

Alternative comprehensive Gröbner bases have the following nice property, which do not hold in standard comprehensive Gröbner bases [SS03].

"There exists a canonical form of an alternative comprehensive Gröbner basis in a natural way."

Since an alternative comprehensive Gröbner basis is already in a form of a Gröbner basis in a polynomial ring over a commutative von Neumann regular ring, we can use a stratified Gröbner basis as a canonical form an alternative comprehensive Gröbner basis. By the same reason above, we can use reductions of an alternative comprehensive Gröbner basis.

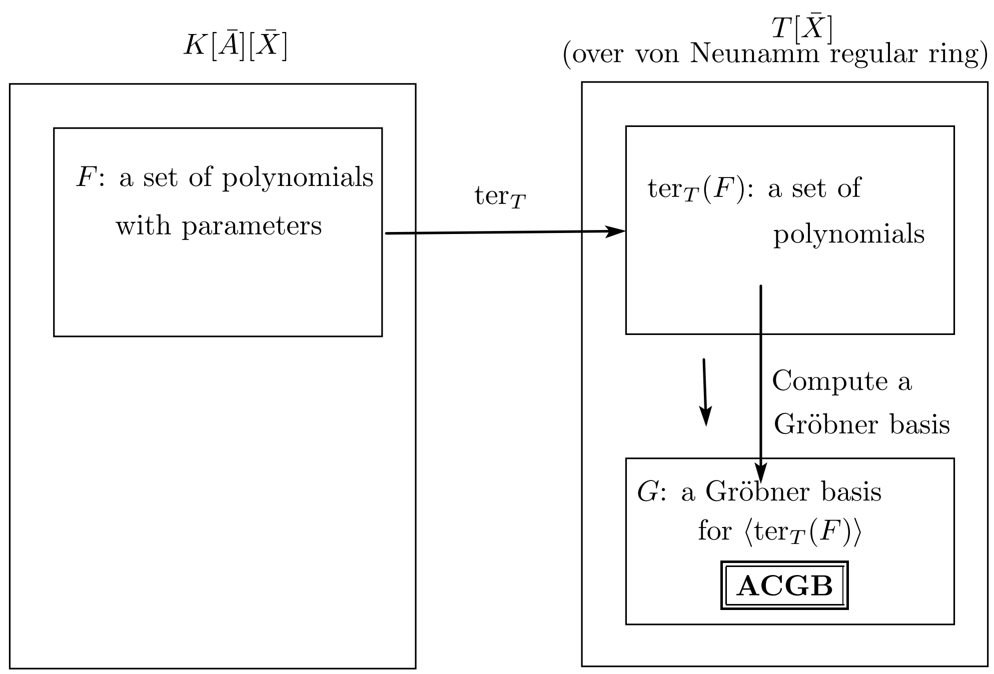

Figure 6.1 A computation method for ACGB

The algorithm ACGB has been implemented in the computer algebra systems Risa/Asir by Suzuki in [SS03]. In the following example, we give the outputs of the program.

Example 6.4.9. Let $F = \{bx^2y+3, axy^2+bxy+b\}$ be a set of polynomials in $\mathbb{C}[a,b][x,y]$, a, b parameters, x, y variables and $\succ$ the graded reverse lexicographic order such that $x \succ y$. The Suzuki's program outputs the following as an alternative comprehensive Gröbner basis for $\langle F \rangle$ with respect to $\succ$;

```
[[(V[b],1)]*1,
[(V[b*a]-V[b],1)]*y+[(V[b*a]-V[b],1/3*b)]*1,
[(V[0]-V[b*a],1),(V[b*a]-V[b],1)]*x+[(V[0]-V[b*a],(-3*a)/(b^2))]*y+[(V[0]
```

```
-V[b*a],(-3)/(b)),(V[b*a]-V[b],(-3)/(b))]*1,
[(V[0]-V[b*a],1)]*y^3+[(V[0]-V[b*a],(2*b)/(a))]*y^2+[(V[0]-V[b*a],(b^2)/(
a^2))]*y+[(V[0]-V[b*a],(1/3*b^3)/(a^2))]*1].
```

We can understand the output as follows;

$$\begin{cases} \{1\}, & \text{if } b = 0, \\ \{y + \frac{1}{3}b, x - \frac{1}{b}\}, & \text{if } ab = 0, b \neq 0, \\ \{x + y - \frac{3}{b}, y^3 + \frac{2b}{a}y^2 + \frac{b^2}{a^2}y + \frac{b^3}{3a^2}\}, & \text{if } ab \neq 0. \end{cases}$$

6.5 ACGB on varieties (ACGB-V)

In this section, we present a special type of ACGB which is called ACGB-V (ACGB on Varieties). When there exists a constraint of parameters $\bar{A}$ in a form of polynomial equations $f_1(\bar{A}) = 0, \ldots, f_l(\bar{A}) = 0$, it is more natural to consider the range of values for $\bar{A}$ to be the variety $\mathbb{V}(f_1(\bar{A}), \ldots, f_l(\bar{A}))$ than a whole space K^m. One of the main ideas of ACGB is that we consider a polynomial in $\bar{A}$ as a function from K^m to K, i.e., as a member of K^{K^m} that is a commutative von Neumann regular ring, and then treat it as a member of the regular closure of $K[\bar{A}]$ in K^{K^m}. When such constrains exists, we can replace K^{K^m} by $K^{\mathbb{V}(f_1(\bar{A}),\ldots,f_l(\bar{A}))}$. Note that the restriction of $K[\bar{A}]$ on $K^{\mathbb{V}(f_1(\bar{A}),\ldots,f_l(\bar{A}))}$ is isomorphic to a quotient ring $K[\bar{A}]/I(\mathbb{V}(f_1, \ldots, f_l))$, where $I(\mathbb{V}(f_1(\bar{A}), \ldots, f_l(\bar{A})))$ denotes an ideal of $K[\bar{A}]$ that consists of all polynomials vanishing at every point of $\mathbb{V}(f_1(\bar{A}), \ldots, \mathbb{V}(\bar{A}))$. Hence, it is isomorphic to $K[\bar{A}]/\operatorname{rad}(\langle f_1(\bar{A}), \ldots, f_l(\bar{A})\rangle)$ in case K is an algebraically closed field. (Here, $\operatorname{rad}(I)$ denotes a radical ideal of I.) The above observation leads us to the following definition.
The basic notion of this section has been studied by the author [SSN03b, SSN03a, Nab05a].

Definition 6.5.1 (ACGB-V). Let K be an algebraically closed field, F a set of polynomials in $K[\bar{A}][\bar{X}]$ and I a polynomial ideal in $K[\bar{A}]$. An **ACGB-V** (**A**lternative **C**omprehensive **G**röbner **B**asis on a **V**ariety) of $\langle F\rangle$ with respect to I is defined as follows. Let T be a regular closure of the quotient ring $K[\bar{A}]/\operatorname{rad}(I)$ in the commutative von Neumann regular ring $K^{\mathbb{V}(I)}$. Then, there exists a stratified Gröbner basis of $\langle F\rangle$ in $T[\bar{X}]$. W call G an ACGB-V of $\langle F\rangle$ with respect to the ideal I.

Theorem 6.5.2. Using the same notation as in the above definition, let $F = \{f_1(\bar{A}, \bar{X}), \ldots, f_l(\bar{A}, \bar{X})\}$ and $G = \{g_1(\bar{X}), \ldots, g_k(\bar{X})\}$ an ACGB-V of $\langle F\rangle$ with respect to I of $K[\bar{A}]$. Then following properties hold for any m-tuple $\bar{a} \in K^m$ belonging to the variety $\mathbb{V}(I)$:

1. $G_{\bar{a}} = \{g_{1\bar{a}}(\bar{X}), \ldots, g_{k\bar{a}}(\bar{X})\}\backslash\{0\}$ is a reduced Gröbner basis of the ideal generated by $F(\bar{a}) = \{f_1(\bar{a}, \bar{X}), \ldots, f_k(\bar{a}, \bar{X})\}$ in $K[\bar{X}]$.
2. For any polynomial $h(\bar{X}) \in T[\bar{X}]$, we have $(h_{\downarrow_G})_{\bar{a}}(\bar{X}) = h_{\bar{a}}(\bar{X}) \downarrow_{G_{\bar{a}}(\bar{X})}$.

(Note that $\bar{a}$ is a prime ideal in $B(K^{K^m})$. For $p \in \operatorname{Spec}(B(R))$ we let $\Phi_p : R \to R_p$ be the canonical homomorphism where R is a commutative von Neumann regular ring. Then, $G_p := \Phi(G)$. That is, $G_{\bar{a}} = \Phi(G)$.)

Proof. This proof is exactly same as the proof of Theorem 3.2 of [SS02] or Theorem 4.3 [SS03]. □

Example 6.5.3. Let F be the set of polynomials $\{a - b, axy - bx^3y - 3a, bxy - 3bx - 5b\}$ in $\mathbb{Q}[a, b][x, y]$, a, b parameters and x, y variables. Take a lexicographic order $\succ$ such that $y \succ x$. When we are interested in only values such that the ideal becomes proper, it is more natural to construct an ACGB-V of $\langle F\rangle$ with respect to the ideal $\langle a - b\rangle$.

Since $\langle a-b\rangle$ is already a radical ideal, we construct a stratified Gröbner basis G of $\{a-b, axy-bx^3y-3a, bxy-3bx-5b\}$ in $T[x,y]$ where T is a regular closure of $\mathbb{Q}[a,b]/\langle a-b\rangle$. This G is the desired $ACGB-V$ of $\langle F\rangle$ and has the following form using terraces:

$$\begin{aligned}G=\{&[(\mathbb{V}(a-b)-\mathbb{V}(a-b,a),1)]y+[(\mathbb{V}(a-b)-\mathbb{V}(a-b,a),\tfrac{-15}{2})]x\\&\qquad\qquad\qquad\qquad+[(\mathbb{V}(a-b)-\mathbb{V}(a-b,a),-8)],\\&[(\mathbb{V}(a-b)-\mathbb{V}(a-b,a),1)]x^2+[(\mathbb{V}(a-b)-\mathbb{V}(a-b,a),\tfrac{2}{3})]x\\&\qquad\qquad\qquad\qquad+[(\mathbb{V}(a-b)-\mathbb{V}(a-b,a),\tfrac{-2}{3})]\}.\end{aligned}$$

We should note that the ACGB-V of $\langle (a-b)+(axy-bx^2y-3a)^3+(bxy-3bx-5b)^4, axy-bx^2y-3a, bxy-3bx-5b\rangle$ with respect to $\langle a-b\rangle$ has the same form.

6.6 Computation methods for ACGB-V

In this section, we present algorithms for computing ACGB-V. As we saw, when there is a constraint of parameters $\bar{A}$ in a form of polynomial equations $f_1(\bar{A})=0,\dots,f_l(\bar{A})=0$, it is more natural to consider the range of values for $\bar{A}$ to be the variety $\mathbb{V}(f_1(\bar{A}),\dots,f_l(\bar{A}))$ than a whole space K^m. First, we describe the case $\langle f_1(\bar{A}),\dots,f_l(\bar{A})\rangle$ is a zero-dimensional ideal in $K[\bar{A}]$. Second, we generalize the method of zero-dimensional case to general cases. These computation method is introduced by the author in [SSN03b, SSN03a, Nab05a].

Definition 6.6.1 (Definition 6.46 [BW93]). Let I be a proper ideal of $K[\bar{A}]$ and $\bar{U}\subseteq\bar{A}$. Then $\bar{U}$ is called independent modulo I if $I_{\bar{U}}=I\cap K[\bar{U}]=\{0\}$. Moreover, $\bar{U}$ is called **maximally independent** modulo I if it is independent modulo I and not properly contained in any other independent set modulo I. The **dimension** $\dim(I)$ of I is defined as

$$\dim(I)=\left\{|\bar{U}|\,\middle|\,\bar{U}\subset\bar{A}\text{ independent modulo }I\right\}.$$

We will, rather obviously, call an ideal of $K[\bar{A}]$ **zero-dimensional** if it is proper and has dimension zero.

Definition 6.6.2 (DCGB). Let $F=\{f_1(\bar{A},\bar{X}),\dots,f_l(\bar{A},\bar{X})\}$ in $K[\bar{A}][\bar{X}]$ and S a set of polynomials $\{s_1(A_1),\dots,s_m(A_m)\}$, where $s_i(A_i)$ is a non-constant univariate polynomial in $K[A_i]$ for each $i=1,\dots,m$. A set $G=\{g_1(\bar{A},\bar{X}),\dots,g_k(\bar{A},\bar{X})\}$ of polynomials in $K[\bar{A}][\bar{X}]$ is called a **discrete comprehensive Gröbner basis** (DCGB) of $\langle F\rangle$ with respect to $(\bar{A},S)$ if it satisfies the following:
$G(\bar{a})=\{g_1(\bar{a},\bar{X}),\dots,g_k(\bar{a},\bar{X})\}\backslash\{0\}$ is a Gröbner basis for $\langle f_1(\bar{a},\bar{X}),\dots,f_l(\bar{a},\bar{X})\rangle$ in $\bar{K}[\bar{X}]$ for any elements $\bar{a}=(a_1,\dots,a_m)\in\bar{K}^m$ satisfying $s_1(a_1)=0,\dots,s_m(a_m)=0$.

Lemma 6.6.3. Let I be a zero dimensional radical ideal in a polynomial ring $K[\bar{A}]$. Then, $K[\bar{A}]/I$ becomes a commutative von Neumann regular ring.

Proof. Present I as an intersection of prime ideals $P_1,\dots,P_k$ of $K[\bar{A}]$. Since I is zero dimensional, each P_i is also zero-dimensional. Therefore, P_i is a maximal ideal, and thus, we can apply the Chinese remainder theorem to obtain an isomorphism $K[\bar{A}]/I\cong K[\bar{A}]/P_1\times\cdots\times K[\bar{A}]/P_k$. The right-hand side is a direct product of fields, hence it is a commutative von Neumann regular ring. □

Theorem 6.6.4 ([SSN03a]). Let I be a zero dimensional ideal in a polynomial ring $K[\bar{A}]$, $F=\{f_1(\bar{A},\bar{X}),\dots,f_l(\bar{A},\bar{X})\}$ a set of polynomials in $K[\bar{A}][\bar{X}]$ and $G=\{g_1(\bar{A},\bar{X}),\dots,g_k(\bar{A},\bar{X})\}$ a stratified Gröbner basis for $\langle F\rangle$ in a polynomial ring $(K[\bar{A}]/\operatorname{rad}(I))[\bar{X}]$ over a commutative von Neumann regular ring $K[\bar{A}]/\operatorname{rad}(I)$. Then, we have the following two properties for any m-tuple $\bar{a} = (a_1, \ldots, a_m) \in \bar{K}^m$ belonging to the variety $\mathbb{V}(I)$:

1. $G(\bar{a}) = \{g_1(\bar{a}, \bar{X}), \ldots, g_k(\bar{a}, \bar{X})\} \backslash \{0\}$ is a reduced Gröbner basis of the ideal generated by $F(\bar{a}) = \{f_1(\bar{a}, \bar{X}), \ldots, f_k(\bar{a}, \bar{X})\}$ in $K[\bar{X}]$.
2. For any polynomial $h(\bar{A}, \bar{X}) \in K[\bar{A}][\bar{X}]$, we have $(h(\bar{A}, \bar{X}){\downarrow_G})(\bar{a}, \bar{X}) = h(\bar{a}, \bar{X}) \downarrow_{G(\bar{a}, \bar{X})}$.

Like the algorithm ACGB, by Theorem 6.6.4 and Algorithm 6.3.5, we can easily construct an algorithm for computing discrete comprehensive Gröbner bases.

Algorithm 6.6.5. DCGB$(F, S, \succ)$ [SSN03a]

Input F : a subset of $K[\bar{A}][\bar{X}]$,
S: a zero-dimensional ideal in $K[\bar{A}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output G : a discrete comprehensive Gröbner basis for $\langle F \rangle$ with respect to S.

1. Consider the map $\psi : K[\bar{A}]/\operatorname{rad}(S) \to K^s$ where $s = |\mathbb{V}(\operatorname{rad}(S))|$ (the number of solutions).
2. Compute $\psi(F)$.
3. Compute a Gröbner basis G for $\langle \psi(F) \rangle$ with respect to $\succ$ in $(K[\bar{A}]/\operatorname{rad}(S)([\bar{X}]$ by the Algorithm 6.3.5.
4. Compute $\psi^{-1}(G)$. There exists a map ψ^{-1} because $K[\bar{A}]/\operatorname{rad}(S)$ is isomorphic to K^s.

In Figure 6.2, we give the rough procedure of Algorithm 6.6.5.

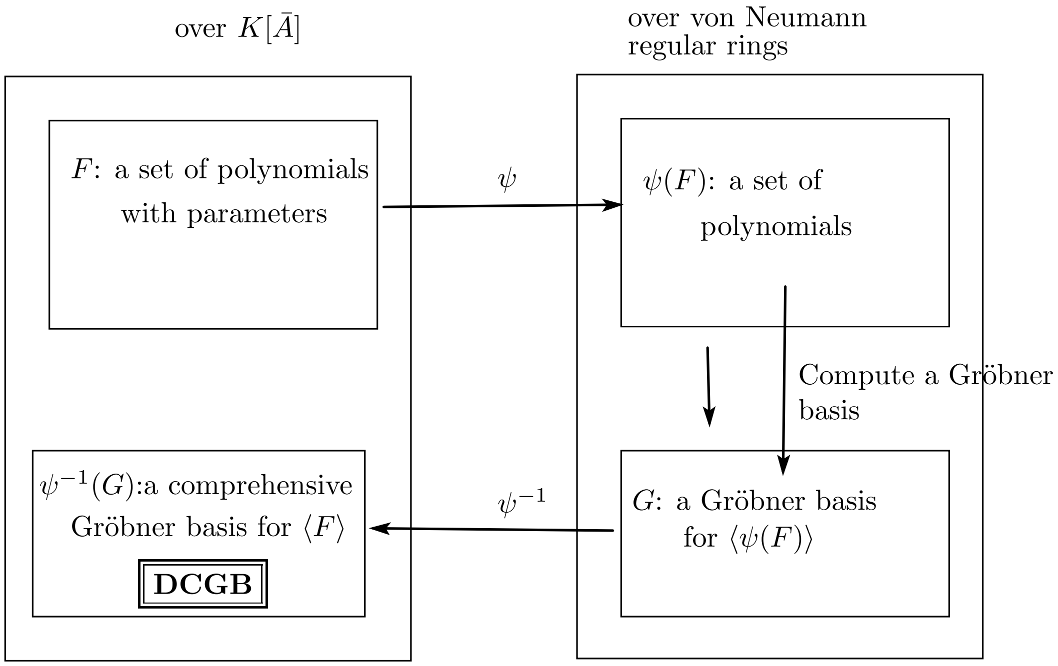

Figure 6.2 A computation method for DCGB

This algorithm has been implemented by the author in prolog[*1]. In the next example, we give an output of the program.

Example 6.6.6. Let $F = \{2xy^2z+x+2, aby^2z+2bx+9, bx+ayz+2, a^3+4a^2+3a, 2b^3-5b^2+3b\}$ be a set of polynomial in $\mathbb{C}[a,b][x,y,z]$, a,b parameters, x,y variables and $\succ$ the graded reverse lexicographic order. Then, $\langle a^3+4a^2+3a, 2b^3-5b^2+3b\rangle$ is a zero dimensional ideal in $\mathbb{C}[a,b]$. We can use the algorithm DCGB. The program outputs the following set G;

```
[(2/3*b^2-5/3*b+1),
(-2/9*a^2*b^2+5/9*a^2*b-4/9*a*b^2+10/9*a*b)*y+(32352520/138649329*a^2*b^2
-6228140/15405481*a^2*b+4717108/46216443*a*b^2-2362394/15405481*a*b)*z+(2
29518223/138649329*a^2*b^2-293241527/92432886*a^2*b+319292681/92432886*a*
b^2-394962473/61621924*a*b),
(-2/3*b^2+5/3*b)*x+(25960/104013*a^2*b^2-4860/11557*a^2*b+2692/34671*a*b^
2-1490/11557*a*b)*z+(758285/624078*a^2*b^2-2890075/1248156*a^2*b+785765/3
12039*a*b^2-3051715/624078*a*b-65/18*b^2+247/36*b),
(-2/9*a^2*b^2+5/9*a^2*b-4/9*a*b^2+10/9*a*b)*z^2+(41/24*a^2*b^2+23/6*a^2*b
+61/12*a*b^2+21/2*a*b)*z+(9001/1944*a^2*b^2-4201/1944*a^2*b+54007/3888*a*
b^2-24727/3888*a*b),
(2/9*a^2*b^2-5/9*a^2*b+4/9*a*b^2-10/9*a*b-2/3*b^2+5/3*b)*z*y^2+(1/9*a^2*b
^2-41/234*a^2*b+2/9*a*b^2-41/117*a*b-1/3*b^2+41/78*b)].
```

Therefore, G is a discrete comprehensive Gröbner basis. In fact, a comprehensive Gröbner basis for $\langle F\rangle$ is $G \cup \{a^3+4a^2+3a, 2b^3-5b^2+3b\}$. That is, we can understand

$$\begin{cases} G & \text{if the support is } \mathbb{V}(a^3+4a^2+3a, 2b^3-5b^2+3) \\ \{1\} & \text{if the support is } \mathbb{C}^2 \setminus \mathbb{V}(a^3+4a^2+3a, 2b^3-5b^2+3). \end{cases}$$

Second, we describe an algorithm for computing ACGB-V. Namely, we extend the algorithm DCGB to the more general situation of ACGB-V. In order to explain the algorithm we need the following lemma.

Lemma 6.6.7 (Lemma 7.47 [BW93]). Let $I \subset K[\bar{A}]$ be an ideal and $\bar{U} \subset \bar{A}$ be a maximal independent set of variables with respect to I. Then, $I \subset K(\bar{U})[\bar{A}\backslash\bar{U}]$ is a zero-dimensional ideal.

In order to compute a maximal independent modulo I, we can apply the algorithm DCGB for computing ACGB-V. However, we can not use the method directly for computing ACGB-V, because we have to regard $\bar{U}$ as a set of parameters.

For example, let $I = \langle ab+b+c+1\rangle$, $F = \{bx+ay+c, cx^2+ab\}$ where a,b,c are parameters and x,y are variables. Then I is not a zero-dimensional ideal in $\mathbb{Q}[a,b,c]$, but $I_{\{a,c\}}$ is a zero-dimensional ideal in $\mathbb{Q}(a,c)[b]$. However, it is only true if $a \neq -1$. The case $a=-1$ is overlooked, though if should not be. By $I = \langle ab+b+c+1\rangle$, there exists $a=-1$. To solve this problem, we propose the following algorithm. The first key point is to compute ACGB where $\bar{U}$ is a set of parameters and $\{\bar{A}\backslash U\}$ is a set of variables. The second key point is an information of **support** (or preterrace). In the following algorithm, we use the notation supp which is from Definition 6.4.5.

[*1] Prolog is a logic programming language. The author used "SICStus Prolog". `http://www.sics.se/isl/sicstuswww/site/index.html`

Algorithm 6.6.8. Div-zero(I)

Input I : a polynomial ideal in $K[\bar{A}]$,
Output $M = \{(V,Q)|Q$: a set of variables in $\bar{A}$; V: a set of polynomials in $K(Q)[\bar{A}\backslash Q]$ with $\mathrm{supp}(\mathrm{lpp}(h_1)) = \mathrm{supp}(\mathrm{lpp}(h_2))$, $\forall h_1, h_2 \in V$ $\}$.

1. $Z \leftarrow I$; $M \leftarrow \emptyset$;
2. $R \leftarrow$ Compute a radical ideal of Z in $K[\bar{A}]$ $(*1)$
3. $U \leftarrow$ Compute a maximal independent set with respect to R in $K[\bar{A}]$ $(*2)$
4. $A \leftarrow$ Compute ACGB with respect to Z (and the lexicographic order $\succ$) where U is a set of parameters and $\{\bar{A}\backslash U\}$ is a set of variables. (We can obtain a reduced stratified Gröbner basis in $\bar{K}^{\bar{K}^{|U|}}[\bar{A}\backslash U]$.)
5. $N \leftarrow$ Classify A into N which is a set of sets of polynomials in $\bar{K}^{\bar{K}^{|U|}}[\bar{A}\backslash U]$. $N = \{N_1|$ supports of all monomials p_1 are same, and $\mathrm{supp}(p_1) = \mathrm{supp}(p_2)$, $\forall p_1, p_2 \in N_1$ $\}$. That is, N_1 is a set of polynomials in $\bar{K}^{\bar{K}^{|U|}}[\bar{A}\backslash U]$ and all polynomials in N_1 have a same support.
   (We can compute this N by using *supports* (head idempotent). This algorithm is similar to the algorithm BC.)
6. **while** $N \neq \emptyset$ **do**
   Select J from N; $N \leftarrow N\backslash\{J\}$
   if $\langle J\rangle$ is a zero-dimensional ideal in $K(U)[\bar{A}\backslash U]$ **then**
   $M \leftarrow M \cup (J, U)$
   else
   $S \leftarrow$ Compute all combinations of polynomials of T_1 where supports of all element of J is $[\mathbb{V}(T_1) - \mathbb{V}(T_2)]$. $(**)$
   while $S \neq \emptyset$ **do**
   Select s_1 from s; $s \leftarrow s\backslash s_1$
   $Z \leftarrow \{s_1\} \cup Z$
   goto 2
   end-while
   end-if
   end-while

($(**)$ Let $[\mathbb{V}(T_1) - \mathbb{V}(T_2)] \subset \bar{K}^{\bar{K}^{|U|}}$ be a *support* of J where T_1, T_2 are sets of polynomials in $K[U]$. Then we can consider that J is restricted to T_1. So we consider about $Z\cup$ {one of T_1 } in the next step.)

Remark: In $(*1)$ and $(*2)$, there exist algorithms for computing a radical ideal and a maximal independent set U modulo I.

By the algorithm ACGB and the Remark, this algorithm clearly terminates and outputs correctly.

By the above algorithm we can obtain zero-dimensional ideals from I in several polynomial rings.
Let us consider $I = \langle ab + b + c + 1\rangle$ again. I is already a radical ideal and $\{a, c\}$ is a maximal independent set with respect to I in $\mathbb{Q}[a, b, c]$. We can compute ACGB of I where a, b are parameters and b is variable. Then, the algorithm ACGB outputs the following:

$[[(\mathbb{V}[(c+1)*(a+1)]-\mathbb{V}[c+1],1)]*1,$
$[(\mathbb{V}[0]-\mathbb{V}[a+1],1)]*b+[(\mathbb{V}[0]-\mathbb{V}[(c+1)*(a+1)],(c+1)/(a+1))]*1].\ (*)$

Look at the first polynomial. Then, we have $\{1\}$ when the support is $[[(\mathbb{V}((c+1)*(a+1))-\mathbb{V}(c+1)]$. So we don't need it. In the second polynomial, we classify the support $[\mathbb{V}(0)-\mathbb{V}(a+1)]$ to $[\mathbb{V}(0)-\mathbb{V}((a+1)(c+1))]$ and $[\mathbb{V}((a+1)(c+1))-\mathbb{V}(a+1)]$ where $\mathbb{V}(0)=\mathbb{C}^2$. First we consider the support $[\mathbb{V}(0)-\mathbb{V}((a+1)(c+1))]$. In the support $[\mathbb{V}(0)-\mathbb{V}((a+1)(c+1))]$, we have $\langle b+\frac{c+1}{a+1}\rangle$ which is a zero-dimensional ideal in $\mathbb{Q}(a,c)[b]$. Next by the algorithm, we have to consider three supports $v_1=[\mathbb{V}((a+1)(c+1))-\mathbb{V}(a+1)]$, $v_2=[\mathbb{V}((a+1)(c+1))-\mathbb{V}(c+1)]$ and $v_3=[\mathbb{V}(a+1,c+1)]$ (because in the support $[\mathbb{V}(0)-\mathbb{V}((a+1)(c+1))]$, I is restricted to $\langle (a+1)(c+1)\rangle$). Actually, now we have

$$\mathbb{V}((a+1)(c+1))=v_1\cup v_2\cup v_3.$$

(1) If we consider the support v_1, then we obtain $\langle b\rangle$ from $(*)$
(2) If we take the support v_2, then we already consider the case.
(3) If we take the support v_3, then we can obtain $\langle ab+b+c,a+1,c+1\rangle=\langle a+1,c+1\rangle$ which is a zero-dimensional ideal in $\mathbb{Q}(b)[a,c]$.

Therefore by the algorithm we obtained:

$$\begin{cases} \langle b+\frac{c+1}{a+1}\rangle, & \text{zero-dim. ideal in } \mathbb{Q}(a,c)[b],\ \text{supp. } [\mathbb{V}(0)-\mathbb{V}((a+1)(c+1)], \quad (1)\\ \langle b\rangle, & \text{zero-dim. ideal in } \mathbb{Q}(a,c)[b],\ \text{supp. } [\mathbb{V}((a+1)(c+1))-\mathbb{V}(a+1)], (2)\\ \langle a+1,c+1\rangle, & \text{zero-dim. ideal in } \mathbb{Q}(b)[a,c],\ \text{supp. } [\mathbb{V}(c+1,a+1)]. \quad (3)\end{cases}$$

We already know the method of DCGB (zero-dimensional case) [SSN03b, SSN03a]. By the algorithm Div-zero, we can obtain some zero-dimensional ideals, and we apply DCGB's method for computing ACGB-V. However, note that when we compute DCGB in several polynomial rings, we regard a maximal independent set as parameters. Because these zero-dimensional ideals have parameters which are in the coefficient domain $K(U)$ of their polynomial ring. For example, let $F=\{bx+ay+c,cx^2+ab\}$ and $\succ$ the lexicographic order such that $x\succ y$.

In case (3), we regard b as a parameter when we compute comprehensive Gröbner basis : (That is, $a+1=0,c+1=0$, the support is $[\mathbb{V}(a+1,c+1)]$)

```
[[(V[-b]-V[-1],1)]*y+[(V[-b]-V[-1],1)]*1,
[(V[0]-V[b],1)]*x+[(V[0]-V[-b],(-1)/(b))]*y+[(V[0]-V[-b],(-1)/(b))]*1,
[(V[0]-V[b],1)]*y^2+[(V[0]-V[b],2)]*y+[(V[0]-V[b^4+b],b^3+1)]*1,
[(V[b],1)]*x^2].
```

To obtain comprehensive Gröbner basis for $\{F,I\}$, we have to also compute ACGB in cases (1) and (2).

In case (1) : ($b+\frac{c+1}{a+1}=0$, the support is $[\mathbb{V}(0)-\mathbb{V}((a+1)(c+1))]$),

```
[[(V[a^2+a,(-c^2-c)*a-c^2-c]-V[c*a+c,a^2+a],1),(V[(-c^2-c)*a^2+(-c^2-c)*a
 ]-V[(-c-1)*a],1),(V[(-c^2-c)*a^2+(-c^2-c)*a]-V[(c^2+c)*a+c^2+c],1)]*1,
[(V[(-c-1)*a^2+(-c-1)*a]-V[a^2+a],1)]*y+[(V[(-c^2-c)*a-c^2-c,(-c-1)*a^2+(
 -c-1)*a]-V[a^2+a,(-c^2-c)*a-c^2-c],(c)/(a))]*1,
[(V[0]-V[(c^2+c)*a^2+(c^2+c)*a],1)]*y^2+[(V[0]-V[(c^2+c)*a^2+(c^2+c)*a],(
 2*c)/(a))]*y+[(V[0]-V[(-c^5-c^4)*a^5+(-4*c^5-4*c^4)*a^4+(-5*c^5-2*c^4+6*
```

```
 c^3+4*c^2+c)*a^3+(-3*c^5+6*c^3+4*c^2+c)*a^2+(-c^5-c^4)*a],(c^3*a^3+3*c^3
 *a^2+(2*c^3-3*c^2-3*c-1)*a+c^3)/(c*a^5+3*c*a^4+3*c*a^3+c*a^2))]*1,
[(V[0]-V[(c^2+c)*a^2+(c^2+c)*a],1),(V[-c^2-c,(-c-1)*a]-V[-c-1],1)]*x+[(V
 [0]-V[(c^2+c)*a^2+(c^2+c)*a],(-a^2-a)/(c+1))]*y+[(V[0]-V[(-c^2-c)*a^2+(-
 c^2-c)*a],(-c*a-c)/(c+1))]*1,
[(V[(c^2+c)*a+c^2+c]-V[(c^2+c)*a+c^2+c,c*a^2+c*a],1)]*x^2].
```

In case (2) : ($b = 0$, the support is $[\mathbb{V}((a+1)(c+1)) - \mathbb{V}(a+1)]$),

```
[[(V[c*a]-V[c],1)]*1,
 [(V[0]-V[a],1)]*y+[(V[0]-V[c*a],(c)/(a))]*1,
 [(V[0]-V[c*a],1)]*x^2]
```

Then in each case, we obtained a parametric Gröbner basis for $\langle F\rangle \cup I$ by using ACGB method.

We described a natural idea to generalize the algorithm DCGB. The first key point of this idea is what variables are regarded as parameters (or variables). Then we compute ACGB because we need the information of the support. The second key point is that "add an information of support to an original ideal in $K[\bar{A}]$". By the algorithm Div-zero we can classify the original ideal I to some zero-dimensional ideals in several polynomial rings. Then, we can apply the computation method DCGB (zero-dimensional) for computing comprehensive Gröbner bases. This is the procedure of computing ACGB-V. Since we need to compute a Gröbner basis in a polynomial ring over a von Neumann regular ring, the outputs hold the nice properties of (reduced) Gröbner basis over von Neumann regular rings. For instance, if we substitute any values for parameters of the outputs (reduced Gröbner bases in a polynomial ring over a von Neumann regular ring), then the set computed is always the reduced Gröbner bases.

6.7 A direct products of fields approach to comprehensive Gröbner bases over finite fields

In this section, we propose a computation method for computing comprehensive Gröbner bases over finite fields by using finite direct products of fields. In fact, this approach is the same as Algorithm 6.6.5. We apply Algorithm 6.6.5 for computing comprehensive Gröbner bases over finite fields.

As we saw some methods for computing comprehensive Gröbner bases in commutative polynomial rings in chapter 4, 5 and 6. In polynomial rings over a finite field, theoretically we can construct an algorithm for computing comprehensive Gröbner bases over finite fields, except for the method of alternative comprehensive Gröbner bases (ACGB). However, there are only implementations for characteristic zero fields.
The method of ACGB can not construct comprehensive Gröbner bases for positive characteristic, because it needs the injection $K[\bar{A}] \to K^{K^m}$. We know that K^{K^m} is a commutative von Neumann regular ring. We can apply the theory of von Neumann regular rings to parametric polynomials. However, if K is a finite field, then the maps $K[\bar{A}] \to K^{K^m}$ are not injections. Therefore, we can not use the method of ACGB for the case finite fields. However, we can apply the method of DCGB for them. In this section, we apply the method of DCGB for computing omprehensive Gröbner bases over finite fields. In [Nab05b], this approach has been studied by the author.

6.7.1 Some mathematical facts

In this subsection we show three lemmas and relations between parametric polynomials and polynomials over von Neumann regular rings. Some of the lemmas are from [SSN03b].

Lemma 6.7.1. Let K be a field, $S_1, \ldots, S_m$ non-empty finite subsets of K, $\bar{A} = \{A_1, \ldots, A_m\}$ indeterminate, $p_i(A_i) := \prod_{q \in S_i}(A_i - q)$ for each $i = 1, \ldots, m$. Let $\bar{\alpha}_1, \ldots, \bar{\alpha}_M$ be an enumeration of the set $\{(a_1, \ldots, a_m) | a_i \in S_i$ for each $i = 1, \ldots, m$ $\}$ and $\bar{\alpha_j} = (\alpha_{j1}, \ldots, \alpha_{jm})$ for each $j = 1, \ldots, M$.
Then $K[\bar{A}]/\langle p_1(A_1), \ldots, p_m(A_m)\rangle$ is isomorphic to $K[\bar{A}]/\langle A_1 - \alpha_{11}, \ldots, A_m - \alpha_{1m}\rangle \times \cdots \times K[\bar{A}]/\langle A_1 - \alpha_{M1}, \ldots, A_n - \alpha_{Mm}\rangle$ and K^M. (Recall that every direct product of fields is a von Neumann regular ring in section 6.1. $K[\bar{A}]/\langle p_1(A_1), .., p_m(A_m)\rangle$ is a von Neumann regular ring.) The mapping Φ from $K[\bar{A}]/\langle p_1(A_1), \ldots, p_m(A_m)\rangle$ to $K[\bar{A}]/\langle A_1 - \alpha_{11}, \ldots, A_n - \alpha_{1n}\rangle \times \cdots \times K[\bar{A}]/\langle A_1 - \alpha_{M1}, \ldots, A_n - \alpha_{Mm}\rangle$ defined by $\Phi(h(\bar{A})) = (h(\bar{\alpha}_1), .., h(\bar{\alpha}_M))$ is an isomorphism.

Proof. This is a easy consequence of the Chinese remainder theorem. $\langle A_1 - \alpha_{i1}, \ldots, A_n - \alpha_{im}\rangle$ is a maximal ideal for each $i = 1, \ldots, M$ in $K[\bar{A}]$. $K[\bar{A}]/\langle A_1 - \alpha_{i1}, \ldots, A_n - \alpha_{im}\rangle$ is isomorphic to the field K. Hence, $K[\bar{A}]/\langle p_1(A_1), \ldots, p_m(A_m)\rangle$ is isomorphic to K^M. □

Let $\mathbb{F}$ be a finite field with cardinality $|\mathbb{F}| = p$, and $\bar{A}$ parameters which can take any elements from $\mathbb{F}$. Then we can consider all elements of $\mathbb{F}$ as every solutions of the equation $\prod_{j=1,\ q_j \in \mathbb{F}}^{p}(Y - q_j) = 0$ where Y is a indeterminate. Hence we can consider that parameters are constrained the equations, and thus, like the case DCGB, we have the quotient ring $\mathbb{F}[\bar{A}]_{\mathbb{F}} := \mathbb{F}[\bar{A}]/\ \langle\prod_{j=1,\ q_j \in \mathbb{F}}^{p}(A_i - q_j) \mid i = 1, \ldots, m\rangle$. Consequently, $\mathbb{F}[A_1, \ldots, A_m]_{\mathbb{F}}$ is isomorphic to $\mathbb{F}^{(p^m)}$ which is a finite direct product of field. Namely, we can consider $\mathbb{F}[\bar{A}]_{\mathbb{F}}$ instead of $\mathbb{F}^{(p^m)}$. This observation leads us to consider Gröbner bases over commutative von Neumann regular ring for comprehensive Gröbner bases over finite fields.

Let f be a polynomial in $\mathbb{F}[A_1, \ldots, A_m]_{\mathbb{F}}[\bar{X}]$ where $A_1, \ldots, A_m$ are parameters and $\bar{X}$ are variables. Then, by the above observation, we can consider f as an element of $\mathbb{F}^{(p^m)}[\bar{X}]$. We can apply this fact to compute Gröbner bases in $\mathbb{F}[A_1, \ldots, A_m]_{\mathbb{F}}[\bar{X}]$.

We need the following two lemmas to construct an algorithm for computing Gröbner bases over finite fields. The following lemma directly follows from theorem 6.2.19.

Lemma 6.7.2 ([SSN03b]). Let K be a field and $R := K^M$ a von Neumann regular ring where $M \in \mathbb{N}$. Fix a term order on $\mathrm{pp}(\bar{A})$. Let $G = \{g_1, g_2, \ldots g_k\}$ be the stratified reduced Gröbner basis of an ideal $\langle f_1, f_2, \ldots, f_l\rangle$ in a polynomial ring $R[\bar{X}]$. Then, $\{{g_1}^{(i)}, {g_2}^{(i)}, \ldots, {g_k}^{(i)}\}$ becomes the reduced Gröbner basis of the ideal $\langle {f_1}^{(i)}, {f_2}^{(i)}, \ldots, {f_l}^{(i)}\rangle$ in the polynomial ring $K[\bar{X}]$ for each $i = 1, 2, \ldots, M$. Where, $h^{(i)}$ denotes a polynomial in $K[\bar{X}]$ given from a polynomial h of $R[\bar{X}]$ with replacing each coefficient c in h by the ith coordinate of c. Remember that c is an element of K^M, so it has a form $c = (c_1, c_2, \ldots, c_M)$ for some elements $c_1, c_2, \ldots c_M$ in K.

Lemma 6.7.3 ([SSN03b]). With the same notations and conditions in lemma 6.7.2, let $G_i = \{{g_1}^{(i)}, {g_2}^{(i)}, \ldots {g_k}^{(i)}\}$ for each i. Then for any polynomial h in $R[\bar{X}]$, we have $(h{\downarrow_G})^{(i)} = h^{(i)}{\downarrow_{G_i}}$ for each i. Remark that, $h{\downarrow_G}$ denotes the normal form of h by the reductions of S.

Proof. The proof is essentially same as the proof of property (2) of theorem 3.3 [SS02] or the proof of property 2 of the theorem 3.2 [SS03]. □

6.7.2 Comprehensive Gröbner bases over finite fields

In this subsection we show comprehensive Gröbner bases over finite fields, comprehensive Gröbner bases with restricted values of parameters and membership problems of parametric polynomials.

- $\boxed{\text{Comprehensive Gröbner bases over finite fields}}$

We have already known the relation between parametric polynomials over finite fields and polynomials over von Neumann regular rings in the previous subsection. We also know Algorithm 6.3.5 for computing Gröbner basis over a von Neumann regular ring. We are ready to state comprehensive Gröbner bases over finite fields by using a finite direct product of fields.

Theorem 6.7.4. Let $\mathbb{F}$ be a finite field with cardinality $|\,\mathbb{F}\,| = p$, and $\bar{A}$ parameters which can take arbitrary elements from $\mathbb{F}$. Then, $\mathbb{F}[\bar{A}]_{\mathbb{F}}$ becomes a von Neumann regular ring as is shown in the previous subsection. Let H be a finite set of polynomials in $\mathbb{F}[\bar{A}]_{\mathbb{F}}[\bar{X}]$, where $\bar{X}$ are variables. Fix a term order on $\mathrm{pp}(\bar{X})$. Considering H to be a finite set of polynomials in $\mathbb{F}^{p^m}[\bar{X}]$, construct the stratified reduced Gröbner basis G of the ideal $\langle H\rangle$ in this polynomial ring. Then we have the following properties. For any m-tuple $\bar{a} = (a_1, \dots a_m) \in \mathbb{F}^m$,

1. the set of polynomials
   $G(\bar{a}) = \{g(\bar{a}, \bar{X}) | g \in G\}$ is the reduced Gröbner basis of the ideal generated by the set of polynomials $H(\bar{a}) = \{f(\bar{a}, \bar{X}) | f \in H\}$ in $\mathbb{F}[\bar{X}]$.
2. For any $h(\bar{A}, \bar{X})$ in $\mathbb{F}[\bar{A}][\bar{X}]$,
   $(h(\bar{A}, \bar{X})\downarrow_G)(\bar{a}, \bar{X}) = h(\bar{a}, \bar{X})\downarrow_{G}(\bar{a})$.

Remark that $h(\bar{a}, \bar{X})$ denotes a polynomial in $\mathbb{F}[\bar{X}]$ given from a polynomial $h(\bar{A}, \bar{X})$ by substituting each A_i with a_i.

Proof. The first property follows from lemma 6.7.1 and lemma 6.7.2, the second property follows from lemma 6.7.1 and lemma 6.7.3. □

The theorem above states that G is a comprehensive Gröbner basis for $\langle H\rangle$, because the property 1 is the same property of comprehensive Gröbner bases. Hence, by this theorem, we can follow Algorithm 6.3.5 for computing comprehensive Gröbner bases over finite fields.
The next algorithm has the same notations in theorem 6.7.4.

Algorithm 6.7.5. $\mathsf{CGBoverF}(I, \succ)$ (CGB over finite fields)

Input: F : a finite set of polynomials in $\mathbb{F}[\bar{A}][\bar{X}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{X})$,
Output: G : a comprehensive Gröbner basis for $\langle F\rangle$ with respect to $\succ$.

1. Consider F as a set of polynomials in $\mathbb{F}^{(p^m)}[\bar{X}]$.
   (We can consider that $\mathbb{F}[\bar{A}]$ has the restriction $\langle\prod_{j=1,\ q_j\in\mathbb{F}}^{p}(A_i - q_j) \mid i = 1, \dots, m\rangle$.)
   Recall that $\mathbb{F}[\bar{A}]_{\mathbb{F}}[\bar{X}]$ is isomorphic to $\mathbb{F}^{(p^m)}[\bar{X}]$. Let π be a such isomorphism map, and compute $\pi(F)$.

2. Compute Gröbner basis H for $\langle \pi(F) \rangle$ in the polynomial ring over a over von Neumann regular ring.
3. Consider H as a set of polynomials in $\mathbb{F}[\bar{A}][\bar{X}]$. Namely, compute $\pi^{-1}(H)$ where π^{-1} is the inverse map of π.
4. $G \leftarrow \pi^{-1}(H)$.

Note that for π and π^{-1}, we can apply an algorithm of the Chinese remainder theorem. Remark that in the author's experience, the computation of $\pi^{-1}(H)$ is much more expensive than the computation of $\pi(F)$. The most expensive part of this algorithm is $\pi^{-1}(F)$.

In Figure 6.3, we see the rough procedure of Algorithm 6.7.5.

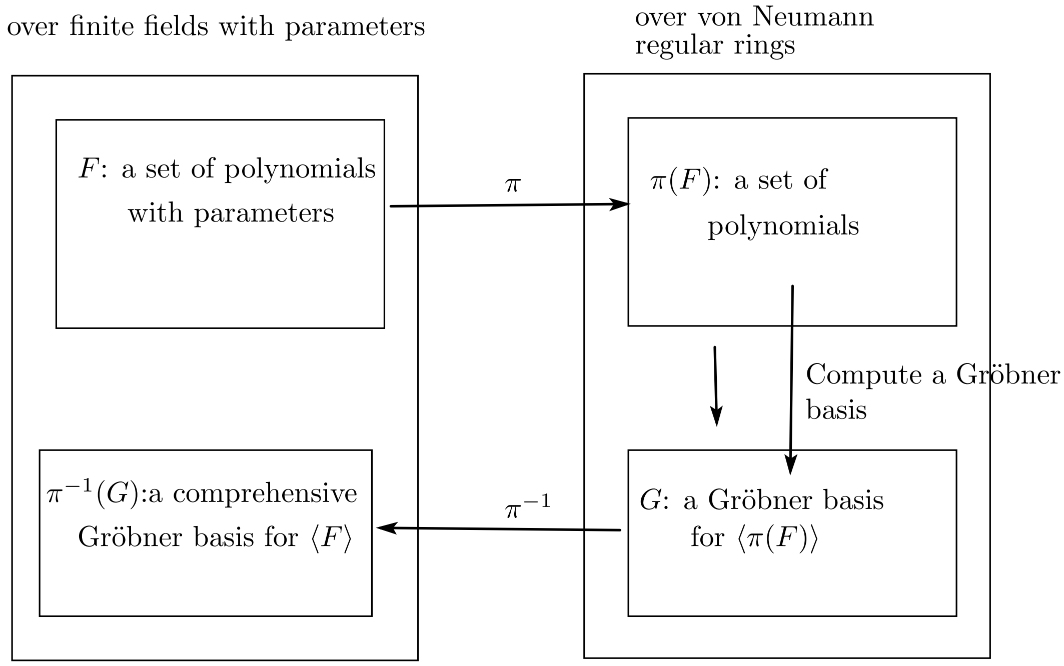

Figure 6.3 A computation method for CGB over finite fields

The algorithm CGBoverF has been implemented in prolog by the author. In the following example, we see an example of the output. The computer is a PC with Celeron 600 MHz, Memory 128 MB RAM, OS:Vine Linux 3.1[*2].

Example 6.7.6. Let f_1, f_2 and f_3 be the following polynomials in $\mathbb{Z}/13\,\mathbb{Z}[a,b][x,y,z]$ where x, y, z are variables and a, b parameters (i.e. the values of parameters are from $\mathbb{Z}/13\,\mathbb{Z}$.).

$$\{f_1 := 2axy^2 + 10z + 2b, f_2 := 7aby^2z + 2bx + 9, f_3 := bxz + ayz + 2a\}.$$

The term order $\succ$ is the graded reverse lexicographic order such that $x \succ y \succ z$. The program outputs the following list of polynomials as a comprehensive Gröbner basis for $\langle f_1, f_2, f_3 \rangle$ with respect to $\succ$.

[*2] The Linux distribution with integrated Japanese Environment for PCs.
`http://vinelinux.org/`

```
(12*a^12*b^12+1),
(a^12*b^12)*z*x+(a*b^11)*z*y+(2*a*b^11),
(a^12*b^12)*y*x+(6*a*b^12)*y^2+(11*a^11*b)*z^2+(11*a^12*b^11)*y+(10*a^11*
 b^2)*z,
(a^12*b^12)*x^2+(2*a^12*b^12)*z^2+(11*a^12*b^11)*x+(3*a^12*b)*z,
(a^12*b^12)*z^3+(a*b^10)*z*y+(8*a^12*b)*z^2+(12*a*b^11+11*a*b^10)*x+(a^2*
 b^10)*y+(2*a*b^10+4*a*b^9),
(a^12*b^12)*z*y^2+(4*a^11*b^12)*x+(5*a^11*b^11),
(a^12*b^12)*y^3+(4*a^10*b)*z^2*y+(4*a^11*b^11)*y^2+(6*a^10*b^2)*z*y+(10*a
 ^10*b^12)*z+(2*a^10*b).
```

(CPU time: 13.690 sec.)
This list has 7 polynomials.

- Restricted values of parameters

In the previous subsection, we usually assume that the parameters can take arbitrary values from $\mathbb{F}$. However, if there exist some constraints among parameters, then it is more natural to construct comprehensive Gröbner bases for only parameters satisfying such constraints, like ACGB-V.

Let us consider the following example.
Let $F = \{a^2+4a+3, b^2+2b, ax^2y+2x+b^2y+5, bxy^2+2abxy+x\}$ in $\mathbb{Z}/7\mathbb{Z}[a,b][x,y]$ where x, y are variables and a, b are parameters whose values are taken from $\mathbb{Z}/7\mathbb{Z}$. Of course, we can compute a comprehensive Gröbner basis for $\langle F\rangle$ by the algorithm CGBoverF. However, in this case, we have $F \cap \mathbb{Z}/7\mathbb{Z}[a,b] = \{a^2+4a+3, b^2+2b\}$. This means that parameters a, b are related to $a^2+4a+3=0$ and $b^2+2b=0$. Therefore, parameter a can take only $4, 6$, parameter b can take only $0, 5$. Hence, we do not need to consider arbitrary values in $\mathbb{Z}/7\mathbb{Z}$. We might use the information $a = 4, 6$ and $b = 0, 5$ to compute a comprehensive Gröbner basis more efficient.

Let F be a set of polynomials in $\mathbb{F}[\bar{A}][\bar{X}]$, and $F \cap \mathbb{F}[\bar{A}] = H (\neq \emptyset)$. Then, since $\mathbb{F}$ is a finite field, $\mathbb{V}(H)$ is the finite set. Let $|\mathbb{V}(H)| = M$ which is the cardinality of $\mathbb{V}(H)$ and $\mathbb{V}(H) = \{\bar{\alpha}_i = (a_{1i}, \dots, a_{mi}) \in \mathbb{F}^m | i = 1, \dots, M\}$. Then, $\mathbb{F}[\bar{A}]/\langle A_1 - a_{11}, \dots, A_m - a_{m1}\rangle \times \cdots \times \mathbb{F}[\bar{A}]/\langle A_1 - a_{1M}, \dots, A_m - a_{mM}\rangle$ is isomorphic to $\mathbb{F}^M$ by lemma 6.7.1. The mapping Φ defined by $\Phi(h(\bar{A})) = (h(\bar{\alpha}_1), \dots, h(\bar{\alpha}_M))$ where $h \in \mathbb{F}[\bar{A}]$. We can compute a comprehensive Gröbner basis for $\langle F\rangle$ and $F \cap \mathbb{F}[\bar{A}]$.
Note that if m-tuple $(b_1, \dots, b_m) \in \mathbb{F}^m$ does not belong to $\mathbb{V}(H)$, then a comprehensive Gröbner basis for F at $(b_1, \dots, b_m)$ is $\{1\}$. As there exists $f \in H$ such that $f(b_1, \dots, b_m) = c \in \mathbb{F}$, c is not 0.

Example 6.7.7. Let $F := \{a^2+4a+3, b^2+2b, ax^2y+2x+b^2y+5, bxy^2+2abxy+x\}$ in $\mathbb{Z}/7\mathbb{Z}[a,b][x,y]$where a, b are parameters, x, y are variables. In this case we have $H := F \cap \mathbb{Z}/7\mathbb{Z}[a,b] = \{a^2+4a+3, b^2+2b\}$. That is, $\mathbb{V}(H) = \{(a,b)|(4,0),(4,5),(6,0),(6,5)\}$. By using $\mathbb{V}(H)$, we can compute efficiently a comprehensive Gröbner basis for $\langle F\rangle$. The following polynomials are the comprehensive Gröbner basis for $\langle F\rangle$ with $V(H)$.

If $a^2+4a+3 \neq 0 \wedge b^2+2b \neq 0$, then
$\{1\}$.

If $a^2+4a+3 = 0 \wedge b^2+2b = 0$, then

$\{(4b+1),(3b)x^2+(3ab+5b)xy+(6ab+3b)y^2+(5ab+b)x+(4ab+b)y+(4b),(3b)y^3+(6ab+2b)y^2+(4ab+2b)y+(5ab+4b),(3b)xy^2+6abxy+(4ab+6b)x\}$.

• Membership problems

Membership problem is a basic problem in polynomial algebra. We can solve the parametric ideal membership problem by comprehensive Gröbner bases.

Remark 6.7.8 (Parametric ideal membership). Let f be a polynomial in $K[\bar{A}][\bar{X}]$ where K is a field. Let F be a set of polynomials in $K[\bar{A}][\bar{X}]$. Then parametric ideal membership is defined by the following. For any m-tuple $\bar{a}=(a_1,\dots,a_m)\in K^m$, $f\in\langle F\rangle$ in $K[\bar{A}][\bar{X}]$ if and only if $f(\bar{a})$ in $\langle F_{\bar{a}}\rangle$ in $K[\bar{X}]$, where $F_{\bar{a}}:=\{h(\bar{a},\bar{X})|h\in F\}$ in $K[\bar{X}]$

We consider parametric polynomials over a finite field as polynomials over a regular ring by. Therefore, we apply the reduction in a polynomial ring over a von Neumann regular ring. We apply this fact for solving parametric ideal membership problems.

Theorem 6.7.9. Let F be a set of polynomials in $\mathbb{F}[\bar{A}][\bar{X}]$, f a polynomial in $\mathbb{F}[\bar{A}][\bar{X}]$ and G a comprehensive Gröbner basis for $\langle F\rangle$ with respect to a term order $\succ$. Then there exists a following isomorphic mapping of the form;

$$\Phi:\mathbb{F}[\bar{A}][\bar{X}]\longrightarrow\mathbb{F}^{(p^m)}[\bar{X}],$$

by lemma 6.7.1.
Then, if $\Phi(f)\xrightarrow{*}_{\Phi(G)}0$, then $f\in\langle F\rangle$.

Proof. $\Phi(G)$ is a Gröbner basis in $\mathbb{F}^{(p^m)}[\bar{X}]$ by theorem 6.7.4, and $\Phi(f)$ is a polynomial in $\mathbb{F}^{(p^m)}[\bar{X}]$. Therefore, we can use the reduction of the theory of von Neumann regular rings. Then, by Lemma 6.7.3 and property (2) of Theorem 6.7.4, this theorem holds. □

Example 6.7.10. Let $F:=\{7bcx^2y+2y+5,bxz+ax+2,2axz^2+bx+10\}$, $f:=9bcx^2y+7bxz+7ax+y$ and $g:=3bcx^2yz+6yz+6aby+7z+6$ in $\mathbb{Z}/11\,\mathbb{Z}[a,b,c][x,y,z]$ where a,b,c are parameters and x,y,z are variables. Then f belongs to ideal generated by F and g doesn't belong to ideal generated by F. This program was implemented in prolog. (The computer is a PC Celeron 600 MHz, Memory 128 MB RAM, OS:Vine Linux 3.1.) The outputs are the following.

```
para_member([9*b*c*x^2*y+7*b*x*z+7*a*x+y],[7*b*c*x^2*y+2*y+5,b*x*z+a*x+2,
2*a*x*z^2+b*x+10],[[a,b,c],[x,y,z]],11).
Member !
yes
```

(CPU time: 8.970 sec.)

```
|?- para_member([3*b*c*x^2*y*z+6*y*z+6*a*b*y+7*z+6],[7*b*c*x^2*y+2*y+5,b*
x*z+a*x+2,2*a*x*z^2+b*x+10],[[a,b,c],[x,y,z]],11).
Not member!
yes
```

(CPU time: 9.030 sec.)

In our program, `para_member(A,B,[P,V],F)` is a function which decides whether `A` is a member of `B` or not. First, this function computes a Gröbner basis G for ideal generated by `B` over a von Neumann regular ring. Second, this function reduces `A` by G in the polynomial ring over a von Neumann regular ring. If $A\xrightarrow{*}_G 0$, then the function returns

"`Member !`". Otherwise, "`Not member!`". `P` is a set of parameters, `V` is a set of variables and `F` is a finite field.

6.7.3 Concluding remarks

In this section, we have studied comprehensive Gröbner bases in the case where the ground field has only finitely many elements. It has been shown that comprehensive Gröbner bases have particularly nice properties in this special case. An algorithm was presented for computing comprehensive Gröbner bases over finite fields, which always produces both canonical comprehensive Gröbner bases and the simple polynomial list.

Actually, our algorithm becomes extremely expensive in both space and time complexity if the ground field or the number of parameters are too big. However, if the cardinality of the field and the number of parameters are small, then our approach is useful for computing comprehensive Gröbner bases over finite fields.

Chapter 7

Two kinds of Gröbner bases in rings of differential operators

It is well-known that the Buchberger algorithm has been generalized to the non-commutative Gröbner bases area via non-commutative Gröbner bases in various contexts ([Gal85, Mor86, Mor88, Mor94, Li,02, KRW90] etc). In this chapter, we describe relations between two kinds of Gröbner bases in rings of differential operators and their algorithms for computing their Gröbner bases. In details, we treat two domains "rings of differential operators with coefficients in a field" and "rings of differential operators with coefficients in a polynomial ring". This chapter is similar to chapter 3 the case of commutative polynomials rings. However, in this chapter, we treat non-commutative rings. This relation has been studied in [Nab07e] by the author.

7.1 Notations

In this section we describe the notations for rings of differential operators and their related definitions. K and L denote fields of characteristic zero such that L is an extension of K. $\bar{X} = \{x_1, \ldots, x_n\}$ denotes a finite set of variables.

Let $K[\bar{X}]$ be a ring of polynomials in n variables over K. Let $\partial_i = \frac{\partial}{\partial x_i} : K[\bar{X}] \to K[\bar{X}]$ be the partial derivative by x_i, $1 \le i \le n$. Let $K[\bar{X}][\bar{D}] := (K[\bar{X}])[\partial_1, \ldots, \partial_n]$ be the rings of differential operators with coefficients in $K[\bar{X}]$, and $K[\bar{X}, \bar{D}] := K[x_1, \ldots, x_n, \partial_1, \ldots, \partial_n]$ the rings of differential operators (and variables) with coefficients in K. The both rings $K[\bar{X}][\bar{D}]$ and $K[\bar{X}, \bar{D}]$ have the commutation rules

$$x_i x_j = x_j x_i,\ \partial_i \partial_j = \partial_j \partial_i,\ \partial_i x_j = x_j \partial_i,\ \text{ for } i \neq j, \text{ and } \partial_i x_i = x_i \partial_i + 1.$$

It is well-known that $K[\bar{X}, \bar{D}]$ is a left-Noetherian associative K-algebra. By an "ideal in $K[\bar{X}, \bar{D}]$" we always mean a left-ideal of $K[\bar{X}, \bar{D}]$. Similarly, $K[\bar{X}][\bar{D}]$ is a left-Noetherian, and "ideal in $K[\bar{X}][\bar{D}]$" we always mean a left-ideal of $K[\bar{X}][\bar{D}]$.

An element $p \in K[\bar{X}, \bar{D}]$ (or $K[\bar{X}][\bar{D}]$) can be written uniquely as a finite sum

$$p = \sum_{(\alpha,\beta)\in E} c_{\alpha\beta} \cdot X^\alpha D^\beta,$$

where $X^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$, $D^\beta = \partial_1^{\beta_1} \cdots \partial_n^{\beta_n}$, $c_{\alpha\beta} \in K \backslash \{0\}$ and $E \subset \mathbb{N}^{2n}$. We call this unique expression **normally ordered** expression. In other words, we have the following natural K-vector space isomorphism between the **commutative polynomial rings** $K[\bar{X}, \bar{\Lambda}] :=$

$K[\bar{X}, \lambda_1, \ldots, \lambda_n]$ and $K[\bar{X}, \bar{D}]$:

$$\begin{aligned} \Psi \ : \ K[\bar{X}, \bar{\Lambda}] &\to K[\bar{X}, \bar{D}] \\ X^\alpha \Lambda^\beta &\mapsto X^\alpha D^\beta. \end{aligned}$$

Note that $K[\bar{X}, \bar{\Lambda}]$ is a commutative rings, $K[\bar{X}, \bar{D}]$ is a non-commutative ring. The notations $\mathrm{pp}(\bar{X}, \bar{D})$ and $\mathrm{pp}(\bar{D})$ are the set of power products of $\bar{X} \cup \bar{D}$ and $\bar{D}$, respectively.

Definition 7.1.1. A order $\succ$ is called a **term order** on the set of normally ordered power products $x^\alpha \partial^\beta$ in $K[\bar{X}, \bar{D}]$ where $\alpha, \beta \in \mathbb{N}^n$ if 1 is the smallest element and $x^\alpha \partial^\beta \succ x^a \partial^b$ implies $x^{\alpha+s} \partial^{\beta+t} \succ x^{a+s} \partial^{b+t}$ where $s, t \in \mathbb{N}^n$. We also call $\succ$ as a **term order** on $\mathrm{pp}(\bar{D})$ if 1 is the smallest element and $\partial^a \succ \partial^b$ implies $\partial^{a+p} \succ \partial^{b+p}$ where $p \in \mathbb{N}^n$.

Actually, in this chapter and the next chapter, we treat Gröbner bases and comprehensive Gröbner bases for several rings $K[\bar{X}, \bar{D}], K[\bar{X}][\bar{D}], K[\bar{A}][\bar{X}, \bar{D}]$ and $(K[\bar{A}][\bar{X}])[\bar{D}]$. Due to avoid the confusion we apply the subscript $\bar{X}$, $\{\bar{X}, \bar{D}\}$, $\bar{A}$ which are depend on the coefficient domain of these rings, for the notations of the rings. As we saw the notation of commutative polynomial rings, we use the same notations for the rings of differential operators as follows.

Definition 7.1.2 (Notations). Let $\succ$ be a term order on $\mathrm{pp}(\bar{X}, \bar{D})$. For a non-zero normally ordered element $f = \sum_{(\alpha,\beta)} c_{(\alpha,\beta)} \cdot X^\alpha D^\beta \in K[\bar{X}, \bar{D}]$ (or $K[\bar{X}][\bar{D}]$) where $X^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$, $D^\beta = \partial_1^{\beta_1} \cdots \partial_n^{\beta_n}$ and $\alpha, \beta \in \mathbb{N}^n$.

- We define the degree of f with respect to $\succ$ as $\deg_{\{\bar{X},\bar{D}\}}(f)$ (or $\deg_{\bar{D}}(f)$). (**Note that the subscript is $\{\bar{X}, \bar{D}\}$ (or $\bar{D}$).**)
- We define the leading coefficient of f as $\mathrm{lc}(f)$ (or $\mathrm{lc}_{\bar{X}}(f)$). That is, $\mathrm{lc}(f) := c_{\deg_{\{\bar{X},\bar{D}\}}(f)} \in K$ (or $\mathrm{lc}_{\bar{X}}(f) := c_{\deg_{\bar{D}}(f)} \in K[\bar{X}]$).
- We define the **initial form** of f as $\mathrm{in}(f)$ (or $\mathrm{in}_{\bar{D}}(f)$). That is, $\mathrm{in}(f) := \mathrm{lc}(f) X^\alpha D^\beta \in K[\bar{X}, \bar{D}]$ where $\deg_{\{\bar{X},\bar{D}\}}(f) = (\alpha, \beta) = (\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n) \in \mathbb{N}^{2n}$ (or $\mathrm{in}_{\bar{X}}(f) := \mathrm{lc}_{\bar{X}}(f) X^\alpha D^\beta \in K[\bar{X}][\bar{D}]$).
- The **set of normally ordered monomials** of f is defined by $\mathrm{Mono}(f)$ (or $\mathrm{Mono}_{\bar{X}}(f)$).
- If F is a subset of $K[\bar{X}, \bar{D}]$, we define the **set of initial form** of F as $\mathrm{in}(F)$ (or $\mathrm{in}_{\bar{X}}(F)$).

Remark: In section 5 and 6, we consider the two rings $K[\bar{A}][\bar{X}, \bar{D}]$ and $(K[\bar{A}][\bar{X}])[\bar{D}]$. For the ring $K[\bar{A}][\bar{X}, \bar{D}]$, we use the notations $\deg_{\{\bar{X},\bar{D}\}}$, $\mathrm{lc}_{\bar{A}}(f)$, $\mathrm{in}_{\bar{A}}(f)$ and $\mathrm{in}_{\bar{A}}(F)$. For the ring $(K[\bar{A}][\bar{X}])[\bar{D}]$, we use the notations $\deg_{\{\bar{D}\}}$, $\mathrm{lc}_{\{\bar{A},\bar{X}\}}(f)$, $\mathrm{in}_{\{\bar{A},\bar{X}\}}(f)$ and $\mathrm{in}_{\{\bar{A},\bar{X}\}}(F)$.

Definition 7.1.3. Fix a term order. The leading monomial $\mathrm{lm}(p)$ of $p \in K[\bar{X}, \bar{D}]$ is the commutative monomial $kx_1^{\alpha_1} \cdots x_n^{\alpha_n} \lambda_1^{\beta_1} \cdots \lambda_n^{\beta_n} \in K[\bar{X}, \bar{\Lambda}]$ such that $kx_1^{\alpha_1} \cdots x_n^{\alpha_n} \partial_1^{\beta_1} \cdots \partial_n^{\beta_n}$ is the largest monomial with respect to $\succ$ in $\mathrm{Mono}(p)$, where $k \in K$. We define the leading power product $\mathrm{lpp}(p)$ of p as the commutative power product $x_1^{\alpha_1} \cdots x_n^{\alpha_n} \lambda_1^{\beta_1} \cdots \lambda_n^{\beta_n} \in K[\bar{X}, \bar{\Lambda}]$. That is, $\mathrm{lc}(p)\,\mathrm{lpp}(p) = \mathrm{lm}(p)$.
When we consider the ring $K[\bar{X}][\bar{D}]$, then we apply $\mathrm{lm}_{\bar{X}}(p)$ as the leading monomial of $p \in K[\bar{X}][\bar{D}]$ and $\mathrm{lpp}_{\bar{X}}(p)$ as the the leading power product of p.

Note that, for $p \in K[\bar{X}, \bar{D}]$ we use the two notations $\mathrm{in}(p)$ and $\mathrm{lm}(p)$. The notation $\mathrm{in}(p)$ is an element of a non-commutative ring, and the notation $\mathrm{lm}(p)$ is an element of a commutative ring. For example, we have $p_1 = \partial_1, p_2 = x_1 \in \mathbb{Q}[x_1, \partial_1]$. Then, $\mathrm{in}(p_1) \cdot \mathrm{in}(p_2) = \partial_1 x_1 = x_1 \partial_1 + 1$ however, $\mathrm{lm}(p_1) \cdot \mathrm{lm}(p_2) = x_1 \lambda_1 = x_1 \partial_1$. Therefore, $\mathrm{in}(p_1) \cdot \mathrm{in}(p_2) \neq \mathrm{lm}(p_1) \cdot \mathrm{lm}(p_2)$.

The concrete examples of the notations is the following.

Example 7.1.4. Let x_1, x_2 be variables and ∂_1, ∂_2 be derivation of x_1, x_2, respectively. Then we have $f = 2x_1\partial_1^2\partial_2 + 2x_2\partial_1^2\partial_2 + \partial_1 + x_1 + 3$.

If we consider polynomials f as a member of $\mathbb{Q}[x_1, x_2, \partial_1, \partial_2]$ with a block order $\succ_{\{\partial_1,\partial_2\},\{x_1,x_2\}} := (\partial_1 \succ_{lex} \partial_2, x_1 \succ_{lex} x_2)$ where $\succ_{lex}$ is the lexicographic order, then

- $\deg_{\{x_1,x_2,\partial_1,\partial_2\}}(f) = (1, 0, 2, 1) \in \mathbb{N}^4$,
- $\mathrm{lc}(f) = 2 \in \mathbb{Q}$,
- $\mathrm{lpp}(f) = x_1\partial_1^2\partial_2$,
- $\mathrm{in}(f) = \mathrm{lm}(f) = 2x_1\partial_1^2\partial_2$,
- $\mathrm{Mono}(f) = \{2x_1\partial_1^2\partial_2, 2x_2\partial_1^2\partial_2, \partial_1, x_1, 3\}$.

If we consider polynomials f as a member of $\mathbb{Q}[x_1, x_2][\partial_1, \partial_2]$ with the lexicographic order $\partial_1 \succ \partial_2$, then

- $\deg_{\{\partial_1,\partial_2\}}(f) = (2, 1) \in \mathbb{N}^2$, $\deg_{\{\partial_1\}} = 2$,
- $\mathrm{lc}_{\{x_1,x_2\}}(f) = 2x_1 + 2x_2 \in \mathbb{Q}[x_1, x_2]$,
- $\mathrm{lpp}_{\{x_1,x_2\}}(f) = \partial_1^2\partial_2$,
- $\mathrm{in}_{\{x_1,x_2\}}(f) = \mathrm{lm}_{\{x_1,x_2\}}(f) = (2x_1 + 2x_2)\partial_1^2\partial_2$,
- $\mathrm{Mono}_{\{x_1,x_2\}}(f) = \{(2x_1 + 2x_2)\partial_1^2\partial_2, \partial_1, x_1, 3\}$.

7.2 Gröbner bases in $K[\bar{X},\bar{D}]$

In this section we quickly review an algorithm for computing Gröbner bases in $K[\bar{X},\bar{D}]$ (Weyl algebra). The results of Buchberger [Buc65, Buc70] on Gröbner bases in polynomial rings have been generalized by several authors to several fields. We can also generalize the theory to $K[\bar{X},\bar{D}]$ ([Oak02, SST99, Gal85, Cas86, Cas87, OS94, Tak89]). This algorithm is the same as the Buchberger algorithm. As we saw the Buchberger algorithm in chapter 2, basically we follow the same way for describing the algorithm.

Definition 7.2.1 (S-polynomial). Fix a term order $\succ$. We have two non-zero **normally ordered elements** $f = c_1X^\alpha D^\beta + f'$ and $g = c_2X^aD^b + g'$ where $X^\alpha = x_1^{\alpha_1}\cdots x_n^{\alpha_n}$, $D^\beta = \partial_1^{\beta_1}\cdots\partial_n^{\beta_n}$, $\deg_{\{\bar{X},\bar{D}\}}(f) = (\alpha, \beta) \in \mathbb{N}^{2n}$, $\mathrm{lc}(f) = c_1$, $\mathrm{lpp}(f) = X^\alpha D^\beta$, $\deg_{\{\bar{X},\bar{D}\}}(g) = (a, b) \in \mathbb{N}^{2n}$, $\mathrm{lc}(g) = c_2$, $\mathrm{lpp}(g) = X^aD^b$ and $f', g' \in K[\bar{X},\bar{D}]$. Then, for the two elements f, g, we define the **S-polynomial** of f and g by

$$\texttt{Spoly1}(f, g) = X^{\alpha'}D^{\beta'}f - \tfrac{c_1}{c_2}X^{a'}D^{b'}g$$

where $\alpha'_i = \max(\alpha_i, a_i) - \alpha_i$, $\beta'_i = \max(\beta_i, b_i) - \beta_i$, $a'_i = \max(\alpha_i, a_i) - a_i$, $b'_i = \max(\beta_i, b_i) - b_i$.

In the next section, we introduce another type of S-polynomials. In order to distinguish another S-polynomial from this S-polynomial, we call this S-polynomial "Spoly1".

Next we introduce a reduction in $K[\bar{X},\bar{D}]$ which is called "Reduce1".

Definition 7.2.2 (Reduction). Let $f = c_1X^\alpha D^\beta + f'$ and $g = c_2X^aD^b + g'$ be normally ordered elements in $K[\bar{X},\bar{D}]$ such that X^α divides X^a and D^α divides D^b, where $X^\alpha = x_1^{\alpha_1}\cdots x_n^{\alpha_n}$, $D^\beta = \partial_1^{\beta_1}\cdots\partial_n^{\beta_n}$, $\mathrm{in}(f) = c_1X^\alpha D^\beta$, $c_1, c_2 \in K$, $X^a \in \mathrm{pp}(\bar{X})$, $D^b \in \mathrm{pp}(\bar{D})$

and $f', g' \in K[\bar{X}, \bar{D}]$. Note that $c_2 X^a D^b$ might not be the initial form of g. A reduction $\xrightarrow{r1}_f$ is defined as follows:

$$g \xrightarrow{r1}_f g - \frac{c_1}{c_2} X^s D^t f,$$

where $X^s X^a = X^\alpha$, $D^t D^b = D^\beta$ and $s, t \in \mathbb{N}^n$.

In this thesis, we call this reduction `Reduce1` (written: $\xrightarrow{r1}$), because in the next section, we introduce another reduction which is called `Reduce2` (see Definition 7.3.1). A reduction $\xrightarrow{r1}_F$ by a set $F \subseteq K[\bar{X}, \bar{D}]$ is also naturally defined, like chapter 2.

Definition 7.2.3 (Gröbner basis). Let I be an ideal in $K[\bar{X}, \bar{D}]$ and $\succ$ a term order on $\mathrm{pp}(\bar{X}, \bar{D})$. A subset $G \subset K[\bar{X}, \bar{D}]$ is called a Gröbner basis of I with respect to $\succ$ if

- I is generated by G, and
- $\mathrm{lm}(I)$ is generated by $\mathrm{lm}(G)$, i.e., $\mathrm{lm}(I) = \langle \mathrm{lm}(G) \rangle$ where $\mathrm{lm}(I) := \{\mathrm{lm}(q) | q \in I\}$.

Using the S-polynomial Spoly1, the reduction Reduce1 and a term order, we can construct an algorithm for computing Gröbner bases for an ideal generated by a given finite subset of $K[\bar{X}, \bar{D}]$. This algorithm is exactly the Buchberger algorithm for $K[\bar{X}, \bar{D}]$ as follows.

Algorithm 7.2.4. GröbnerBasisW($F, \succ$) (Gröbner bases in Weyl algebra)

Input $F = \{f_1, \ldots, f_s\}$: a finite set of normally ordered elements in $K[\bar{X}, \bar{D}]$,
  $\succ$: a term order.
Output G: a Gröbner basis for $\langle F \rangle$ with respect to $\succ$.
begin
  $G \leftarrow F$
  $P \leftarrow \{(f_i, f_j) \,|\, 1 \le i < j \le s\}$
   while $P \ne \emptyset$ **do**
    Take any element (f, f') from P
     $P \leftarrow P \backslash \{(f, f')\}$
     $h \leftarrow$ `Spoly1`(f, f')
     $r \leftarrow h \downarrow_G$ (by `Reduce1`) (see below $(*)$)
      if $r \ne 0$ **then**
       $P \leftarrow P \cup \{(g, r) \,|\, g \in G\}$
       $G \leftarrow G \cup \{r\}$
      end-if
   end-while
  `return`(G)
end
($(*)$ $h \downarrow_G$ denotes a normal form of h by $\xrightarrow{r1}_G$, i.e., $h \downarrow_G$ is irreducible by $\xrightarrow{r1}_G$)

Example 7.2.5. Let I be an ideal in $\mathbb{Q}[x_1, x_2, \partial_1, \partial_2]$ generated by $x_2\partial_2^2 + x_1\partial_2, x_1^2\partial_2 - x_1^2, x_1\partial_2 + x_1$ and $\succ$ the lexicographic order such that $x_1 \succ x_2 \succ \partial_1 \succ \partial_2$. A Gröbner basis G for I with respect to $\succ$ is $G = \{\partial_2^2 x_2, x_1\}$.

The algorithm above is not efficient. The study of efficiency is an active research area in commutative case. Actually, Many of the familiar results on Gröbner bases in $K[\bar{X}]$ also hold in $K[\bar{X}, \bar{D}]$. However, the generalization of the first Buchberger's criterion [Buc79] does not hold. For example, let $f = \partial_2 + x_1, g = \partial_1 \in \mathbb{C}[x_1, x_2, \partial_1, \partial_2]$. Then, by the first Buchberger's criterion, $\{f, g\}$ is a Gröbner basis. However, $\mathsf{Spoly1}(f, g) = \partial_1 f - \partial_2 g = x_1\partial_1 + 1$, $\mathsf{Spoly1}(f, g) \xrightarrow{r1}_g 1$. Hence, the Gröbner basis is $\{1\}$. Therefore, the generalization of the criterion does not hold.

We are able to define a reduced Gröbner basis in $K[\bar{X}, \bar{D}]$ like commutative polynomial rings. The definition of reduced Gröbner bases is the following.

Definition 7.2.6 (Reduced Gröbner Bases). The reduced Gröbner basis G of an ideal $I \subseteq K[\bar{X}, \bar{D}]$ with respect to a monomial order $\succ$ is a Gröbner basis such that:

1. $\mathrm{lc}(g) = 1$ for all $g \in G$ with respect to $\succ$,
2. for all $g \in G$, no monomial of $\mathrm{Mono}(g)$ lies in $\langle \mathrm{lm}(G\backslash\{g\})\rangle$.

Although a Gröbner basis may contain redundant elements, the reduced Gröbner basis does not contain any. The reduced Gröbner is uniquely determined by an ideal and a term order $\succ$.

Algorithm 7.2.7. $\mathsf{RGr\ddot{o}bnerBasisW}(F, \succ)$ (Reduced Gröbner Bases)

Input $F = \{f_1, \dots, f_m\}$: a subset of $K[\bar{X}, \bar{D}]$,
$\succ$: a term order.
Output G: a reduced Gröbner basis for $\langle F\rangle$ with respect to $\succ$.

- **We assume that** $\mathsf{RGr\ddot{o}bnerBasisW}(F)$ **is an algorithm which outputs the reduced Gröbner basis with respect to $\succ$ in $K[\bar{X}, \bar{D}]$.**

In the algorithm CGBW and the proof of Theorem 8.1.19, we need the properties of the reduced Gröbner bases, and an algorithm for computing reduced Gröbner bases. Therefore, we introduced the definition of reduced Gröbner bases and assumed the algorithm RGröbnerBasisW.

7.3 Approach by Insa and Pauer for computing Gröbner bases in $K[\bar{X}][\bar{D}]$

There are several results of Gröbner bases [SST99, Cas86, Cas87, OS94] in rings of differential operators, however the coefficient rings are fields (of rational function) or rings of power series. In this section and the next section, we consider an algorithm for computing Gröbner bases in $K[\bar{X}][\bar{D}]$. That is, the coefficient ring is a polynomial ring and the main variables are $\bar{D} := \{\partial_1, \dots, \partial_n\}$. In this section, first we review the Insa-Pauer algorithm [IP98] which returns Gröbner bases in $K[\bar{X}][\bar{D}]$. The second part of this section, we introduce Zhou-Winkler's criterion [ZW06] for computing Gröbner basis in $K[\bar{X}][\bar{D}]$.

7.3.1 Insa-Pauer's algorithm

In [IP98], Insa and Pauer studied the theory of Gröbner bases in $K[\bar{X}][\bar{D}]$. In their paper, they introduced a special S-polynomial and a special reduction in order to compute Gröbner bases in $K[\bar{X}][\bar{D}]$. In this subsection, we describe the special S-polynomial, the special reduction and the Insa-Pauer algorithm.

Definition 7.3.1. Let F be a set of normally ordered elements of $K[\bar{X}][\bar{D}]$ and $g = a\beta + g' \in K[\bar{X}][\bar{D}]$ where $a \in K[\bar{X}]$, $\beta \in \mathrm{pp}(\bar{D})$ and $g' \in K[\bar{X}][\bar{D}]$. Moreover, let

$F' := \{f \in F \mid \mathrm{lpp}_{\bar{X}}(f) \text{ divides } \beta\}$. If $a \in \langle \mathrm{lc}_{\bar{X}}(F') \rangle \subseteq K[\bar{X}]$, the element a can be written as $a = \sum_{f_i \in F'} h_i \,\mathrm{lc}_{\bar{X}}(f_i)$ where $h_i \in K[\bar{X}]$. Then a reduction $\xrightarrow{r2}_f$ is defined as follows:

$$g \xrightarrow{r2}_F g - \sum_{f_i \in F'} h_i \frac{\beta}{\mathrm{lpp}_{\bar{X}}(f_i)} f_i.$$

In this thesis, we define this reduction as `Reduce2` (written: $\xrightarrow{r2}$). Actually, reducing g by F and reducing g by F' is the same. In this case, we can write the reduction $g \xrightarrow{r2}_{F'}$ instead of $g \xrightarrow{r2}_F$.

In fact, we can see this reduction as "the extended Gröbner bases algorithm " (see chapter 2).

Definition 7.3.2 ([IP98]). Let G be a set of normally ordered elements of $K[\bar{X}][\bar{D}]$ and let I be an ideal in $K[\bar{X}][\bar{D}]$ generated by G. For $E \subseteq G$, let

$$S_E := \left\{ (c_e)_{e \in E} \;\middle|\; \sum_{e \in E} c_e \,\mathrm{lc}_{\bar{X}}(e) = 0 \right\}.$$

(We can consider S_E as a set of syzygies for $\mathrm{lc}_{\bar{X}}(E) \subseteq K[\bar{X}]$.) Then for $s = (c_e)_{e \in E} \in S_E$,

$$\mathsf{Spoly2}(E, s) = \sum_{e \in E} c_e D^{\max(E) - \deg_{\bar{D}}(e)} e$$

is called S-polynomial with respect to E and s ,where

$$\max(E) := (\max_{e \in E} \deg_{\bar{D}}(e)_1, \ldots, \max_{e \in E} \deg_{\bar{D}}(e)_n) \in \mathbb{N}^n .$$

In this thesis, we called this special S-polynomial as " Spoly2".

Insa-Pauer defined the Gröbner bases in $K[\bar{X}][\bar{D}]$ which is the following.

Definition 7.3.3 (Gröbner bases). Let I be an ideal in $K[\bar{X}][\bar{D}]$ and let G be a finite subset of $I \backslash \{0\}$. For $i \in \mathbb{N}^n$ let

$$\mathrm{lc}_{\bar{X}}(i, I) := \langle \mathrm{lc}_{\bar{X}}(f) | f \in I, \deg_{\bar{D}}(f) = i \rangle.$$

Then G is a **Gröbner basis** of I (with respect to a term order $\succ$ on $\mathrm{pp}(\bar{D})$) if and only if $\forall i \in \mathbb{N}^n$ the ideal $\mathrm{lc}_{\bar{X}}(i, I) \subseteq K[\bar{X}]$ is generated by

$$\{\mathrm{lc}_{\bar{X}}(g) \mid g \in G, i \in \deg_{\bar{D}}(g) + \mathbb{N}^n \} .$$

Remark: In fact the definition is equivalent to the following. G is called a **Gröbner basis** with respect to $\succ$ if

- I is generated by G,
- $\mathrm{lm}_{\bar{X}}(I)$ is generated by $\mathrm{lm}_{\bar{X}}(G)$, i.e., $\mathrm{lm}_{\bar{X}}(I) = \langle \mathrm{lm}_{\bar{X}}(G) \rangle$ where $\mathrm{lm}_{\bar{X}}(I) = \{\mathrm{lm}_{\bar{X}}(f) | f \in I\}$.

One might easily understand this definition than Definition 7.3.3. However, in this thesis, we adopt Definition 7.3.3 as the definition of Gröbner basis in $K[\bar{X}][\bar{D}]$.

A Gröbner basis in $K[\bar{X}][\bar{D}]$ has a lot of properties which are well-known in polynomial rings over a field. For instance, if G is a Gröbner basis for an ideal I in $K[\bar{A}][\bar{X}]$, then

$\forall g \in I$, $g \xrightarrow{r2}_G 0$. In this thesis, we do not describe the detail of properties of Gröbner bases in $K[\bar{X}][\bar{D}]$ (see [IP98]). Insa-Pauer presented the following algorithm which outputs a Gröbner basis in $K[\bar{X}][\bar{D}]$.

Algorithm 7.3.4. $\mathsf{InPaD}(F, \succ)$ [IP98]

Input: $F = \{f_1, \ldots, f_s\}$: a finite subset of $K[\bar{X}][\bar{D}]$,
$\succ$: a term order on $\mathrm{pp}(\bar{D})$,
Output: G: a Gröbner basis of $\langle F \rangle$ with respect to $\succ$ in $K[\bar{X}][\bar{D}]$.
begin
  $G \leftarrow F$; $B \leftarrow \{(f_{i_1}, f_{i2} \ldots f_{ip}) \mid 1 \le i_1 < i_2 \cdots < i_p \le s, 2 \le p \le s\}$
  while $B \neq \emptyset$ **do**
   Take any element E from B; $B \leftarrow B \backslash \{E\}$
   $S_E \leftarrow$ Compute a syzygy module of $\mathrm{lc}_{\bar{X}}(E)$ in $K[\bar{X}]$
   while $S_E \neq \emptyset$ **do**
    Take any element α from S_E; $S_E \leftarrow S_E \backslash \{\alpha\}$
    $h \leftarrow \mathsf{Spoly2}(E, \alpha)$
    $r \leftarrow h \downarrow_G$ (see below $(*)$)
    if $(r \neq 0)$ **then**
     $B \leftarrow B \cup \left\{(r, g_{j_1}, \ldots, g_{j_q}) \,\middle|\, \text{distinct elements } g_{j_1}, \ldots, g_{j_p} \in G, 1 \le p \le |G| \right\}$
     $G \leftarrow G \cup \{r\}$
    end-if
   end-while
  end-while
  `return`(G)
end
($(*)$ $h \downarrow_G$ denotes a normal form of h by $\xrightarrow{r2}_G$, i.e., $h \downarrow_G$ is irreducible by $\xrightarrow{r2}_G$)

7.3.2 Zhou-Winkler's criterion

Here, we describe the techniques to remove unnecessary combinations in the algorithm InPaD. As we said earlier, we need the special S-polynomial Spoly1 and the special reduction Reduce1 in order to compute Gröbner bases in $K[\bar{X}][\bar{D}]$. In this point, this algorithm is very complicated. Actually, there exists a criterion for eliminating unnecessary combinations which is Zhou and Winkler's work [ZW06]. By this criterion, we can improve the algorithm InPaD. We introduce the criterion.

Definition 7.3.5 ([ZW06]). Let I be an ideal in $K[\bar{X}][\bar{D}]$, $E_1 = \{f_1, \ldots, f_s\} \subseteq I$ and $E_2 = \{g_1, \ldots, g_t\} \subseteq I$. Then Spoly2s corresponding to E_1 (or E_2) are said to be of grade s (or t). If $s < t$, then Spoly2s corresponding to E_1 are said to be of lower grade than Spoly22 corresponding to E_2.

Lemma 7.3.6 ([ZW06]). Let I be an ideal in $K[\bar{X}][\bar{D}]$. For $E = \{f_1, \ldots, f_k\} \subseteq I$, if some $\mathrm{lc}(f_i)$ is divided by $\mathrm{lc}(f_j)$ ($i \neq j$, $1 \le i, j \le k$) in $K[\bar{X}][\bar{D}]$, then all Spoly2s corresponding to E_2 are simplified to Spoly2s of lower grade.

Theorem 7.3.7 ([ZW06]). Let $G = \{f_1, \ldots, f_m\}$ and J be the left ideal of $K[\bar{X}][\bar{D}]$ generated by G. If all Spoly2s with grade k are reduced to 0 by G, then for $E = \{g_1, \ldots, g_k, g_{k+1}\} \subseteq G$ with some $\mathrm{lc}(g_j)$ divided by another $\mathrm{lc}(g_i)$, all of Spoly2s corresponding to E are reduced to 0 by G.

Example 7.3.8. From example 5 of [IP98], we have a Gröbner basis G of $\langle x_1\partial_2, x_2\partial_1\rangle$ in $\mathbb{C}[x_1,x_2][\partial_1,\partial_2]$ with the graded lexicographic order $\partial_1 \succ \partial_2$. This Gröbner basis was computed by only using Spoly2s of grade 2 in [IP98].

$$G = \{x_1\partial_2, x_2\partial_1, x_2\partial_2 - x_1\partial_1, x_1^2\partial_1, x_1\partial_1^2 + 2\partial_1\}.$$

However for any three elements $\{f_i, f_j, f_k\} \subset G$, there exists an $\mathrm{lc}(f_l)$ which is divided by another $\mathrm{lc}(f_s) \in \{f_i, f_j, f_k\}\backslash\{f_l\}$ where $i, s \in \{i, j, k\}$. Therefore by Theorem 7.3.7, we ignore all Spoly2 with higher grade than 2.
For instance, select $\{x_2\partial_1, x_2\partial_2 - x_1\partial_1, x_1^2\partial_1\}$, then obviously $\mathrm{lc}(x_2\partial_2 - x_1\partial_1) = x_1$ divide $\mathrm{lc}({x_1}^2\partial_1) = {x_1}^2$.

By Theorem 7.3.7, we can improve the algorithm InPaD to more efficient one for computing Gröbner bases in $K[\bar{X}][\bar{D}]$.

The algorithm InPaD which includes Zhou-Winkler's criterion, has been implemented by the author in the computer algebra system Risa/Asir. The following example, we see outputs of the program.

Example 7.3.9. Let $F_1 = \{x_2^2\partial_3\partial_4 + x_1\partial_2, x_3\partial_2 + x_4\partial_4, \partial_3 + x_1\partial_4, x_1\partial_2\partial_3 + x_2\partial_3\}$ and $F_2 = \{x_2^2\partial_3^2 + x_1, x_3^2\partial_2 + \partial_3, \partial_1\partial_2^2 + x_1\partial_3, x_1^2\partial_3\}$ be subsets of $\mathbb{C}[x_1,x_2,x_3,x_4][\partial_1,\partial_2,\partial_3,\partial_4]$. We have the graded lexicographic order $\succ$ such that $\partial_1 \succ \partial_2 \succ \partial_3 \succ \partial_4$. Then the program outputs the following as a Gröbner basis for $\langle F_1\rangle$ with respect to $\succ$

```
Number of the maximal grade of Spoly:
2
[-x1*x3*d4+x4*d4,d2,d3,x1*d4]
```

This output means that we need only Spoly2s of grade 2 for computing Gröbner basis (by the Zhou-Winkler's criterion), and a Gröbner basis for $\langle F_1\rangle$ is

$$\{-x_1x_3\partial_4 + x_4\partial_4, \partial_2, \partial_3 x_1\partial_4\}.$$

The program outputs the following as a Gröbner basis for $\langle F_2\rangle$ with respect to $\succ$

```
Number of the maximal grade of Spoly:
3
[x1,d3,d2,x2*d3,x3*d3]
```

The maximal grade of Spoly2s is 3, and a Gröbner basis for $\langle F_2\rangle$ is

$$\{x_1, \partial_3, \partial_2, x_2\partial_3, x_3\partial_3\}.$$

7.4 Approach via block orders for computing Gröbner bases in $K[\bar{X}][\bar{D}]$

Here we present another algorithm for computing Gröbner bases in $K[\bar{X}][\bar{D}]$ which is much more efficient than the algorithm InPaD. Obviously, $K[\bar{X}][\bar{D}]$ is isomorphic to $K[\bar{X},\bar{D}]$. By the algorithm GröbnerBasisW (or RGröbnerBasisW) and a block order with $\bar{D} \gg \bar{X}$ in $K[\bar{X},\bar{D}]$, we can obtain an efficient algorithm for computing Gröbner bases in $K[\bar{X}][\bar{D}]$. The key theorem is the following. Remember that we applied the same idea in chapter 3 for computing Gröbner bases in $K[\bar{A}][\bar{X}]$. In rings of differential operators, we can follow the same way and obtain an efficient algorithm.

Theorem 7.4.1. Let F be a subset of $K[\bar{X}][\bar{D}]$. Then, F can be seen as a subset of $K[\bar{X}, \bar{D}]$. Let $G = \{g_1, \dots, g_s\}$ be a Gröbner basis for $\langle F \rangle$ in $K[\bar{X}, \bar{D}]$ with respect to a block order $\succ_{\bar{D},\bar{X}} := (\succ_1, \succ_2)$ (i.e, $\bar{D} \gg \bar{X}$). Then, G is also a Gröbner basis for $\langle F \rangle$ with respect to $\succ_1$ in $K[\bar{X}][\bar{D}]$.

Proof. Take $\forall h \in \langle F \rangle \subseteq K[\bar{X}][\bar{D}]$ such that $\deg_{\bar{D}}(h) = j$, then we prove that $\mathrm{lc}_{\bar{X}}(h)$ is generated by $\{\mathrm{lc}_{\bar{X}}(g) | g \in G, j \in \deg_{\bar{D}}(g) + \mathbb{N}^n\}$. Since h can be seen as an element of $K[\bar{X}, \bar{D}]$ and G is a Gröbner basis for $\langle F \rangle$ in $K[\bar{X}, \bar{D}]$, h can be written as

$$h = h_1 g_1 + \cdots + h_s g_s$$

such that $\mathrm{in}(h) \succ_{\bar{D},\bar{X}} \mathrm{in}(h_1 g_1) \succ_{\bar{D},\bar{X}} \cdots \succ_{\bar{D},\bar{X}} \mathrm{in}(h_s g_s)$ in $K[\bar{X}, \bar{D}]$ where $h_1, \dots, h_s \in K[\bar{X}, \bar{D}]$. As $h = h_1 g_1 + \cdots + h_s g_s$ in $K[\bar{X}, \bar{D}]$, $\mathrm{in}_{\bar{X}}(h) = \mathrm{in}_{\bar{X}}(h_1 g_1 + \cdots + h_s g_s)$ in $K[\bar{X}][\bar{D}]$. By the block order on $K[\bar{X}, \bar{D}]$, we have

$$\mathrm{in}_{\bar{X}}(h) \succ_1 \mathrm{in}_{\bar{X}}(h_1 g_1) \succ_1 \cdots \succ_1 \mathrm{in}_{\bar{X}}(h_s g_s)$$

in $K[\bar{X}][\bar{D}]$. W.l.o.g., $h_1 g_1, \dots, h_k g_k$ have the same degree of h (i.e., $\deg_{\bar{D}}(h) = j$) where $k \leq s$. That is,

$$\mathrm{lm}_{\bar{X}}(h) = \mathrm{lm}_{\bar{X}}(h_1 g_1) + \cdots + \mathrm{lm}_{\bar{X}}(h_k g_k).$$

(Note that now we are considering the leading monomials in commutative ring $K[\bar{X}, \bar{\Lambda}]$ (see section 7.2)). Hence, we can obtain the following form;

$$\mathrm{lm}_{\bar{X}}(h) = \mathrm{lm}_{\bar{X}}(h_1)\,\mathrm{lm}_{\bar{X}}(g_1) + \cdots + \mathrm{lm}_{\bar{X}}(h_k)\,\mathrm{lm}_{\bar{X}}(g_k).$$

Since $\deg_{\bar{D}}(h) = \deg_{\bar{\Lambda}}(h) = j$ and $\deg_{\bar{\Lambda}}(g_1), \dots, \deg_{\bar{\Lambda}}(g_s) \in \{i \,|\, j = i + \mathbb{N}^n\}$, $\mathrm{lc}_{\bar{X}}(h) \in \{\mathrm{lc}_{\bar{X}}(g) | g \in G, j \in \deg_{\bar{\Lambda}}(g) + \mathbb{N}^n\}$. Hence, by Ψ (see section 2), we have $\mathrm{lc}_{\bar{X}}(h) \in \{\mathrm{lc}_{\bar{X}}(g) | g \in G, j \in \deg_{\bar{D}}(g) + \mathbb{N}^n\}$. Therefore, G is also a Gröbner basis in $K[\bar{X}][\bar{D}]$ with respect to $\succ_1$. $\square$

Algorithm 7.4.2. $\bar{D}$-GröbnerBasis$(F, \succ_1)$

Input F: a finite subset of $K[\bar{X}][\bar{D}]$,
$\succ_1$: a term order on $\mathrm{pp}(\bar{D})$,
Output G: a Gröbner basis of $\langle F \rangle$ in $K[\bar{X}][\bar{D}]$.

1. Consider F as a subset of $K[\bar{X}, \bar{D}]$.
2. Compute a Gröbner basis G for $\langle F \rangle$ with respect to a block order $\succ_{\bar{D},\bar{X}} := (\succ_1, \succ_2)$ in $K[\bar{X}, \bar{D}]$.
3. Consider G as a subset of $K[\bar{X}][\bar{D}]$. Then, by Lemma 7.4.1, G is a Gröbner basis for $\langle F \rangle$ with respect to $\succ_1$ in $K[\bar{X}][\bar{D}]$.

This algorithm is exactly the same as the algorithm GröbnerBasisW. The key idea is a block order with $\bar{D} \gg \bar{X}$. If we apply the Insa-Pauer algorithm to compute Gröbner bases in $K[\bar{X}][\bar{D}]$, then we need syzygy computations as the special S-polynomials (`Spoly2`) and "extended Gröbner bases algorithm" as the special reduction (Reduce2). In general, syzygy computations and "extended Gröbner bases algorithm" are very expensive. Therefore, the Insa-Pauer algorithm is expensive, too. However, if we apply the algorithm $\bar{D}$-GröbnerBasis to compute Gröbner bases in $K[\bar{X}][\bar{D}]$, then we need only normal S-polynomials (`Spoly1`) and reductions (`Reduce1`). We do not need `Spoly2` and `Reduce2` to compute Gröbner bases in $K[\bar{X}][\bar{D}]$. In this point, the algorithm $\bar{D}$-GröbnerBasis is much more efficient and simpler than the algorithm InPaD.

Example 7.4.3. Let I be an ideal in $\mathbb{Q}[x_1, x_2][\partial_1, \partial_2]$ generated by $x_1\partial_2, x_2\partial_1 + x_1\partial_1^2$. We have the lexicographic order $\succ$ such that $\partial_1 \succ \partial_2$. We compute a Gröbner basis for I with respect to $\succ$ by the algorithm $\bar{D}$-GröbnerBasis.

1. First, we consider the set $\{x_1\partial_2, x_2\partial_1 + x_1\partial_1^2\}$ as a subset of $\mathbb{Q}[x_1, x_2, \partial_1, \partial_2]$.
2. Second, we compute a Gröbner basis for I with respect to a block order $\succ_{\{\partial_1,\partial_2\},\{x_1,x_2\}}$ with $\partial_1 \succ \partial_2 \gg x_1 \succ_{lex} x_2$ where $\succ_{lex}$ is the lexicographic order. Then, by the algorithm RGröbnerBasisW, we obtain the reduced Gröbner basis G with respect to $\succ_{\{\partial_1,\partial_2\},\{x_1,x_2\}}$ as follows:
$$G = \left\{\partial_2 x_2^2 - 4\partial_2 x_2 + 4\partial_2, \partial_2 x_1, -\partial_2^2 x_2 + 2\partial_2^2 - 2\partial_2, (-x_2+2)\partial_1, -x_1\partial_1 + \partial_2 x_2 - 2\partial_2\right\}.$$
3. We can see G as a subset of $\mathbb{Q}[x_1, x_2][\partial_1, \partial_2]$. Therefore, a Gröbner basis for I in $\mathbb{Q}[x_1, x_2][\partial_1, \partial_2]$ with respect to $\succ$ is G. □

Now we have the following question.

"Don't we really need the special S-polynomial Spoly2 **and the special reduction** Reduce2**?"**

If we compute a reduced Gröbner basis in $K[\bar{X}][\bar{D}]$, this answer is "**Yes, we do**". As we saw the commutative case in chapter 3, the algorithm $\bar{D}$-GröbnerBasis outputs sometimes redundant elements. Therefore in order to eliminate redundant elements from the output, we need the special reduction Reduce2. In the next section, we treat reduced Gröbner bases in $K[\bar{X}][\bar{D}]$.
If we need just a normal Gröbner basis in $K[\bar{X}][\bar{D}]$, then this answer is "**No, we don't**". **We don't need the special S-polynomial** Spoly2 **and the special reduction** Reduce2**.**

7.5 Reduced Gröbner bases in rings over a polynomial ring

In chapter 3, we saw reduced Gröbner bases in (commutative) polynomial rings over a polynomial ring. Here we also present reduced Gröbner bases in $K[\bar{X}][\bar{D}]$. In the previous section, we saw two algorithms which compute Gröbner bases in $K[\bar{X}][\bar{D}]$. However, both algorithms have problems to compute a reduced Gröbner basis.

For instance:
(1) : Let $f_1 = x_1^2\partial_1 - x_1$ and $f_2 = (x_1^3 - x_1)\partial_1 - x_1^2 + 1$ be differential operators in $\mathbb{Q}[x_1][\partial_1]$. Then, a Gröbner basis of $\langle f_1, f_2\rangle$ is $\{f_1, f_2\}$, because Spoly2$(f_1, f_2) = 0$, $f_1 \xrightarrow{r2}_{\{f_2\}} f_1$ and $f_2 \xrightarrow{r2}_{\{f_1\}} f_2$in $\mathbb{Q}[x_1][\partial_1]$. However, we have
$$f_3 = x_1 \cdot f_1 - f_2 = x_1\partial_1 - 1.$$
Of course, f_3 is an element of $\langle f_1, f_2\rangle$. Moreover, we have $f_3|f_1$, $f_3|f_2$. This means $\langle f_3\rangle = \langle f_1, f_2\rangle$. That is, $\{f_3\}$ is a Gröbner basis, too. $\{f_3\}$ is simpler than $\{f_1, f_2\}$. **However,** $\{f_3\}$ **cannot be computed by the algorithm** InPaD.

(2) : Let $\bar{X} = \{x_1, x_2, x_3, x_4\}$, $\bar{D} = \{\partial_1, \partial_2, \partial_3, \partial_4\}$ and $F = \{x_1\partial_2 - \partial_2 + x_4\partial_3^2, x_1\partial_3 + x_4\} \subset$

$\mathbb{Q}[\bar{X}][\bar{D}]$. By the algorithm 5, we compute a Gröbner basis for $\langle F\rangle$ with respect to a block order with $\bar{D} \gg \bar{X}$. We have the lexicographic order $\succ$ such that $\partial_1 \succ \partial_2 \succ \partial_3 \succ \partial_4 \succ x_1 \succ x_2 \succ x_3 \succ x_4$. Then the reduced Gröbner basis for $\langle F\rangle$ with respect to $\succ$ in $\mathbb{Q}[\bar{X}, \bar{D}]$ is

$$\{g_1 = x_1\partial_3 + x_4,\ g_2 = x_1\partial_2 - \partial_2 + x_4\partial_3^2,\ g_3 = -\partial_2\partial_3 - x_4\partial_2 + x_4\partial_3^3\}.$$

By Theorem 7.4.1, $\{g_1, g_2, g_3\}$ is a Gröbner basis for $\langle F\rangle$ with respect to $\succ$ in $\mathbb{Q}[\bar{X}][\bar{D}]$. However, there exists a redundant element in the set $\{g_1, g_2, g_3\}$. Look at g_3, then

$$\mathrm{lm}_{\bar{X}}(g_3) = -\partial_2\partial_3 \in \langle \mathrm{lm}_{\bar{X}}(g_1), \mathrm{lm}_{\bar{X}}(g_2)\rangle = \langle x_1\partial_3, (x_1 - 1)\partial_2\rangle.$$

That is, g_3 can be written as

$$g_3 = -\partial_2 g_1 + \partial_3 g_2.$$

Thus, g_3 is a redundant element. $\{g_1, g_2\}$ is a Gröbner basis, too. However, we cannot compute the set $\{g_1, g_2\}$ by algorithm $\bar{D}$-GröbnerBasis.

What is a reduced Gröbner basis in $K[\bar{X}][\bar{D}]$? How do we compute it?

The definition of reduced Gröbner bases is the following.

Definition 7.5.1. Let $\succ_{\bar{D},\bar{X}} := (\succ_1, \succ_2)$ be a block order and I an ideal in $K[\bar{X}][\bar{D}]$. Then, a **reduced Gröbner basis** G for I with respect to $\succ_1$ and $\succ_{\bar{X},\bar{D}}$ is a Gröbner basis for I in $K[\bar{X}][\bar{D}]$ such that

1. For all $p \in G$, $\mathrm{lc}(p) = 1$ with respect to $\succ_{\bar{D},\bar{X}}$,
2. For all $p \in G$, no monomial in $\mathrm{Mono}_{\bar{X}}(p)$ lies in $\langle \mathrm{lm}_{\bar{X}}(G\backslash \{p\})\rangle$ in $K[\bar{X}][\bar{D}]$ with respect to $\succ_1$,
3. For all $p \in G$, no monomial in $\mathrm{Mono}(p)$ lies in $\langle \mathrm{lm}(G\backslash \{p\})\rangle$ in $K[\bar{X}, \bar{D}]$ with respect to $\succ_{\bar{D},\bar{X}}$.

By using the two reduction Reduce1 and Reduce2, we can construct an algorithm for computing reduced Gröbner bases in $K[\bar{X}][\bar{D}]$.

Algorithm 7.5.2. RGB$(F, \succ_{\{\bar{D},\bar{X}\}})$

Input: F: a finite subset of $K[\bar{X}][\bar{D}]$,
$\succ_{\{\bar{D},\bar{X}\}} = (\succ_1, \succ_2)$: a block order such that $\bar{D} \gg \bar{X}$,
  $\succ_1$: a term order on $\mathrm{pp}(\bar{D})$,
  $\succ_2$: a term order on $\mathrm{pp}(\bar{X})$,
Output: G: a reduced Gröbner basis for $\langle F\rangle$ with respect to $\succ_1$ and $\succ_{\bar{D},\bar{X}}$.
begin
  $G \leftarrow$ $\Big($RGröbnerBasisW(F) with respect to $\succ_1\Big)$ or $\Big($InPaD(F) with respect to $\succ_1\Big)$
  $E1 \leftarrow 0$
 while $E1 \neq 1$ **do**
  if there exist $p \in G$ such that $\left(p \xrightarrow{r1}_{\{G\backslash\{p\}\}} p_1 \text{ and } p \neq p_1\right)$
      or $\left(p \xrightarrow{r2}_{\{G\backslash\{p\}\}} p_1 \text{ and } p \neq p_1\right)$ **then**
   if $p_1 \neq 0$ **then**
    $G \leftarrow \{G\backslash\{p\}\} \cup \{p_1\}$
   else-if

$G \leftarrow G \backslash \{p\}$
end-if
else-if
$E1 \leftarrow 1$
end-if
end-while
`return`(G)
end

Theorem 7.5.3. The algorithm $\mathsf{RGB}(F, \succ)$ terminates. The output forms a reduced Gröbner basis for $\langle F \rangle$ with respect to $\succ_1$ and $\succ_{\bar{D},\bar{X}}$ in $K[\bar{X}][\bar{D}]$.

Proof. In the first line, we compute a Gröbner basis for $\langle F \rangle$ wit respect to $\succ_1$ in $K[\bar{X}][\bar{D}]$ by RGröbnerBasis or InPaD. This step obviously terminates. In the first **while-loop**, if there exists an element $q \in G$ which can be reduced to q_1 by $\xrightarrow{r1}_{\{G\backslash\{p\}\}}$ or $\xrightarrow{r2}_{\{G\backslash\{p\}\}}$, then we have $\mathrm{lpp}_{\bar{X}}(q) \succ \mathrm{lpp}_{\bar{X}}(q_1)$ ($\mathrm{lpp}_{\bar{X}}(q_1)$ is smaller or equal than $\mathrm{lpp}_{\bar{X}}(q)$ with respect to $\succ_1$). That is, the result of $\xrightarrow{r1}_{\{G\backslash\{p\}\}}$ or $\xrightarrow{r2}_{\{G\backslash\{p\}\}}$ to p in $K[\bar{X}][\bar{D}]$ or $K[\bar{X}][\bar{D}]$ has a leading power product which cannot be greater than $\mathrm{lpp}_{\bar{X}}(q)$ w.r.t. $\succ_1$. Therefore, iterated application of $\xrightarrow{r1}$ or $\xrightarrow{r2}$ to G will eventually terminate. This algorithm terminates. The output is a reduced Gröbner basis. □

In chapter 3, we defined two kind of reduced Gröbner bases "weak" and "strong" ones in $K[\bar{A}][\bar{D}]$. In the ring $K[\bar{X}][\bar{D}]$, we can define two kind of reduced Gröbner bases, too. Actually, Definition 7.5.1 is about weak reduced Gröbner bases in $K[\bar{X}][\bar{D}]$. As we saw in chapter 3, weak reduced Gröbner bases are not uniquely determined by a given ideal in $K[\bar{X}][\bar{D}]$. We can easily construct string reduced Gröbner bases in $K[\bar{X}][\bar{D}]$. However, we do not describe then in this thesis. One can easily follow the same way of chapter 3 for constructing strong ones.

Chapter 8

Comprehensive Gröbner bases in rings of differential operators

In this chapter we present algorithms for computing comprehensive Gröbner bases and comprehensive Gröbner systems in $K[\bar{X}, \bar{D}]$ and $K[\bar{X}][\bar{D}]$ rings of differential operators. That is, we consider non-commutative comprehensive Gröbner bases. In [KW91, Kre92], Kredel and Weispfenning studied parametric Gröbner bases for non-commutative polynomials. In these papers, they applied the Weispfenning method [Wei92] to compute comprehensive Gröbner bases. In this chapter, we describe algorithms which are different from the Kredel and Weispfenning approach, for computing comprehensive Gröbner bases (and comprehensive systems) in rings of differential operators. Furthermore, these algorithms have been implemented by the author in the computer algebra system Risa/Asir, and theses algorithms are more efficient than the Kredel-Weispfenning one. Actually, our algorithms are the generalization of the Suzuki-Sato algorithms (which we saw in chapter 4) to the rings of differential operators. This chapter is bases on the author's paper [Nab07e].

8.1 Comprehensive Gröbner bases in $K[\bar{A}][\bar{X}, \bar{D}]$

Here we present comprehensive Gröbner bases and comprehensive Gröbner systems in $K[\bar{A}][\bar{X}, \bar{D}]$ rings of differential operators. Basically, we follow the Suzuki-Sato algorithm for constructing the algorithms for computing them in $K[\bar{A}][\bar{X}, \bar{D}]$, because in general, the Suzuki-Sato algorithm is faster than other existing algorithm. First, we treat the theory of the stability of ideals in $K[\bar{A}][\bar{X}, \bar{D}]$.

8.1.1 The Stability of ideals

In this subsection, we describe the stability of (left) ideals under specializations in rings of differential operators $K[\bar{A}][\bar{X}, \bar{D}]$. First we introduce a definition of Gröbner bases in $K[\bar{A}][\bar{X}, \bar{D}]$. As we saw definitions of Gröbner bases in several domains, one can easily imagine a definition of Gröbner bases which is the following.

Definition 8.1.1. Let I be an ideal in $K[\bar{A}][\bar{X}, \bar{D}]$ and $\succ$ a term order on $K[\bar{X}, \bar{D}]$. A subset $G \subset K[\bar{A}][\bar{X}, \bar{D}]$ is called a **Gröbner basis** of I with respect to $\succ$ if

- I is generated by G, and
- $\mathrm{lm}_{\bar{A}}(I) = \langle \mathrm{lm}_{\bar{A}}(G) \rangle$ where $\mathrm{lm}_{\bar{A}}(I) = \{\mathrm{lm}_{\bar{A}}(f) | f \in I\}$.

Theorem 7.4.1 leads us to the computation method of Gröbner bases in $K[\bar{A}][\bar{X}, \bar{D}]$. The key idea is again a block order. By block orders with $\bar{X}, \bar{D} \gg \bar{A}$ and the algorithm

GröbnerBasisW, we can compute a Gröbner basis in $K[\bar{A}][\bar{X},\bar{D}]$. As we saw in chapter 4, we define the following ring homomorphism.

Every ring homomorphism $\pi : K[\bar{A}] \to L$ extends naturally to a homomorphism $\pi : K[\bar{A}][\bar{X},\bar{D}] \to L[\bar{X},\bar{D}]$. The image under π of an ideal $I \subseteq K[\bar{A}][\bar{X},\bar{D}]$ generates the extension $\pi(I) := \{\pi(f)|f \in I\} \subseteq L[\bar{X},\bar{D}]$.

Definition 8.1.2. We call an ideal $I \subseteq K[\bar{A}][\bar{X},\bar{D}]$ **stable** under the ring homomorphism π and a term order $\succ$ if it satisfies

$$\pi(\mathrm{lm}_{\bar{A}}(I)) = \mathrm{lm}(\pi(I))$$

where $\pi(\mathrm{lm}_{\bar{A}}(I)) := \{\pi(\mathrm{lm}_{\bar{A}}(f))|f \in I\}$ and $\mathrm{lm}(\pi(I)) := \{\mathrm{lm}(f)|f \in \pi(I)\}$.

In several papers [Bec94, Gia87, Kal97, Sat05], the stability of ideals under specialization was studied in commutative polynomial rings. We can easily extend the concept of the stability of ideals under specialization to the rings of differential operators. Then, the generalization of "[Kal97, Theorem 3.1]" which is the next theorem, also holds in rings of differential operators. This theorem is the key theorem for constructing the algorithms for computing comprehensive Gröbner bases and comprehensive Gröbner systems.

Theorem 8.1.3 ([Kal97]). Let π be a ring homomorphism from $K[\bar{A}]$ to L, I an ideal in $K[\bar{A}][\bar{X},\bar{D}]$ and $G = \{g_1,\dots,g_s\}$ a Gröbner basis of I with respect to a term order $\succ$. We assume that the g_i's are ordered in such a way that there exists an $r \in \{1,\dots,s\}$ with $\pi(\mathrm{lc}_{\bar{A}}(g_i)) \neq 0$ for $i \in \{1,\dots,r\}$ and $\pi(\mathrm{lc}_{\bar{A}}(g_i)) = 0$ for $i \in \{r+1,\dots,s\}$. Then the following three conditions are equivalent.

1. I is stable under π and $\succ$.
2. $\{\pi(g_1),\dots,\pi(g_q)\}$ is a Gröbner basis of $\langle\pi(I)\rangle$ with respect to the term order $\succ$.
3. For every $i \in \{r+1,\dots,s\}$, $\pi(g_i)$ is reducible to 0 modulo $\{\pi(g_1),\dots,\pi(g_q)\}$ in $L[\bar{X},\bar{D}]$.

Proof. This proof is exactly same as Theorem 4.3.2 and [Kal97]. Note that the ring $K[\bar{X},\bar{D}]$ is a left Noetherian ring. □

8.1.2 Comprehensive Gröbner systems

Here we present an algorithm for computing comprehensive Gröbner systems in $K[\bar{A}][\bar{X},\bar{D}]$. Basically, this algorithm is the generalization of the Suzuki-Sato algorithm (see Algorithm 4.4.3) to the rings of differential operators. As we saw earlier, for arbitrary $\bar{a} \in L^m$, we define the canonical specialization homomorphism $\sigma_{\bar{a}} : K[\bar{A}] \to L$ induced by $\bar{a}$, and we can naturally extend it to $\sigma_{\bar{a}} : (K[\bar{A}])[\bar{X}] \to L[\bar{X}]$.

Definition 8.1.4 (Comprehensive Gröbner Systems). Let F be a subset of $K[\bar{A}][\bar{X},\bar{D}]$, $A_1,\dots,A_l$ algebraically constructible subsets of L^m and $G_1,\dots,G_l$ subsets of $K[\bar{A}][\bar{X},\bar{D}]$. Let S be a subset of L^m such that $S \subseteq A_1 \cup \dots \cup A_l$. A finite set $G = \{(A_1,G_1),\dots,(A_l,G_l)\}$ of pairs is called a **comprehensive Gröbner system** on S for $\langle F\rangle$ if $\sigma_{\bar{a}}(G_i)$ is a Gröbner basis of the ideal $\langle\sigma_{\bar{a}}(F)\rangle$ in $L[\bar{X}]$ for each $i = 1,\dots,l$ and $\bar{a} \in A_i$. Each (A_i,G_i) is called a segment of G. We simply say G is a comprehensive Gröbner system for $\langle F\rangle$ if $S = L^m$.

Definition 8.1.5. Let $S_1,\dots,S_l$ and $T_1,\dots,T_l$ be finite subset of polynomials $K[\bar{A}]$. A finite set $G = \{(S_1,T_1,G_1),\dots,(S_l,T_l,G_l)\}$ of triples is also called a **comprehensive Gröbner system** on S for $\langle F\rangle$, if $\{(\mathbb{V}(S_1)\setminus\mathbb{V}(T_1),G_1),\dots,(\mathbb{V}(S_l)\setminus\mathbb{V}(T_l),G_l)\}$ is a comprehensive Gröbner system on S for $\langle F\rangle$. Each (S_i, T_i, G_i) is also called a segment of the comprehensive Gröbner system G.

The next two lemmas are the direct consequences of Theorem 8.1.3.

Lemma 8.1.6. Let F be a subset $K[\bar{A}][\bar{X},\bar{D}]$. F can be seen as a subset of $K[\bar{A},\bar{X},\bar{D}]$ and we write the subset as F again. Let G be a Gröbner basis for $\langle F\rangle$ in $K[\bar{A},\bar{X},\bar{D}]$ with respect to a block order $\succ_{\{\bar{X},\bar{D}\},\bar{A}} := (\succ_1,\succ_2)$ (i.e., $\{\bar{X},\bar{D}\} \gg \bar{A}$). G can be seen as a subset of $K[\bar{A}][\bar{X},\bar{D}]$ and we write the subset as G again. Suppose that $\{h_1,\ldots,h_s\} := \{\mathrm{lc}_{\bar{A}}(g) | g \in G\}$ and $h := \mathsf{LCM}(h_1,\ldots,h_s)$. Then, for any $\bar{a} \in L^m \setminus \mathbb{V}(h)$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F)\rangle$ with respect to $\succ_1$ in $L[\bar{X},\bar{D}]$.

Proof. By Theorem 8.1.3 (3), I is stable. $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(F)\rangle$ with respect to $\succ_1$. $\square$

Lemma 8.1.7. Let F be a subset $K[\bar{A}][\bar{X},\bar{D}]$ and S a subset of $K[\bar{A}]$. F can be seen as a subset of $K[\bar{A},\bar{X},\bar{D}]$ and we write the subset as F again. Let G be the reduced Gröbner basis for $\langle F\cup S\rangle$ in $K[\bar{A},\bar{X},\bar{D}]$ with respect to a block order $\succ_{\{\bar{X},\bar{D}\},\bar{A}} := (\succ_1,\succ_2)$, (i.e., $\{\bar{X},\bar{D}\} \gg \bar{A}\}$). G can be seen as a subset of $K[\bar{A}][\bar{X},\bar{D}]$ and we write the subset as G again. Suppose that $B := \{b | b \in \langle S\rangle, b \in G\}$, $\{h_1,\ldots,h_s\} := \{\mathrm{lc}_{\bar{A}}(g) | g \in G\backslash B\}$ and $h := \mathsf{LCM}(h_1,\ldots,h_s)$. Then, for any $\bar{a} \in \mathbb{V}(S)\setminus\mathbb{V}(h)$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(F)\rangle$ with respect to $\succ_1$ in $L[\bar{X},\bar{D}]$. Actually, we have $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G\backslash B)$.

Proof. If we take $g \in G\backslash B$, then for all $\bar{a} \in \mathbb{V}(S)\setminus\mathbb{V}(h)$ we have $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) \neq 0$. If we take $g \in G \cap B$, then we have $\sigma_{\bar{a}}(g) = 0$ and $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) = 0$. Of course, $\langle 0\rangle$ is stable. Therefore, G is stable under the specialization $\sigma_{\bar{a}}$. By Theorem 8.1.3, $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G\backslash B)$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(F)\rangle$. $\square$

By the two lemmas, we can construct an algorithm for computing comprehensive Gröbner systems in $K[\bar{A}][\bar{X},\bar{D}]$.

Algorithm 8.1.8. $\mathsf{CGSW}(F,\succ_1)$ (Comprehensive Gröbner Systems in Weyl algebra)

Input F: a finite subset of $K[\bar{A}][\bar{X},\bar{D}]$,
  $\succ_1$: a term order on $\mathrm{pp}(\bar{X},\bar{D})$,
  $(\succ_{\{\bar{X},\bar{D}\},\bar{A}} = (\succ_1,\succ_2)$: a block order such that $\bar{X},\bar{D} \gg \bar{A}$ on $\mathrm{pp}(\bar{X},\bar{D},\bar{A})$,
  $\succ_2$: a term order on $\mathrm{pp}(\bar{A})$,)
Output H: a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ_1$ on L^m.
begin
  $G \leftarrow \mathsf{RGr\ddot{o}bnerBasisW}(F,\succ_{\{\bar{X},\bar{D}\},\bar{A}})$
if $1 \in G$ **then**
$\mathsf{return}(\{(\emptyset,\{1\},G)\})$
end-if
  $S \leftarrow \{h_1,\ldots,h_l\} := \{\mathrm{lc}_{\bar{A}}(g) | \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G\}$
if $S \neq \emptyset$ **then**
  $h \leftarrow \mathsf{LCM}(h_1,\ldots,h_l)$
  $H \leftarrow \{(\emptyset,\{h\},G)\} \cup \mathsf{CGSMainW}(G,\{h_1\},\succ_1) \cup \cdots \cup \mathsf{CGSMainW}(G,\{h_l\},\succ_1)$
else
  $H \leftarrow \{(\emptyset,\{1\},G)\}$
end-if
`return`(H)
end

Algorithm 8.1.9. CGSMainW($F, Z, \succ$)

Input F: a finite subset of $K[\bar{A}][\bar{X}, \bar{D}]$,
Z : a finite set of polynomials in $K[\bar{A}]$,
$\succ_1$: a term order on $\mathrm{pp}(\bar{X}, \bar{D})$,
$(\succ_{\{\bar{X},\bar{D}\},\bar{A}} = (\succ_1, \succ_2)$: a block order such that $\bar{X}, \bar{D} \gg \bar{A}$ on $\mathrm{pp}(\bar{X}, \bar{D}, \bar{A})$,
$\succ_2$: a term order on $\mathrm{pp}(\bar{A})$,)
Output H: a comprehensive Gröbner system for $\langle F \rangle$ on $\mathbb{V}(Z)$ with respect to $\succ_1$.
begin
$G \leftarrow$ RGröbnerBasisW($F \cup Z, \succ_{\{\bar{X},\bar{D}\},\bar{A}}$)
if $1 \in G$ **then**
$H \leftarrow (\{(Z, \{1\}, \{1\})\}$
else
$B \leftarrow \{g | g \in G \cap K[\bar{A}], g \in \langle Z \rangle\}$
$S \leftarrow \{h_1, \ldots, h_l\} := \{\mathrm{lc}_{\bar{A}}(g) | \, \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G \backslash B\}$
if $S \neq \emptyset$ **then**
$h \leftarrow$ LCM($h_1, \ldots, h_l$)
$H \leftarrow \{(Z, \{h\}, G \backslash B)\} \cup$ CGSMainW($G, Z \cup \{h_1\}, \succ_1$) $\cup \cdots$
$\cdots \cup$ CGSMainW($G, Z \cup \{h_l\}, \succ_1$)
else
$H \leftarrow \{(Z, \{1\}, G \backslash B)\}$
end-if
end-if
return(H)
end

Remark : We can apply a lot of optimization techniques [BW93, Mon02, MM06, SS03, Wei03] to obtain small and nice outputs comprehensive Gröbner systems. For instance,
(1) : We can factorize each element of S into the set of their irreducible factors.
(2) : We can check each condition of parameters of segments for eliminating redundant segments.
(3) : Since a leading coefficient of each polynomial of a segment does not vanish by the specialization, we can apply `Reduce1` in $K(\bar{A})[\bar{X}, \bar{D}]$ where $K(\bar{A})$ is the field of rational functions. That is, we can obtain reduced Gröbner bases by using only Reduce1 in $K(\bar{A})[\bar{X}, \bar{D}]$ in each segment.
(4) : We applied the algorithm RGröbnerBasisW in the algorithms above. The algorithm RGröbnerBasisW returns the reduced Gröbner basis in $K[\bar{A}, \bar{X}, \bar{D}]$. However, the algorithm does not return a reduced Gröbner basis in $K[\bar{A}][\bar{X}, \bar{D}]$. Namely, the outputs has some redundant elements. Therefore, we can eliminate these elements from the outputs. See Algorithm 7.5.2.

Theorem 8.1.10. Let F be a finite subset of $K[\bar{A}][\bar{X}, \bar{D}]$. Then the algorithm `CGSW`(F) terminates and outputs a comprehensive Gröbner system for $\langle F \rangle$ on L^m.

Proof. This proof is almost same as Theorem 4.4.5.
First we show the termination. The sub-algorithms RGröbnerBasisW and LCM terminates. We prove the termination of `CGSMainW`. We suppose that `CGSMainW`(F, Z) does not terminate where $F \subset K[\bar{A}][\bar{X}, \bar{D}]$ and $Z \subset K[\bar{A}]$, then there exists an infinite sequence $F_0, F_1, \ldots$, such that $F_0 = F$ and $F_i \neq F_{i+1}$ for $i \in \mathbb{N}$. By the algorithm, $F_{l+1} = F_l \cup \{h_l\}$ for some $h_n \in K[\bar{A}]$ such that $h_l \notin \langle F_l \rangle$. Hence we have $\langle F_l \rangle \subsetneq \langle F_{l+1} \rangle$ for each l. We

also have an infinite sequence $Z_0, Z_1, \dots$ such that $Z_0 = Z$ and $Z_j \neq Z_{j+1}$ for $j \in \mathbb{N}$, and $Z_{s+1} = Z_s \cup \{h_s\}$ for each $h_s \in K[\bar{A}]$ such that $h_s \notin \langle Z_s \rangle$. Hence, we have $\langle Z_s \rangle \subsetneq \langle Z_{s+1} \rangle$ for each s. We know that every infinite ascending chain $M_1 \subseteq M_2 \subseteq \cdots$ of ideals of $K[\bar{A}]$ stabilizes. That is, there exists N such that $M_n = M_{N+1} = \cdots = M_{N+j} = \cdots$ for all $0 \leq j$. Hence, $\langle Z_s \rangle \subsetneq \langle Z_{s+1} \rangle$ contradicts by the fact. This means that $\langle F_l \rangle \subsetneq \langle F_{l+1} \rangle$ is also contradiction. Therefore, `CGMainW` terminates.

Next we show that, if $(Z, h, G) \in H$, then the triple (Z, h, G) forms a segment of a comprehensive Gröbner system for $\langle F \rangle$, i.e., $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ for each $\bar{a} \in \mathbb{V}(Z) \backslash \mathbb{V}(h)$. This is directly consequence of Lemma 8.1.6 and Lemma 8.1.7. Finally, we have to prove that the conditions in H covers the whole L^m, i.e., $L^m = \bigcup_{(P,h,G) \in H} \mathbb{V}(P) \backslash \mathbb{V}(h)$.
In the algorithm, if the first "`if`" of `CGSW` is true, then the output is $\{(\emptyset, \{1\}, G)\}$. The condition is $\mathbb{V}(\emptyset) \backslash \mathbb{V}(1) = L^m$. If the second "`if`" of `CGSW` is false, then the output is $\{(\emptyset, \{1\}, G)\}$. The condition is L^m. If the second "`if`" of `CGSW` is true, then we have to consider $\{(\emptyset, h, G)\} \cup$ `CGSMainW`$(G \cup \{h_1\}, \{h_1\}) \cup \cdots \cup$ `CGSMainW`$(G \cup \{h_l\}, \{h_l\})$. Let us consider a subalgorithm `CGSMainW`. We assume that one of inputs of `CGSMainW` is (F, Z) where $F \subset K[\bar{A}][\bar{X}, \bar{D}]$ and $Z \subset K[\bar{A}]$. Let G' be a Gröbner basis of $\langle F \rangle$ with respect to $\succ_{\{\bar{X}, \bar{D}\}, \bar{A}}$ and let $h' = h'_1 \cdots h'_l$ in $K[\bar{A}]$. Then, the following equation always holds.

$$\mathbb{V}(Z) = (\mathbb{V}(Z) \backslash \mathbb{V}(h')) \cup \bigcup_{i=1}^{l} \mathbb{V}(Z \cup h'_i).$$

The equation above follows by the induction on the well-founded tree of the algorithm. Therefore, the condition of $\{(\emptyset, h', G)\} \cup$ `CGSMainW`$(G \cup \{h'_1\}, \{h'_1\}) \cup \cdots \cup$ `CGSMainW`$(G \cup \{h'_l\}, \{h'_l\})$ is L^m. $\square$

The algorithm CGSW has been implemented by the author in the computer algebra system Risa/Asir. We give some examples of comprehensive Gröbner systems in the following.

Example 8.1.11. Let $F = \{2a\partial_1 + bx_2\partial_2 + x_2, x_1\partial_1 + ax_2\}$ be a subset of $\mathbb{Q}[a, b][x_1, x_2, \partial_1, \partial_2]$, a, b parameters and x_1, x_2 variables, and ∂_1, ∂_2 partial derivatives by x_1, x_2, respectively. Let $\succ$ be the graded lexicographic orders such that $x_1 \succ x_2 \succ \partial_1 \succ \partial_2$. We compute a comprehensive Gröbner basis for $\langle F \rangle$ with respect to $\succ$.

1: We compute a Gröbner basis for $\langle F \rangle$ in $\mathbb{Q}[a, b, x_1, x_2, \partial_1, \partial_2]$ with respect to the block order with $x_1 \succ x_2 \succ \partial_1 \succ \partial_2 \gg a \succ b$. Then the reduced Gröbner basis is $F_1 = \{ab^2, a^2b, -a\partial_1, ax_2 - ab, bx_2\partial_2 + x_2, x_1\partial_1 + ab\}$. By Lemma 8.1.7, if all elements of $\{a, b\}$ is not zero under specialization σ, then $\sigma(F_1)$ is a Gröbner basis for $\langle \sigma(F) \rangle$. Moreover, if $ab \neq 0$, then ab^2, a^2b become constants. Hence, in this case the Gröbner basis is $\{1\}$. Therefore, $(\{0\}, \{ab\}, \{1\})$ is one of the segments.

2-1: We consider the case $\{a = 0\}$. In this case we have to compute the reduced Gröbner basis for $\langle F \cup \{a\} \rangle$ with respect to $x_1 \succ x_2 \succ \partial_1 \succ \partial_2 \gg a \succ b$. Then the reduced Gröbner basis is $F_2 = \{x_1\partial_1, bx_2\partial_2 + x_2, a\}$. By $a = 0$, we can eliminate a from F_2. Therefore, $(\{a\}, \{b\}, F_2 \backslash \{a\})$ is one of the segments.

2-2 Next we consider the case $\{a = 0, b = 0\}$. Then the reduced Gröbner basis for $\langle F \cup \{a, b\} \rangle$ is $F_3 = \{x_1\partial_1, x_2, a, b\}$. We eliminate $\{a, b\}$ from F_3. Then $(\{a, b\}, \{1\}, F_3 \backslash \{a, b\})$ is one of the segments.

3-1 We consider the case $\{b = 0\}$. Then, the reduced Gröbner basis for $\langle F \cup \{b\}\rangle$ is $F_4 = \{x_1\partial_1, x_2, a\partial_1, b\}$. Therefore, one of the segment is $(\{b\}, \{a\}, F_4\backslash\{b\})$.

3-2 Next we have to consider the case $\{b = 0, a = 0\}$. However, we already computed this case in 2-2.

By the steps, we obtained a comprehensive Gröbner system which is the following;

$$\big\{(\{0\}, \{ab\}, \{1\}), (\{a\}, \{b\}, F_2\backslash\{a\}), (\{a, b\}, \{1\}, F_3\backslash\{a, b\}), (\{b\}, \{a\}, F_4\backslash\{b\})\big\}.$$

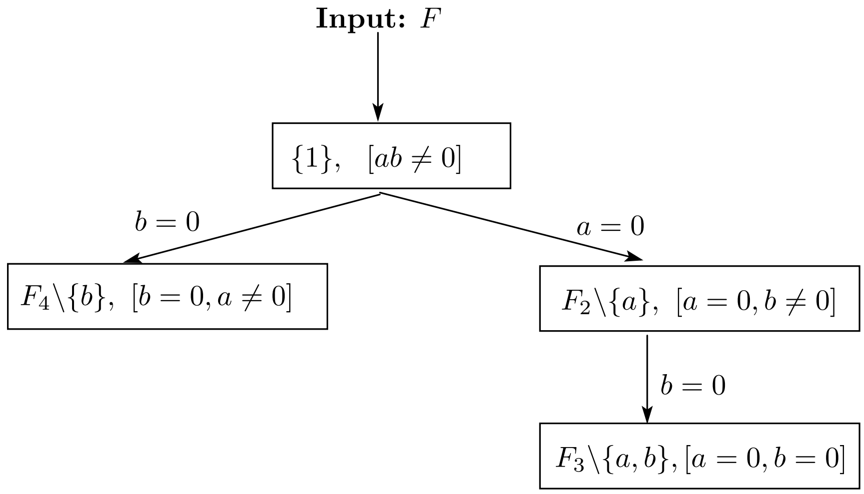

Figure 8.1

Actually, our program outputs the following list as a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$:

```
[b]==0,  [a]!=0,
[x2,d1]
[b,a]==0,  [1]!=0,
[x1*d1,x2]
[a]==0,  [b]!=0,
[x1*d1,b*x2*d2+x2]
[0]==0,  [a*b]!=0,
[1]
```

This output means :

$$\begin{cases} \{x_2, \partial_1\} & \text{if } a \neq 0, b = 0, \\ \{x_1\partial_1, x_2\} & \text{if } a = 0, b = 0, \\ \{x_1\partial_1, bx_2\partial_2 + x_2\} & \text{if } a = 0, b \neq 0, \\ \{1\}, & \text{if } ab \neq 0. \end{cases}$$

Example 8.1.12. Let $F = \{x_1\partial_1 + ax_2\partial_2, bx_1^2\partial_2 + x_2\}$ be a subset of $\mathbb{Q}[a, b][x_1, x_2, \partial_1, \partial_2]$, x_1, x_2 variables, a, b parameters and ∂_1, ∂_2 the partial derivatives by x_1 and x_2 respectively (i.e., $\partial_i x_i = x_i\partial_i + 1,\ i = 1, 2$). Let $\succ$ be the graded lexicographic order such that $x_1 \succ x_2 \succ \partial_1 \succ \partial_2$. We compute a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$.

1: We compute a Gröbner basis for $\langle F\rangle$ in $\mathbb{Q}[a, b, x_1, x_2, \partial_1, \partial_2]$ with respect to the block order with $x_1 \succ x_2 \succ \partial_1 \succ \partial_2 \gg a \succ b$. Then the reduced Gröbner basis F_1 is the following

$F_1 = \{ -ba^3 - 2ba^2 + ba + 2b, (-a+1)x_2, (ba^2+ba-2b)x_1, -x_2\partial_2 - x\partial_1 + a - 1, (ba - b)x_1^2, bx_1^2\partial_2 + x_2, bx_1x_2\partial_2^2 - bax\partial_2 - x_2\partial_1, -bx_2^2\partial_2^3 + (-ba^2 - 3ba + 4b)\partial_2 - x_2\partial_1^2, -1\}$.

The set of every leading coefficient of F_1 is

$$\mathrm{lc}_{\{a,b\}}(F_1) = \{ -ba^3 - 2ba^2 + ba + 2b, -a+1, ba^2+ba-2b, ba-b, b\}.$$

If all element of $\mathrm{lc}_{\{a,b\}}(F_1)$ is not zero under specialization σ_α where $\alpha \in \mathbb{Q}^2$, then $\sigma_\alpha(F_1)$ is a Gröbner basis for $\langle \sigma_\alpha(F)\rangle$. Therefore, $(\{0\}, \mathrm{lc}_{\{a,b\}}(F_1), F_1)$ is one of the segments of a comprehensive Gröbner system for $\langle F\rangle$. In the segment $(\{0\}, \mathrm{lc}_{\{a,b\}}(F_1), F_1)$, we can easily simplify $\mathrm{lc}_{\{a,b\}}(F_1)$ and F_1. Since $-ba^3-2ba^2+ba+2b = -(b)(a-1)(a+1)(a+2)$, $ba^2+ba-2b = b(a+2)(a-1)$, $ba-b = b(a-1)$ and $-1 \neq 0$, we can change $\mathrm{lc}_{\{a,b\}}(F_1)$ into $\{b, a-1, a+1, a+2\}$. Namely, if $b(a-1)(a+1)(a+2) \neq 0$, F_1 is a Gröbner basis under specialization σ_α where $\alpha \in \mathbb{Q}^2 \setminus \mathbb{V}(b(a-1)(a+1)(a+2))$. Look at $-ba^3-2ba^2+ba+2b$ in F_1. If $b(a-1)(a+1)(a+2) \neq 0$, $-ba^3 - 2ba^2 + ba + 2b$ is always a constant under the specialization σ_α. Hence, the Gröbner basis is $\{1\}$. Therefore, one of the segments is $(\{0\}, \{b(a-1)(a+1)(a+2)\}, \{1\})$.

2: We consider the case $\{b = 0\}$. Then the reduced Gröbner basis F_2 for $\langle F \cup \{b\}\rangle$ with respect to the block order with $x_1 \succ x_2 \succ \partial_1 \succ \partial_2 \gg a \succ b$ is the following

$$F_2 = \{b, x_2, x_1\partial_1 - a\}.$$

The set of non-constant leading coefficient of $F_2\backslash\{b\}$ is empty. Therefore we have a segment $(\{b\}, \{1\}, \{x_2, x_1\partial_1 - a\})$.

3-1: We consider the case $\{a+1=0\}$. Then the reduced Gröbner basis is

$$F_3 = \{a+1, x_2, bx_1, x_1\partial_1 + 1\}.$$

The set of non-constant leading coefficient of $F_3\backslash\{a+1\}$ is $\{b\}$.
One of the segments is $(\{a+1\}, \{b\}, \{x_2, bx_1, x_1\partial_1 + 1\})$.

3-2: Next we have to consider the case $\{a+1=0, b=0\}$. Then the reduced Gröbner basis is

$$F_4 = \{b, a+1, x_2, x_1\partial_1 + 1\}.$$

The set of non-constant leading coefficient of $F_3\backslash\{a+1, b\}$ is empty. One of the segments is $(\{a+1, b\}, \{1\}, \{x_2, x_1\partial_1 + 1\})$.

4-1: We consider the case $a+2=0$. Then the reduced Gröbner basis is

$$F_5 = \{a+2, x_2, x_1\partial_1 + 2, bx_1^2\}.$$

The set of non-constant leading coefficient of $F_3\backslash\{a+2\}$ is $\{b\}$.
One of the segments is $(\{a+2\}, \{b\}, \{x_2, x_1\partial_1 + 2, bx_1^2\})$.

4-2: We have to consider the case $\{a+2=0, b=0\}$. Then the reduced Gröbner basis is

$$F_6 = \{b, a+2, x_1, x_1\partial_1 + 2\}.$$

One of the segments is $(\{a+2,b\},\{1\},\{x_2,x_1\partial_1+2\})$.

5-1: We consider the case $\{a+1=0\}$. Then the reduced Gröbner basis is

$$F_7=\{a-1,x_2\partial_2+x\partial_1,bx_1^2\partial_2+x_2,-bx_1x_2\partial_2^2+bx_1\partial_2+y\partial_1,-bx_2^2\partial_2^3-x_2\partial_1^2\}.$$

The set of non-constant leading coefficient of $F_3\backslash\{a-1\}$ is $\{b\}$.
One of the segments is $(\{a-1\},\{b\},\{F_7\backslash\{a-1\}\})$.

5-2: We consider the case $\{a+1=0,b=0\}$. Then the reduced Gröbner basis is

$$F_8=\{b,a-1,x_2,x_1\partial_1-1\}.$$

One of the segments is $(\{a-1,b\},\{1\},\{x_2,\partial_1-1\})$.

Solution : By the steps, a comprehensive Gröbner system for $\langle F\rangle$ is
$\Big\{(\{0\},\{b(a-1)(a+1)(a+2)\},\{1\}),(\{b\},\{1\},\{x_2,x_1\partial_1-a\}),(\{a+1\},\{b\},\{x_1,x_2\}),$
$(\{a+1,b\},\{1\},\{x_2,x_1\partial_1+1\}),(\{a+2\},\{b\},\{x_2,-x_1\partial_1-2,x_1^2\}),(\{a+2,b\},\{1\},\{x_2,x_1\partial_1+2\}),$
$(\{a-1\},\{b\},\{F_7\backslash\{a-1\}\}),(\{a-1,b\},\{1\},\{x_2,x_1\partial_1-1\})\Big\}.$
We can see this comprehensive Gröbner system as follows

$$\begin{cases}\{1\} & \text{if } b(a-1)(a+1)(a+2)\neq 0,\\ \{x_2,x_1\partial_1-a\} & \text{if } b=0,\\ \{x_2,bx_1,x_1\partial_1+1\} & \text{if } a+1=0,b\neq 0\\ \{x_2,x_1\partial_1+1\} & \text{if } a+1=b=0,\\ \{x_2,x_1\partial_1+2,bx_1^2\} & \text{if } a+2=0,b\neq 0,\\ \{x_2,x_1\partial_1+2\} & \text{if } a+2=b=0,\\ \{F_7\backslash\{a-1\}\} & \text{if } a-1=0,b\neq 0,\\ \{x_2,x_1\partial_1-1\} & \text{if } a-1=b=0.\end{cases}$$

Remark: Actually, the segment $(\{b\},\{1\},\{x_2,\partial_1-a\})$ includes the segments $(\{a+1,b\},\{1\},\{x_2,x\partial_1+1\})$, $(\{a+2,b\},\{1\},\{x_2,x_1\partial_1+2\})$ and $(\{a-1,b\},\{1\},\{x_2,x_1\partial_1-1\})$. Therefore, we can eliminate these segments from the list. We are able to compares each condition of the parameters in the procedure. This is one of the optimization techniques to get small and nice outputs comprehensive Gröbner systems. (In our program this techniques has been implemented.) In this example, we followed the algorithm CGSW.

Example 8.1.13. Let $F=\{2\partial_1+bx_2\partial_2+x_2,x_1\partial_1-ax_2\}\subset\mathbb{Q}[a,b][x_1,x_2,\partial_1,\partial_2]$, a,b parameters, x_1,x_2 variables and ∂_1,∂_2 partial derivative by x_1,x_2, respectively. Let $\succ$ be the graded lexicographic order such that $x_1\succ x_2\succ\partial_1\succ\partial_2$. Then our program outputs the following list as a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$:

```
[a]==0,  [b]!=0,
[b*x2*d2+x2,d1]
[b]==0,  [1]!=0,
[x2,d1]
[0]==0,  [b*a]!=0,
[1]
```

This meaning is the following:

$$\begin{cases} \{bx_2\partial_2 + x_2, \partial_1\} & \text{if } a = 0, b \neq 0, \\ \{x_2, \partial_1\} & \text{if } b = 0, \\ \{1\}, & \text{if } ab \neq 0. \end{cases}$$

Example 8.1.14. Let $F = \{2x_1d_1 + ax_2d_2 + 1, x_2d_2 + 2ax_3d_3 - 1, cd_1d_3 - bx_2^2\} \subset \mathbb{Q}[a,b,c][x_1,x_2,x_3,d_1,d_2,d_3]$, a,b,c parameters, x_1,x_2,x_3 variables and d_1,d_2,d_3 partial derivative by x_1,x_2,x_3, respectively. Let $\succ$ be the lexicographic order such that $d_3 \succ d_2 \succ d_1 \succ x_3 \succ x_2 \succ x_1$. Then our program outputs the following list as a comprehensive Gröbner system for $\langle F\rangle$ with respect to $\succ$:

```
[2*a-1]==0,  [b*c]!=0,
[x3*d3-3,x2*d2+2,d1,x2^2]
[a+1]==0,  [b]!=0,
[c*d1*d3-b*x2^2,x3*d3-x1*d1,-x2*d2+2*x1*d1+1,-c*x1*d1^2-c*d1+b*x2^2*x3]
[a-1]==0,  [c*b]!=0,
[x3*d3-1,d1,x2]
[b]==0,  [c]!=0,
[d1*d3,-2*a^2*x3*d3+2*x1*d1+a+1,a*x2*d2+2*x1*d1+1,-2*x1*d1^2+(-a-3)*d1]
[b,c]==0,  [1]!=0,
[2*a^2*x3*d3-2*x1*d1-a-1,a*x2*d2+2*x1*d1+1]
[c]==0,  [b]!=0,
[2*a^2*x3*d3-2*x1*d1-a-1,a*x2*d2+2*x1*d1+1,-4*x1^2*d1^2+(6*a-8)*x1*d1-2*a^
 2+3*a-1,2*x1*x2*d1+(-2*a+1)*x2,x2^2]
[0]==0,  [c*b*(a-1)*(a+1)*(2*a-1)]!=0,
[1]
```

This meaning is the following:

$$\begin{cases} \left\{x_3d_3 - 3, x_2d_2 + 2, d_1, x_2^2\right\} & \text{if } 2a - 1 = 0, bc \neq 0 \\ \left\{cd_1d_3 - bx_2^2, x_3d_3 - x_1d_1, -x_2d_2 + 2x_1d_1 + 1,\right. & \text{if } a + 1 = 0, b \neq 0 \\ \left. -cx_1d_1^2 - cd_1 + bx_2^2x_3\right\} & \\ \left\{x_3d_3 - 1, d_1, x_2\right\} & \text{if } a - 1 = 0, bc \neq 0 \\ \left\{d_1d_3, -2a^2x_3d_3 + 2x_1d_1 + a + 1,\right. & \text{if } b = 0, c \neq 0 \\ \left. ax_2d_2 + 2x_1d_1 + 1, -2x_1d_1^2 + (-a-3)d_1\right\} & \\ \left\{2a^2x_3d_3 - 2x_1d_1 - a - 1, ax_2d_2 + 2x_1d_1 + 1\right\}, & \text{if } b = c = 0, \\ \left\{2a^2x_3d_3 - 2x_1d_1 - a - 1, ax_2d_2 + 2x_1d_1 + 1,\right. & \text{if } c = 0, b \neq 0, \\ -4x_1^2d_1^2 + (6a-8)x_1d_1 - 2a^2 + 3a - 1, & \\ \left. 2x_1x_2d_1 + (-2a+1)x_2, x_2^2\right\} & \\ \left\{1\right\} & \text{if } cb(a-1)(a+1)(2a-1) \neq 0. \end{cases}$$

8.1.3 Comprehensive Gröbner bases

Here we present an algorithm for computing comprehensive Gröbner bases. In fact, we modify the algorithm `CGSW` to compute comprehensive Gröbner bases.

Definition 8.1.15. Let F and G be subsets of $K[\bar{A}][\bar{X},\bar{D}]$. $G \subset \langle F\rangle$ is called a **comprehensive Gröbner basis** for $\langle F\rangle$ if $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(F)\rangle$ for each $\bar{a} \in L^m$.

We already saw comprehensive Gröbner systems in previous section which has conditions of parameters. However, comprehensive Gröbner bases do not have conditions of parameters. A comprehensive Gröbner basis is a set of differential operators. In this point, comprehensive Gröbner bases are different from comprehensive Gröbner systems. In this section we consider an algorithm for computing comprehensive Gröbner bases. Then, we need the following concept (which we also saw in chapter 4).

Definition 8.1.16. Let $F \subset K[\bar{A}][\bar{X}, \bar{D}]$, $s_1, \ldots, s_l, t_1, \ldots, t_l \subset K[\bar{A}]$ and $G_1, \ldots, G_i \subset K[\bar{A}][\bar{X}, \bar{D}]$. Then a comprehensive Gröbner system $\{(s_1, t_1, G_1), \ldots, (s_l, t_l, G_l)\}$ for $\langle F \rangle$ is called **faithful** if $G_i \subset \langle F \rangle$ for each $i = 1, \ldots, l$.

Actually, we describe an algorithm for computing **faithful** comprehensive Gröbner systems. If $\{(s_1, t_1, G_1), \ldots, (s_l, t_l, G_l)\}$ is a faithful comprehensive Gröbner system for $\langle F \rangle$, then by the definitions, $G_1 \cup \cdots \cup G_l$ is a comprehensive Gröbner basis for $\langle F \rangle$. Therefore, we modify the algorithm CGSMainW (or CGSW) to compute a faithful comprehensive Gröbner system. The key ideal which is from [SS06] is to apply a new variable U. In [SS06], they introduced a new auxiliary variable U besides $\bar{X}$ and $\bar{A}$ in order to compute comprehensive Gröbner bases. We follow this technique to compute comprehensive Gröbner bases in rings of differential operators. We define homomorphisms σ^0 and σ^1 from $K[\bar{A}][U, \bar{X}, \bar{D}]$ to $K[\bar{A}][\bar{X}, \bar{D}]$ as a specialization of U with 0 and 1 respectively, i.e. $\sigma^0(f(U, \bar{A}, \bar{X}, \bar{D})) = f(0, \bar{A}, \bar{X}, \bar{D})$ and $\sigma^1(f(U, \bar{A}, \bar{X}, \bar{D})) = f(1, \bar{A}, \bar{X}, \bar{D})$. Before we introduce the algorithm for computing comprehensive Gröbner bases, we need the following lemma.

Lemma 8.1.17 ([SS06]). Let F and S be subsets of $K[\bar{A}][\bar{X}, \bar{D}]$. For any $g \in \langle (U \cdot F) \cup (U-1) \cdot S) \rangle \subseteq K[\bar{A}][U, \bar{X}, \bar{D}]$, then $\sigma^0(g) \in \langle S \rangle \subseteq K[\bar{A}][\bar{X}, \bar{D}]$ and $\sigma^1(g) \in \langle F \rangle \subseteq K[\bar{A}][\bar{X}, \bar{D}]$.

The next theorem is the main result of this section. By the following theorem, we can construct an algorithm for computing comprehensive Gröbner bases.

Theorem 8.1.18. Let F be a subset of $K[\bar{A}][\bar{X}, \bar{D}]$, and S a subset of $K[\bar{A}]$. Furthermore, let G be a Gröbner basis for $\langle (U \cdot F) \cup (U-1) \cdot S \rangle$ in $K[\bar{A}][U, \bar{X}, \bar{D}]$ with respect to a block order $\succ_{U, \{\bar{X}, \bar{D}\}} := (\succ_1, \succ_2)$ with $U \gg \{\bar{X}, \bar{D}\}$. Suppose that $B_1 := \{g \in G | \deg_{\{\bar{X}, \bar{D}\}}(\mathrm{lpp}_{\bar{A}}(g)) = 0, \mathrm{lc}_{\bar{A}}(g) \in \langle S \rangle\}$, $B_2 := \{g \in G | \deg_U(\mathrm{lpp}_{\bar{A}}(g)) = 0\}$, $G' = G \backslash (B_1 \cup B_2)$, $\{h_1, \ldots, h_l\} := \{\mathrm{lc}_{\bar{A}}(g) | g \in G'\}$ and $h = \mathsf{LCM}(h_1, \ldots, h_l)$. Then for each $\bar{a} \in \mathbb{V}(S) \backslash \mathbb{V}(h)$, $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}, \bar{D}]$. Actually, we have $\sigma_{\bar{a}}(\sigma^1(G)) = \sigma_{\bar{a}}(\sigma^1(G'))$.

Proof. Note that any differential operator of G have a linear form of U, i.e., the degree of U is at most 1. By Lemma 8.1.17, $\sigma^1(G)$ is a basis of $\langle F \rangle$. We prove that $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$. By the sets G', B_1 and B_2, we have $G = G' \cup B_1 \cup B_2$. We consider three cases B_1, B_2 and G'. Take for all $f_1 \in B_1$. Then f_1 can be written as $f_1 = f_{11}U + f_{12}$ where $f_{11}, f_{12} \in K[\bar{A}]$ and $f_{11} \in \langle S \rangle$. By Lemma 8.1.17, $\sigma^0(f_1) = f_{12} \in \langle S \rangle$, thus we have $\sigma_{\bar{a}}(f_{12}) = 0$. Hence, $\sigma_{\bar{a}}(f_1) = 0$. Take for all $f_2 \in B_2$. Then, by Lemma 8.1.17, $\sigma^0(f_2) = f_2 \in \langle S \rangle$. Hence, $\sigma_{\bar{a}}(f_2) = 0$. By Lemma 8.1.7, we have $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G')$ which is a Gröbner basis for $\langle U \cdot F \cup (U-1) \cdot S \rangle$ in $L[\bar{X}, U, \bar{D}]$. Take for all $g \in G'$. Then, g can be written as $g = U \cdot g_1 + g_2$ where $g_1, g_2 \in K[\bar{A}][\bar{X}, \bar{D}]$. By Lemma 8.1.17, we have $\sigma^0(g) = g_2 \in \langle S \rangle$, thus $\sigma_{\bar{a}}(g_2) = 0$. That is, $\sigma_{\bar{a}}(g) = \sigma_{\bar{a}}(U \cdot g_1)$. Since every power product of $\sigma_{\bar{a}}(G')$ has a variable U whose degree is 1 and $U \gg \{\bar{X}, \bar{D}\}$, $\sigma^1(\sigma_{\bar{a}}(G'))$ is a Gröbner basis of $\langle \sigma^1(\sigma_{\bar{a}}(U \cdot F) \cup (U-1) \cdot S) \rangle = \langle \sigma^1(\sigma_{\bar{a}}(U \cdot F)) \rangle = \langle \sigma_{\bar{a}}(F) \rangle$. Therefore, it follows that $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}, \bar{D}]$. □

Theorem 8.1.18 leads us to have the following algorithm which outputs a comprehensive Gröbner bases. In fact, the algorithm `CGBMainW` returns a faithful comprehensive Gröbner system.

Algorithm 8.1.19. $\mathsf{CGBW}(F, \succ_1)$ (Comprehensive Gröbner Bases in Weyl algebra)

Input F : a finite subset of $K[\bar{A}][\bar{X},\bar{D}]$,
$\succ_1$: a term order on $\mathrm{pp}(\bar{X},\bar{D})$,
Output G: a comprehensive Gröbner basis for $\langle F\rangle$ with respect to $\succ_1$.
begin
$G \leftarrow \emptyset$
$Z \leftarrow \emptyset$
$H \leftarrow \mathsf{CGBMainW}(F, Z, \succ_1)$
while $H \neq \emptyset$ **do**
Select (s,t,E) from H; $H \leftarrow H\backslash\{(s,t,E)\}$
$G \leftarrow G \cup E$
end-while
return(G)
end

Algorithm 8.1.20. $\mathsf{CGBMainW}(F, Z, \succ_1)$

Input F : a finite subset of $K[\bar{A}][\bar{X},\bar{D}]$,
Z : a finite subset of $K[\bar{A}]$,
$\succ_1$: a term order on $\mathrm{pp}(\bar{X},\bar{D})$,
$\Big(\succ_{U,\{\bar{X},\bar{D}\},\bar{A}} = (\succ,\succ_1,\succ_2)$: a block order such that $U \gg \{\bar{X},\bar{D}\} \gg \bar{A}$ on $\mathrm{pp}(U,\bar{X},\bar{D},\bar{A})$,
$\succ_2$: a term order on $\mathrm{pp}(\bar{A})$, $\succ$: a term order on $\mathrm{pp}(U)$,$\Big)$
Output G: a faithful comprehensive Gröbner system on $\mathbb{V}(S)$ for $\langle F\rangle$.
begin
if $1 \in \langle Z\rangle$ **then**
$G \leftarrow \emptyset$
else
$H \leftarrow \mathsf{RGr\ddot{o}bnerBasisW}((U\cdot F)\cup((U-1)\cdot Z), \succ_{U,\{\bar{X},\bar{D}\},\bar{A}})$
$B_1 \leftarrow \{g \in H \mid \deg_U(\mathrm{lpp}_{\bar{A}}(g)) = 0\}$
$B_2 \leftarrow \{g \in H \mid \deg_{\{\bar{X},\bar{D}\}}(\mathrm{lpp}_{\bar{A}}(g)) = 0, \mathrm{lc}_{\bar{A}}(g) \in \langle Z\rangle\}$
$H' \leftarrow H\backslash(B_1 \cup B_2)$
$S \leftarrow \{h_1,\ldots,h_l\} := \{\mathrm{lc}_{\bar{A}}(g) \mid g \in H'\}$
if $S \neq \emptyset$ **then**
$h \leftarrow \mathsf{LCM}(h_1,\ldots,h_l)$
$G \leftarrow \{(Z,\{h\},\sigma^1(H')\}\cup \mathtt{CGBMainW}(F, Z\cup\{h_1\},\succ_1)\cup\cdots\cup \mathtt{CGBMainW}(F, Z\cup\{h_1\},\succ_1)$
else
$G \leftarrow \{(Z,\{1\},\sigma^1(H')\}$
end-if
end-if
`return`(G)
end

Theorem 8.1.21. Let F be a finite subset of $K[\bar{A}][\bar{X},\bar{D}]$. Then the algorithm CGBW returns a comprehensive Gröbner bases for $\langle F\rangle$. That is, the algorithm CGBMainW returns a faithful comprehensive Gröbner system for $\langle F\rangle$.

Proof. Since the algorithm $\mathsf{RGr\ddot{o}bnerBasisW}(F)$ outputs the reduced Gröbner basis for $\langle F\rangle$

in $K[\bar{A}][U,\bar{X},\bar{D}]$, $h_i \in K[\bar{A}]$ is not in the ideal $\langle Z\rangle$. We modified the algorithm CGSW and CGSMainW for constructing the algorithm CGBW and CGBMainW above. Hence, the algorithm CGBMain terminates, and by Theorem 8.1.18 and Lemma 8.1.7, returns a faithful comprehensive Gröbner system. Therefore, the algorithm CGBW terminates and returns a comprehensive Gröbner basis. □

This algorithm CGBW have been implemented by the author in the computer algebra system Risa/Asir. In the following examples, we can see some examples of comprehensive Gröbner bases and how the algorithm works.

Example 8.1.22. Let $F = \{2a\partial_1 + bx_2\partial_2 + x_2, x_1\partial_1 + ax_2\}$ be a subset of $\mathbb{Q}[a,b][x_1,x_2,\partial_1,\partial_2]$, a,b parameters and x_1,x_2 variables, and ∂_1,∂_2 partial derivatives by x_1, x_2, respectively. Let $\succ$ be the graded lexicographic order with $x_1 \succ x_2 \succ \partial_1 \succ \partial_2$. We compute a comprehensive Gröbner basis for $\langle F\rangle$ with respect to $\succ$.

- We compute a faithful comprehensive Gröbner system for $\langle F\rangle$.

1: We compute a Gröbner basis for $\langle F\rangle$ in $\mathbb{Q}[a,b,x_1,x_2,\partial_1,\partial_2]$ with respect to the block order with $x_1 \succ x_2 \succ \partial_1 \succ \partial_2 \gg a \succ b$. Then The reduced Gröbner basis is the following

$$F_1 = \{ab^2, a^2b, -a\partial_1, ax_2 - ab, bx_2\partial_2 + x_2, x_1\partial_1 + ab\}.$$

By Lemma 4.4.2, if all elements of $\{a,b\}$ is not zero under specialization σ, then $\sigma(F_1)$ is a Gröbner basis for $\langle\sigma(F)\rangle$. Therefore, $(\{0\},\{ab\},F_1)$ is one of the segments of a faithful comprehensive Gröbner basis.

2-1: Let u be a new variable. We consider the case $\{a = 0\}$. In this case we have to compute the reduced Gröbner basis for $\langle u\cdot F \cup (u-1)\cdot(a)\rangle$ with respect to $u \gg x_1 \succ x_2 \succ \partial_1 \succ \partial_2 \gg a \succ b$. Then the reduced Gröbner basis is

$$F_2 = \{ab^2, a^2b, -a\partial_1, ax_2 - ab, au - a, bux_2\partial_2 + ux_2, ux_1\partial_1 + ab\}.$$

In case $\{a = 0\}$, $ab^2, a^2b, -a\partial_1, ax_2 - ab, au - a$ are always zero. Hence, we eliminate them from F_2. Next we substitute 1 into the new variable u. Then we have the following set

$$F_2' = \{bx_2\partial_2 + x_2, x_1\partial_1 + ab\}.$$

Therefore, one of the segments is $(\{a\},\{b\},F_2')$.

2-2: Next we consider the case $\{a = 0, b = 0\}$. Then the reduced Gröbner basis for $\langle u\cdot F \cup \{(u-1)\cdot a, (u-1)\cdot b\}\rangle$ is

$$F_3 = \{b, a\partial_1, ax_2, au - a, ux_2, ux_1\partial_1\}.$$

In case $\{a = 0, b = 0\}$, $b, a\partial_1, ax_2, au - a$ are always zero, so we eliminate them from F_3. Next we substitute 1 into the new variable u. Then we have the following set

$$F_3' = \{x_2, x_1\partial_1\}.$$

Therefore, one of the segments is $(\{a,b\},\{1\},F_3')$.

3-1: We consider the case $b = 0$. Then the reduced Gröbner basis for $\langle u\cdot F \cup \{(u-1)\cdot b\}\rangle$ is

$$F_4 = \{ab - b, bu, au\partial_1, ux_2, ux_1\partial_1\}.$$

In case $\{b = 0\}$, $ab - b, bu$ is always zero, so we eliminate them from F_4. We substitute 1 into the new variable u. Then we have the following set

$$F_4' = \{a\partial_1, x_2, x_1\partial_1\}.$$

One of the segments is $(\{b\}, \{a\}, F_4')$.

3-2: Next we have to consider the case $\{b = 0, a = 0\}$. However, we already computed this case in 2-2.

By the steps, we obtained a **faithful comprehensive Gröbner system**

$$\big\{(\{0\}, \{ab\}, F_1), (\{a\}, \{b\}, F_2'), (\{a, b\}, \{1\}, F_3'), (\{b\}, \{a\}, F_4')\big\}.$$

Therefore, a comprehensive Gröbner basis for $\langle F \rangle$ is

$$F_1 \cup F_2' \cup F_3' \cup F_4' = \big\{x_1\partial_1 + ab, bx_2\partial_2 + x_2, -a\partial_1, ab^2, a^2b, ax_2 - ab\big\}.$$

See Figure 8.2.

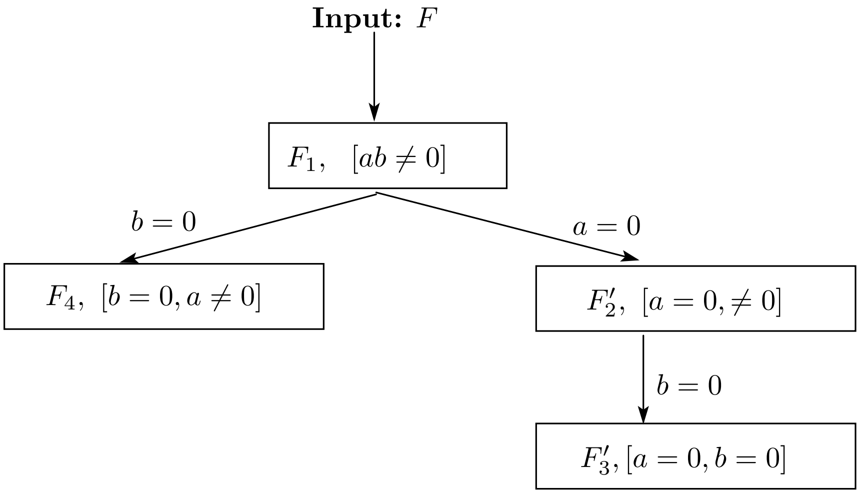

Figure 8.2

Example 8.1.23. Let $F_1 = \{2x_1^2d_1 + ax_2d_2, x_1x_2d_2 + 2x_1^2, bx_2\}$, $F_2 = \{2d_1 + ad_2 + 1, 2x_3d_2 - 1, cd_1d_3 - bx_2^2d_1\}$ be subsets of $\mathbb{Q}[a, b, c][x_1, x_2, x_3, d_1, d_2, d_3]$, a, b, c parameters and x_1, x_2, x_3 variables, and d_1, d_2, d_3 partial derivatives by x_1, x_2, x_3, respectively. Let $\succ$ be the lexicographic order such that $x1 \succ x_2 \succ x_3 \succ d_1 \succ d_2 \succ d_3$. Then our program outputs the following list as a comprehensive Gröbner basis for $\langle F_1 \rangle$ with respect to $\succ$:

```
[a*x2^2*d1*d2^2+(a*x2*d1-4*a*x2)*d2+16*x1,a*x2^3*d2^3+3*a*x2^2*d2^2+a*x2*
d2,b*a,b*x2,(a^2+3*a)*x2^2*d2^2+(a^2+3*a)*x2*d2]
```

Our program outputs the following list as a comprehensive Gröbner basis for $\langle F_2 \rangle$ with respect to $\succ$:

```
[2*x3*d2-1,2*c*x3+c*a,a*d2+2*d1+1,-c*a*d2-c,-a*b*d2-b,2*b*x3+a*b,a*d2+2*d
1+1,2*x3*d2-1].
```

Example 8.1.24. Let $F_1 = \{2x_1^2d_1 + ax_2d_2, x_1x_2d_2 + 2x_1^2, bx_2\}$, $F_2 = \{2x_1d_1 + ax_2d_2 + 1, x_2d_2 + 2ax_3d_3 - 1, cd_1d_3 - bx_2^2\}$ be subsets of $\mathbb{Q}[a, b, c][x_1, x_2, x_3, d_1, d_2, d_3]$, a, b, c parameters and x_1, x_2, x_3 variables, and d_1, d_2, d_3 partial derivatives by x_1, x_2, x_3, respectively.

Let $\succ$ be the lexicographic order such that $x1 \succ x_2 \succ x_3 \succ d_1 \succ d_2 \succ d_3$.
Then our program outputs the following list as a comprehensive Gröbner basis for $\langle F_1 \rangle$ with respect to $\succ$:

```
[b*a,(a^2+3*a)*x2^2*d2^2+(a^2+3*a)*x2*d2,a*x2^3*d2^3+3*a*x2^2*d2^2+a*x2*d2
 ,a*x2^2*d1*d
 2^2+(a*x2*d1-4*a*x2)*d2+16*x1,b*x2].
```

Our program outputs the following list as a comprehensive Gröbner basis for $\langle F_2 \rangle$ with respect to $\succ$:

```
[-2*c*b*a^3+c*b*a^2+2*c*b*a-c*b,4*b*a^2*x3^2*d3^2+(-c*d1*d2^2+(4*b*a^2-10
 *b*a)*x3)*d3+6*b,2*a*x3*d3+x2*d2-1,(c*d1*d2+2*b*a*x2*x3)*d3-3*b*x2,c*d1*
 d3-b*x2^2,-2*a^2*x3*d3+2*x1*d1+a+1,-2*c*a^2*x3*d3^2+(-2*c*a^2+c*a+c)*d3+
 2*b*x1*x2^2,(-2*c*a^3-2*c*a^2)*x3*d3^2+(-2*c*a^3-c*a^2+2*c*a+c)*d3,(c*b*
 a+c*b)*d1,(c*a+c)*d1*d3,(-c*b*a-c*b)*x3*d3-4*c*b*a^2+c*b*a+5*c*b,8*b*a^2
 *x3^2*d3^2+((c*a-c)*d1*d2^2+(8*b*a^2-20*b*a)*x3)*d3+12*b,(2*c*b*a^2+c*b*
 a-c*b)*x2,((-c*a+c)*d1*d2+4*b*a*x2*x3)*d3-6*b*x2,(-c*a+c)*d1*d3-2*b*x2^2
 ,(2*c*b*a+2*c*b)*d1,(2*c*a+2*c)*d1*d3,(-2*c*b*a-2*c*b)*x3*d3-8*c*b*a^2+2
 *c*b*a+10*c*b,(-8*c*a^3-8*c*a^2)*x3*d3^2+(-8*c*a^3-4*c*a^2+8*c*a+4*c)*d3
 ,12*b*a^2*x3^2*d3^2+((2*c*a-c)*d1*d2^2+(12*b*a^2-30*b*a)*x3)*d3+18*b,((-
 2*c*a+c)*d1*d2+6*b*a*x2*x3)*d3-9*b*x2,(-2*c*a+c)*d1*d3-3*b*x2^2,-4*a^2*x
 3*d3+4*x1*d1+2*a+2,4*b*a^2*x3^2*d3^2+(c*a*d1*d2^2+(4*b*a^2-10*b*a)*x3)*d
 3+6*b].
```

8.2 Comprehensive Gröbner bases in $(K[\bar{A}][\bar{X}])[\bar{D}]$

Here we describe comprehensive Gröbner bases in $(K[\bar{A}][\bar{X}])[\bar{D}]$ with coefficients in $K[\bar{A}][\bar{X}]$. In chapter 7, we have seen the relations between Gröbner bases in $K[\bar{X}, \bar{D}]$ and Gröbner bases in $K[\bar{X}][\bar{D}]$. The key idea for computing Gröbner bases in $K[\bar{X}][\bar{D}]$ was a block order such that $\bar{D} \gg \bar{X}$. By this relation and the algorithm CGBW (or CGSW), we can easily compute comprehensive Gröbner bases (or comprehensive Gröbner systems) in $(K[\bar{A}][\bar{X}])[\bar{D}]$.

Theorem 8.2.1. Let I be an ideal in $(K[\bar{A}][\bar{X}])[\bar{D}]$. Since $(K[\bar{A}][\bar{X}])[\bar{D}]$ is isomorphic to $K[\bar{A}][\bar{X}, \bar{D}]$, the ideal I can be seen as an ideal in $K[\bar{A}][\bar{X}, \bar{D}]$ and we write the ideal as I'. Let G be a comprehensive Gröbner system for I' on L^m with respect to a block order $\succ_{\bar{D},\bar{X}} := (\succ_1, \succ_2)$, (i.e., $\bar{D} \gg \bar{X}$) in $K[\bar{A}][\bar{X}, \bar{D}]$. The set G can be seen as a set of triples in $(K[\bar{A}][\bar{X}])[\bar{D}]$ and we write the set as G again. Then G is a comprehensive Gröbner system for I with respect to $\succ_1$ in $(K[\bar{A}][\bar{X}])[\bar{D}]$.

Proof. We take an arbitrary segment $(S_i, T_i, G_i) \in G$. Since G is a comprehensive Gröbner system for I in $K[\bar{A}][\bar{X}, \bar{D}]$ with respect to a block order $\succ_{\bar{D},\bar{X}}$, for any $\bar{a} \in \mathbb{V}(S_i) \backslash \mathbb{V}(T_i)$, $\sigma_{\bar{a}}(G_i)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(I) \rangle$ with respect to $\succ_{\bar{D},\bar{X}}$. By Theorem 7.4.1, $\sigma_{\bar{a}}(G_i)$ is also a Gröbner basis for $\langle \sigma_{\bar{a}}(I) \rangle$ in $L[\bar{X}][\bar{D}]$ with respect to $\succ_1$. This means that G is a comprehensive Gröbner system for I with respect to $\succ_1$ in $(K[\bar{A}][\bar{X}])[\bar{D}]$. $\square$

Theorem 8.2.2. Let I be an ideal in $(K[\bar{A}][\bar{X}])[\bar{D}]$. Since $(K[\bar{A}][\bar{X}])[\bar{D}]$ is isomorphic to $K[\bar{A}][\bar{X}, \bar{D}]$, the ideal I can be seen as an ideal in $K[\bar{A}][\bar{X}, \bar{D}]$ and we write the ideal as I'. Let G be a comprehensive Gröbner basis for I' with respect to a block order $\succ_{\bar{D},\bar{X}} := (\succ_1, \succ_2)$, (i.e., $\bar{D} \gg \bar{X}$) in $K[\bar{A}][\bar{X}, \bar{D}]$. G can be seen as a subset of $(K[\bar{A}][\bar{X}])[\bar{D}]$ and we write the set as G again. Then G is a comprehensive Gröbner basis for I with respect to $\succ_1$ in $(K[\bar{A}][\bar{X}])[\bar{D}]$.

Proof. Since G is a comprehensive Gröbner basis for I with respect to $\succ_{\bar{D},\bar{X}}$ in $K[\bar{A},\bar{X}][\bar{D}]$, for any $\bar{a} \in L^m$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(I) \rangle$ with respect to $\succ_1$. By Theorem 7.4.1, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(I) \rangle$ with respect to $\succ_1$ in $L[\bar{X}][\bar{D}]$. Therefore, G is a comprehensive Gröbner basis for I with respect to $\succ_1$ in $(K[\bar{A}][\bar{X}])[\bar{D}]$. $\square$

Actually, when we compute a comprehensive Gröbner basis (or system) in $(K[\bar{A}][\bar{X}])[\bar{D}]$, we need a Gröbner basis computation with respect to a block order with $\bar{D} \gg \bar{X} \gg \bar{A}$. By using the block order and the algorithm CGBW (or CGSW), we are able to compute a comprehensive Gröbner basis (or system) in $(K[\bar{A}][\bar{X}])[\bar{D}]$.

Corollary 8.2.3. Let F be a subset of $(K[\bar{A}][\bar{X}])[\bar{D}]$. Since $(K[\bar{A}][\bar{X}])[\bar{D}]$ is isomorphic to $k[\bar{A},\bar{X},\bar{D}]$, the set F can be seen as a subset of $K[\bar{A},\bar{X},\bar{D}]$ and we write the subset as F'. Let G be a Gröbner basis for $\langle F' \rangle$ in $K[\bar{A},\bar{X},\bar{D}]$ with respect to $\succ_{\bar{D},\bar{X},\bar{A}} := (\succ_1, \succ_2, \succ_3)$ (i.e., $\bar{D} \gg \bar{X} \gg \bar{A}$). G can be seen as a subset of $(K[\bar{A}][\bar{X}])[\bar{D}]$ and we write the subset as G again. Suppose that $\{h_1, \dots, h_s\} := \{\mathrm{lc}_{\bar{A}}(g) | g \in G \backslash K[\bar{A}]\}$ and $h := \mathsf{LCM}(h_1, \dots, h_s)$. Then, for any $\bar{a} \in \mathbb{V}(G \cap K[\bar{A}]) \backslash \mathbb{V}(h)$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ with respect to $\succ_1$ in $L[\bar{X}][\bar{D}]$.

Proof. By Theorem 8.2.2 and Theorem 7.4.1, this corollary holds. $\square$

In the following examples, we give some examples of comprehensive Gröbner bases and systems in $(\mathbb{C}[\bar{A}][\bar{X}])[\bar{D}]$.

Example 8.2.4. Let $F = \{x_1\partial_1 + ax_2\partial_2, bx_1^2\partial_2 + x_2\}$ be a subset of $(\mathbb{C}[a,b][x_1,x_2])[\partial_1,\partial_2]$, a, b parameters and $\succ$ be the lexicographic order such that $\partial_1 \succ \partial_2$.

(1) : We ready to compute a comprehensive Gröbner **system** for $\langle F \rangle$ with respect to $\succ$. First, we have to compute a comprehensive Gröbner systems for $\langle F \rangle$ in $\mathbb{C}[a,b][x_1,x_2,\partial_1,\partial_2]$ with respect to a block order $\partial_1 \succ \partial_2 \gg x_1 \succ_1 x_2$. In this example $\succ_1$ is the lexicographic order. Second, by the algorithm CGSW, we obtain a comprehensive Gröbner system G as follows:

$$G = \Big\{(\{a+2\},\{b\},\{x_1\partial_1+2, x_1^2, x_2\}), (\{a+1\},\{b\},\{x_1,x_2\}), (\{a-1\}\{b\},\{bx_1^2\partial_2 + x_2, bx_1x_2\partial_2^2 - bx_1\partial_2 - x_2\partial_1, x_2\partial_2 + x_1\partial_1\}), (\{b\},\{1\},\{x_2, x_1\partial_1 - a\}), (\{\emptyset\},\{b(a-1)(a+1)(a+2)\},\{1\})\Big\}.$$

By Theorem 8.2.1, G is also a comprehensive Gröbner system for I in $(\mathbb{C}[a,b][x_1,x_2])[\partial_1,\partial_2]$ with respect to $\succ$.

(2): We ready to compute a comprehensive Gröbner **basis** for $\langle F \rangle$ with respect to $\succ$. First, we have to compute a comprehensive Gröbner basis for $\langle F \rangle$ in $\mathbb{C}[a,b][x_1,x_2,\partial_1,\partial_2]$ with respect to a block order $\partial_1 \succ \partial_2 \gg x_1 \succ_1 x_2$. Second, by the algorithm CGSW, we can obtain a comprehensive Gröbner basis G as follows:

$$G = \Big\{ -a^3b - 2a^2b + ab + 2b, bx_1^2\partial_2 + x_2, -bx_1^2\partial_2^2 + x_1\partial_1 - a, bx_1x_2\partial_2^2 - abx_1\partial_2 - x_2\partial_1, x_2\partial_2 + x_1\partial_1 - a + 1, (a-1)x_2, (a^2b + ab - 2b)x_1, (-a-1)x_2\partial_2 - 2x_1\partial_1 + a - 1, (-ab + b)x_1^2, (-a-2)x_2\partial_2 - 3x_1\partial_1 + 2a - 2\Big\}.$$

By Theorem 8.2.2, G is also a comprehensive Gröbner basis for I in $(\mathbb{C}[a,b][x_1,x_2])[\partial_1,\partial_2]$ with respect to $\succ$.

Example 8.2.5. Let $F = \{2x_1d_1 + ax_2d_2 + 1, bx_2d_2 + 2ax_3d_3, d_1d_3 - x_2^2d_2\}$ be a subset of $(\mathbb{Q}[a,b][x_1,x_2,x_3])[d_1,d_2,d_3]$, a,b parameters and x_1,x_2,x_3 variables, and d_1,d_2,d_3 partial derivatives by x_1,x_2,x_3, respectively. Let $\succ$ be the lexicographic order such that $d_1 \succ d_2 \succ d_3$.
Our program outputs the following list as a **comprehensive Gröbner system** for $\langle F\rangle$ with respect to $\succ$:

```
[b]==0,  [a-1]!=0,
[a*x2*d2+2*x1*d1+1,x2^2*d2,d3]
[b,a+2]==0,  [1]!=0,
[-2*x2*d2+2*x1*d1+1,-x2^2*d2,d3]
[a+2]==0,  [(b-4)*b]!=0,
[2*x1*d1+1,x2*d2,d3]
[b,a-1]==0,  [1]!=0,
[-d1*d3,x2*d2+2*x1*d1+1,-x2^2*d2,x2*d3,x3*d3]
[a-1]==0,  [b]!=0,
[d1*d3,-2*x3*d3+2*b*x1*d1+b,2*x3*d3+b*x2*d2,2*x3*d3^2+(-b+2)*d3,x2*d3]
[a+2,b-4]==0,  [1]!=0,
[(d1-x2*x3)*d3,2*x3*d3-2*x1*d1-1,x3*d3-x2*d2,2*x3*d3^2+(-2*x1*x2*x3+1)*d3]
[a-1,b-4]==0,  [1]!=0,
[-d1*d3,-x3*d3+4*x1*d1+2,x3*d3+2*x2*d2,x3*d3^2-d3,x2*d3]
[b-4]==0,  [(a-1)*(a+2)]!=0,
[2*x1*d1+1,x2*d2,d3]
[0]==0,  [(b-4)*(a-1)*(a+2)*(b)]!=0,
[2*x1*d1+1,x2*d2,d3]
```

Our program outputs the following list as a **comprehensive Gröbner basis** for $\langle F\rangle$ with respect to $\succ$:

```
[(-d1+x2*x3)*d3,a*x2*d2+2*x1*d1+1,(2*x2*d2-2*x1*x2*x3+a+1)*d3,x2*x3*d3-x2
 ^2*d2,2*a*x3*d3+b*x2*d2,2*x3*d3^2+(-2*x1*x2*x3-a-b+3)*d3,(a+2)*x2*d3,(b-
 4)*x2*d3,(a^2+a-2)*d3,((b-4)*a-b+4)*d3,(a+2)*x2*d3,2*x3*d3^2+(-2*x1*x2*x
 3-a-b+3)*d3,2*a*x3*d3+b*x2*d2,(2*a-4)*x3*d3+(-b+4)*a*x2*d2+8*x1*d1+4,(-d
 1+x2*x3)*d3,6*x3*d3^2+((2*a-2)*x1*x2*x3-3*a-3*b+9)*d3,-2*a*x3*d3+(4*a-b)
 *x2*d2+8*x1*d1+4,(-3*d1+(-a+1)*x2*x3)*d3,2*a*x3*d3+(2*a+b)*x2*d2+4*x1*d1
 +2,(-a+1)*x2*x3*d3-3*x2^2*d2,a*x2*d2+2*x1*d1+1,(-b+4)*x2*d3,b*x2*x3*d3-4
 *x2^2*d2,(-4*d1+b*x2*x3)*d3,-6*x3*d3^2+(6*x1*x2*x3+(b-1)*a+2*b-5)*d3,x2*
 x3*d3-x2^2*d2,(2*x2*d2-2*x1*x2*x3+a+1)*d3,(-b*a+b-4)*x3*d3-2*b*x2*d2,(b-
 4)*x2*d3]
```

8.3 Other approaches

In chapter 5 and 6, we introduce the different approaches for computing parametric Gröbner bases. We can also apply these approaches for rings of differential operators.

8.3.1 Nabeshima's approach for computing comprehensive Gröbner systems

In chapter 5, we saw the new algorithm for computing comprehensive Gröbner systems in $K[\bar{A}][\bar{X}]$. One can easily generalize the approach to $K[\bar{A}][\bar{X},\bar{D}]$ and $(K[\bar{A}][\bar{X}])[\bar{D}]$. We have the following key theorem of the algorithm.

Theorem 8.3.1. Let F be a subset of $K[\bar{A}][\bar{X},\bar{D}]$, $H = \{g, g_1, \ldots, g_l\}$ a Gröbner basis

for $\langle F \rangle$ with respect to $\succ$. Select g from H, and set $r := \frac{1}{\mathrm{lc}_{\bar{A}}(g)}$ (r is a new variable) and $g' := \mathrm{lpp}_{\bar{A}}(g) + r \cdot (g - \mathrm{lm}_{\bar{A}}(g))$. Suppose that $H' := (H \backslash \{g\}) \cup \{g'\} = \{g', g_1, \ldots, g_l\} \subset K[r, \bar{A}][\bar{X}, \bar{D}]$, and G' is a Gröbner basis of H' with respect to $\succ$ in $K[r, \bar{A}][\bar{X}, \bar{D}]$. Furthermore, $G := \left\{ f \in K[\bar{A}][\bar{X}, \bar{D}] \,\middle|\, f \neq 0, f = \mathrm{lc}_{\bar{A}}(g)^k \cdot \sigma_{r=1}(q),\ \deg_r(q) = k \in \mathbb{N}, q \in G' \right\}$ and $\{h_{01}, \ldots, h_{0e}\} := \{\mathrm{lc}_{\bar{A}}(f) \in K[\bar{A}] : f \in G\}$.
Then, for any $\bar{a} \in L^m \backslash (\mathbb{V}(\mathrm{lc}_{\bar{A}}(g)) \cup \mathbb{V}(h))$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ with respect to $\succ$ in $L[\bar{X}, \bar{D}]$ where $h = \mathsf{LCM}(h_{01}, \cdots, h_{0e})$. ($\sigma_{r=1}(q)$ means substituting 1 for the variable r of q.)

Proof. This proof is the same as Theorem 5.2.1. Note that $K[\bar{A}][\bar{X}, \bar{D}]$ is a left Noetherian ring. □

By this theorem, we can follow the algorithm NEW in chapter 5 for computing comprehensive Gröbner systems in rings of differential operators.

8.3.2 The approach of von Neumann regular rings

In chapter 6, we saw the relations between comprehensive Gröbner bases and Gröbner bases in a polynomial ring over a commutative von Neumann regular ring. That is, we saw alternative comprehensive Gröbner bases. We can apply this approach for rings of differential operators, too. Then, the alternative comprehensive Gröbner bases in rings of differential operators have the good properties of reduced Gröbner bases in a ring of differential operators over a commutative von Neumann regular ring. For instance, there exists a canonical form of an alternative comprehensive Gröbner basis.

8.4 Applications

In this section, we give one simple example for applying the elimination property of Gröbner bases to systems of linear differential equations.

We have the following linear differential equations.

$$\begin{cases} 2xy'' = 0 \\ y''' + ax^2y' - bxy = 0. \end{cases}$$

This system of linear ordinary differential equations can be written as

$$\begin{cases} (2x\partial^2)y = 0 \\ (\partial^3 + ax^2\partial - bx)y = 0 \end{cases}$$

where ∂ is the partial derivative $\frac{\partial}{\partial x}$ and a, b are parameters. Set $f_1 = 2x\partial^2, f_2 = \partial^3 + ax^2\partial - bx$ in $\mathbb{C}[x][\partial]$. We compute a comprehensive Gröbner system for $\langle f_1, f_2 \rangle$ in $C[x][\partial]$. Then, the program outputs the following as the comprehensive Gröbner system.

```
[a-b]==0,  [b]!=0,
[d^2,-x*d+1]

[a,b]==0,  [1]!=0,
[d^2]

[b]==0,  [a]!=0,
[d]
```

```
[0]==0,   [(b)*(a-b)]!=0,
[1]
```

This output means the following;

$$\begin{cases} \{\partial^2, -x\partial + 1\} & \text{if } a-b=0, b \neq 0, \\ \{\partial^2\} & \text{if } a=b=0, \\ \{\partial\} & \text{if } b=0, a \neq 0, \\ \{1\} & \text{if } b(a-b) \neq 0. \end{cases}$$

Hence, the system of linear differential equations can be reduced to :

$$\begin{cases} \{y'', -xy' + y\} & \text{if } a-b=0, b \neq 0, \\ \{y''\} & \text{if } a=b=0, \\ \{y'\} & \text{if } b=0, a \neq 0, \\ \{y\} & \text{if } b(a-b) \neq 0. \end{cases}$$

Therefore;
(1) Case $\{a-b=0, b \neq 0\}$:
Since $y''=0$, $y=cx$ is the general solution of the system where $c \in \mathbb{C}$.

(2) Case $\{a=b=0\}$:
Since $y''=0$, $y=cx$ is the general solution of the system where $c \in \mathbb{C}$.

(3) Case $\{b=0\}$:
Since $y'=0$, $y=c$ is the general solution of the system where $c \in \mathbb{C}$.

(4) Case $\{b(a-b) \neq 0\}$:
Since $y=0$, we have only trivial solution $y=0$.

Chapter 9

Comprehensive Gröbner bases for modules

This chapter presents algorithms for computing comprehensive Gröbner bases and comprehensive Gröbner systems for modules. Several algorithms [Wei92, Mon02, MM06, SS03, SS06, Wei03] are known for computing comprehensive Gröbner bases in polynomial rings (see chapter 4). In this chapter we consider the problem of computing comprehensive Gröbner bases and comprehensive systems for modules. Theoretically, Gröbner basis algorithms admit natural extensions to modules. However, especially in the parametric situation, complexity is an important issue. An efficient algorithm for computation of comprehensive Gröbner bases over polynomial rings was proposed by Suzuki and Sato (see chapter 4 [SS06]). We describe the generalization of the Suzuki-Sato algorithm to the module case.

By studying comprehensive Gröbner bases for modules, we can solve a lot of parametric problems. For example, consider the problem of syzygy computations. In the ordinary setting, computing a Gröbner basis over a module is closely related to the computation of syzygies. In parametric setting, by computing a comprehensive Gröbner basis (or system) we can obtain parametric syzygies (we will see this application).
Almost all results of this chapter are from the author's paper [Nab07a].

9.1 Notations

Remember that K and L denote fields such that L is an extension of K. $\bar{X} = \{X_1, \ldots, X_n\}$ and $\bar{A} = \{A_1, \ldots, A_m\}$ denote finite sets of variables such that $\bar{X} \cap \bar{A} = \emptyset$. $K[\bar{A}][\bar{X}]$ is the polynomial ring with coefficients in $K[\bar{A}]$.

We have already seen the notation of $K[\bar{A}, \bar{X}]^r$ in chapter 2. We introduce the notations of $K[\bar{A}][\bar{X}]^r$ as follows.

Let f, g be non-zero vectors in $K[\bar{A}][\bar{X}]^r$ and $\succ_m$ be an arbitrary module order on $\mathrm{pp}(\bar{X})^r$. We apply the subscript $\bar{A}$ for notations of $K[\bar{A}][\bar{X}]^r$.

- The **set of module power products** of f that appear with a non-zero coefficient, is written $\mathrm{pp}(f)$.
- The biggest module power product of $\mathrm{supp}_{\bar{A}}(f)$ with respect to $\succ_m$ is denoted by $\mathrm{lpp}_{\bar{A}}(f)$ and is called the **leading power product** of g with respect to $\succ_m$.
- The coefficient corresponding to $\mathrm{lpp}_{\bar{A}}(f)$ is called the **leading coefficient** of f with respect to $\succ_m$ which is defined by $\mathrm{lc}_{\bar{A}}(f)$.

- The product $\mathrm{lc}(f)\,\mathrm{lpp}(f)$ is called the **leading monomial** of f with respect to $\succ_m$ which is defined by $\mathrm{lm}_{\bar{A}}(f)$.
- The **set of monomials** of f is denoted by $\mathrm{Mono}_{\bar{A}}(f)$.
- If $\mathrm{lpp}(f) = A_1^{\alpha_1} \cdots A_m^{\alpha_m} X_1^{\beta_1} \cdots X_n^{\beta_n} e_i \in \mathrm{pp}(\bar{A}, \bar{X})^r$, then $\deg_{\{\bar{X},\bar{A}\}}(\mathrm{lpp}(f)) := (\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_n) \in \mathbb{N}^{m+n}$, and $\deg_{\bar{X}}(\mathrm{lpp}(f)) := (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n$. **(We apply the subscript $\bar{X}$ for the degree.)**

Example 9.1.1. Let a, b, x, y be variables and $f = \begin{pmatrix} 2ax - bx + y^2 \\ axy + 3 \end{pmatrix}$ be a vector.

If we consider f as an element of $\mathbb{Q}[a, b, x, y]^2$ with a module order $\succ_m := (POT, \succ_{\{x,y\},\{a,b\}}) = (POT, (x \succ_{lex} y, a \succ_{lex} b))$, then we have the following

- $\mathrm{pp}(f) = \{axe_1, bxe_1, y^2e_1, axye_2, e_2\}$,
- $\mathrm{lpp}(f) = axe_1$,
- $\mathrm{lc}(f) = 2$,
- $\mathrm{lm}(f) = 2axe_1$,
- $\mathrm{Mono}(f) = \{2axe_1, bxe_1, y^2e_1, axye_2, 3e_2\}$,
- $\deg_{\{a,b,x,y\}}(\mathrm{lpp}(f)) = (1, 0, 1, 0) \in \mathbb{N}^4$, $\deg_{\{a\}}(\mathrm{lpp}(f)) = 1$.

We have a module order $\succ_m := (POT, x \succ_{lex} y)$. If we consider f as a vector of $\mathbb{Q}[a, b][x, y]^2$, then we have the following

- $\mathrm{pp}_{\{a,b\}}(f) = \{xe_1, y^2e_1, xye_2, e_2\}$,
- $\mathrm{lpp}_{\{a,b\}}(f) = xe_1$,
- $\mathrm{lc}_{\{a,b\}}(f) = 2a - b$,
- $\mathrm{lm}_{\{a,b\}}(f) = (2a - b)xe_1$,
- $\mathrm{Mono}_{\{a,b\}}(f) = \{(2a - b)xe_1, y^2e_1, axye_2, 3e_2\}$,
- $\deg_{\{x,y\}}(\mathrm{lpp}_{\bar{A}}(f)) = (1, 0) \in \mathbb{N}^2$, $\deg_{\{y\}}(\mathrm{lpp}_{\bar{A}}(f)) = 0$.

9.2 Comprehensive Gröbner systems for modules

Here we present an algorithm for computing comprehensive Gröbner systems for modules. This algorithm is the generalization of the `Suzuki-Sato` algorithm (see chapter 4) to modules.
As we saw the algorithms for computing comprehensive Gröbner systems in several domains, we need to consider the stability of submodules under specializations.

In order to describe the stability of submodules under specializations, we need to introduce a definition of Gröbner bases in $K[\bar{A}][\bar{X}]^r$, and an algorithm for computing them.

Definition 9.2.1. Let $\succ_m$ be a module order on $\mathrm{pp}(\bar{X})^r$. A finite set $G = \{g_1, \ldots, g_s\}$ of a submodule M in $K[\bar{A}][\bar{X}]^r$ is said to be a **Gröbner basis** with respect to $\succ_m$ if

$$\mathrm{lm}_{\bar{A}}(M) = \langle \mathrm{lm}_{\bar{A}}(g_1), \ldots, \mathrm{lm}_{\bar{A}}(g_s) \rangle$$

where $\mathrm{lm}_{\bar{A}}(M) = \{\mathrm{lm}_{\bar{A}}(f) | f \in M\}$.

In order to compute a Gröbner basis above, we need the following special module order.

Definition 9.2.2 (Hybrid module order 1). Let $\succ_m$ be a module order on $\mathrm{pp}(\bar{X})^r$ and $\succ_1$ a term order on $\mathrm{pp}(\bar{A})$. Let $A_1X_1e_i, A_2X_2e_j \in \mathrm{pp}(\bar{A}, \bar{X})^r$ for $1 \le i, j \le r$

where $A_1, A_2 \in \mathrm{pp}(\bar{A})$ and $X_1, X_2 \in \mathrm{pp}(\bar{X})$. Then a **hybrid module order 1** $\succ_{hm1}$ on $\mathrm{pp}(\bar{A}, \bar{X})^r$ is defined as follows

$$A_1X_1e_i \succ_{hm1} A_2X_2e_j \iff X_1e_i \succ_m X_2e_j \text{ or } (X_1e_i = X_2e_j \text{ and } A_1 \succ_1 A_2).$$

This hybrid module order 1 is written as $\succ_{hm1} := (\succ_m, \succ_1)$.

Remark: If $\succ_m$ is TOP, then we have to consider

$$\bar{X} \succ e_1 \succ e_2 \succ \cdots \succ e_r \succ \bar{A}.$$

This order is different from $\succ_m$, and not TOP and POT.
If $\succ_m$ is POT, then we have to consider

$$e_1 \succ e_2 \succ \cdots \succ e_r \succ \bar{X} \succ \bar{A}.$$

Actually, this order is still POT, so nothing difficulties.

The following lemma tells us how to compute Gröbner bases in $K[\bar{A}][\bar{X}]^r$.

Lemma 9.2.3. Let F be a set of vectors in $K[\bar{A}][\bar{X}]^r$. Then, F can be seen as a set of vectors in $K[\bar{A}, \bar{X}]^r$. Let $G = \{g_1, \dots, g_s\}$ be a Gröbner basis for $\langle F \rangle$ in $K[\bar{A}, \bar{X}]^r$ with respect to a hybrid module order 1 $\succ_{hm1} := (\succ_m, \succ_1)$ where $\succ_m$ is a module order on $\mathrm{pp}(\bar{X})^r$ and $\succ_1$ is a term order on $\mathrm{pp}(\bar{A})$.
Then, G can be also seen as a set of vectors in $K[\bar{A}][\bar{X}]^r$. This set G is a Gröbner basis for $\langle F \rangle$ in $K[\bar{A}][\bar{X}]^r$ with respect to a module order $\succ_m$.

Proof. For all $h \in \langle F \rangle \subseteq K[\bar{A}][\bar{X}]^r$, we prove that $\mathrm{lm}_{\bar{A}}(h)$ is generated by $\{\mathrm{lm}_{\bar{A}}(g) | g \in G\}$. Since h can be seen as an element of $K[\bar{A}, \bar{X}]^r$ and G is a Gröbner basis for $\langle F \rangle$ with respect to $\succ_{hm1}$ in $K[\bar{A}, \bar{X}]^r$, h can be written as

$$h = h_1g_1 + \cdots + h_sg_s$$

such that $\mathrm{lm}(h) \succ_{hm1} \mathrm{lm}(h_1g_1) \succ_{hm1} \cdots \succ_{hm1} \mathrm{lm}(h_sg_s)$ in $K[\bar{A}, \bar{X}]^r$ where $h_1, \dots, h_s \in K[\bar{A}, \bar{X}]$. By the hybrid module order 1 $\succ_{hm1}$ on $\mathrm{pp}(\bar{A}, \bar{X})$, we also have

$$\mathrm{lm}_{\bar{A}}(h) \succ_m \mathrm{lm}_{\bar{A}}(h_1g_1) \succ_m \cdots \succ_m \mathrm{lm}_{\bar{A}}(h_sg_s)$$

in $K[\bar{A}][\bar{X}]^r$. As $h = h_1g_1 + \cdots + h_sg_s$ in $K[\bar{A}, \bar{X}]^r$, $\mathrm{lm}_{\bar{A}}(h) = \mathrm{lm}_{\bar{A}}(h_1g_1 + \cdots + h_sg_s)$ in $K[\bar{A}][\bar{X}]^r$. W.l.o.g., $h_1g_1, \dots, h_kg_k$ have the same leading power product $\mathrm{lpp}_{\bar{A}}(h)$ where $k \leq s$. That is,

$$\mathrm{lm}_{\bar{A}}(h) = \mathrm{lm}_{\bar{A}}(h_1g_1) + \cdots + \mathrm{lm}_{\bar{A}}(h_kg_k).$$

Obviously $\mathrm{lm}_{\bar{A}}(h_ig_i) = \mathrm{lm}_{\bar{A}}(h_i)\,\mathrm{lm}_{\bar{A}}(g_i)$, hence

$$\mathrm{lm}_{\bar{A}}(h) = \mathrm{lm}_{\bar{A}}(h_1)\,\mathrm{lm}_{\bar{A}}(g_1) + \cdots + \mathrm{lm}_{\bar{A}}(h_k)\,\mathrm{lm}_{\bar{A}}(g_k).$$

Therefore, $\mathrm{lm}_{\bar{A}}(h) \in \langle \{\mathrm{lm}_{\bar{A}}(g) \,|\, g \in G\} \rangle$. G is a Gröbner basis for $\langle F \rangle$ with respect to $\succ_m$ in $K[\bar{A}][\bar{X}]^r$. □

Algorithm 9.2.4. GröbnerBasisM$(F, \succ_m)$

Input F: a finite set of vectors in $K[\bar{A}][\bar{X}]^r$,
$\succ_m$: a module order on $\mathrm{pp}(\bar{X})^r$,
Output G: a Gröbner basis of $\langle F \rangle$ with respect to $\succ_m$ in $K[\bar{A}][\bar{X}]^r$.

1. Consider F as a set of vectors in $K[\bar{A},\bar{X}]^r$.
2. Compute the **reduced Gröbner basis** G for $\langle F\rangle$ with respect to $\succ_{hm1}=(\succ_m,\succ_1)$ where $\succ_1$ is a term order on $\mathrm{pp}(\bar{A})$.
3. Consider G as a set of vectors in $K[\bar{A}][\bar{X}]^r$. Then, by Lemma 9.2.3, G is a Gröbner basis for $\langle F\rangle$ with respect to $\succ_m$ in $K[\bar{A}][\bar{X}]^r$.

Remark: In the second step of the algorithm GröbnerBasisM, we compute the reduced Gröbner basis in $K[\bar{A},\bar{X}]^r$. Actually, we don't need to compute the reduced Gröbner basis. It is sufficient to compute a (normal) Gröbner basis. However, we need the properties of reduced Gröbner bases for proving Theorem 9.3.8 (the algorithm FCGB). Therefore we applied them for computing Gröbner bases in $K[\bar{A}][\bar{X}]^r$.

Every ring homomorphism $\pi : K[\bar{A}] \to L$ extends naturally to a homomorphism $\pi : K[\bar{A}][\bar{X}] \to L[\bar{X}]$. Moreover, we extend the homomorphism $\pi : K[\bar{A}][\bar{X}]^r \to L[\bar{X}]^r$ for modules. The image under π of a submodule $I \subseteq K[\bar{A}][\bar{X}]^r$ generates the extension submodule $\pi(I) := \{\pi(f)|f \in I\} \subseteq L[\bar{X}]^r$.

Definition 9.2.5. We call a submodule $I \subseteq K[\bar{A}][\bar{X}]^r$ **stable** under the ring homomorphism π and a order $\succ_m$ if for each $i = 1,\dots,r$, it satisfies

$$\pi(\mathrm{lm}_{\bar{A}}(I)) = \mathrm{lm}(\pi(I))$$

where $\pi(\mathrm{lm}_{\bar{A}}(I)) := \{\pi(\mathrm{lm}_{\bar{A}}(f))|f \in I\}$ and $\mathrm{lm}(\pi(I)) := \{\mathrm{lm}(f)|f \in \pi(I)\}$.

As we saw the stability of Gröbner bases in several domains, we can easily generalize the theory of the stability of ideals under specialization to submodules. Then, in $K[\bar{A}][\bar{X}]^r$, the generalization of **"Kalkbrener [Kal97] Theorem 3.1"** also holds. This theorem is the key theorem of this chapter which is the following.

Theorem 9.2.6. Let π be a ring homomorphism from $K[\bar{A}]$ to L, I a submodule of $K[\bar{A}][\bar{X}]^r$ and $G = \{g_1,\dots,g_s\}$ a Gröbner basis for I with respect to a module order $\succ_m$ where $r \in \mathbb{N}$.
We assume that the g_i's are ordered in such a way that there exists an $q \in \{1,\dots,s\}$ with $\pi(\mathrm{lc}(g_i)) \neq 0$ for $i \in \{1,\dots,q\}$ and $\pi(\mathrm{lc}(g_i)) = 0$ for $i \in \{q+1,\dots,s\}$. Then the following three conditions are equivalent.

1. I is stable under π and $\succ_m$.
2. $\{\pi(g_1),\dots,\pi(g_q)\}$ is a Gröbner basis for $\pi(I)$ in $L[\bar{X}]^r$ with respect to a module order $\succ_m$.
3. For every $i \in \{q+1,\dots,s\}$ the polynomial $\pi(g_i)$ is reducible to 0 modulo $\{\pi(g_1),\dots,\pi(g_q)\}$ in $L[\bar{X}]^r$.

Proof. This proof is almost same as Theorem 4.3.2. □

For arbitrary $\bar{a} \in L^m$, we define the canonical specialization homomorphism $\sigma_{\bar{a}} : K[\bar{A}] \to L$ induced by $\bar{a}$, and we can naturally extend it to $\sigma_{\bar{a}} : (K[\bar{A}])[\bar{X}] \to L[\bar{X}]$. Moreover, we extend the homomorphism $\sigma_{\bar{a}} : K[\bar{A}][\bar{X}]^r \to L[\bar{X}]^r$ for modules.

The next two corollaries are the direct consequences of Theorem 9.2.6.

Corollary 9.2.7. Let $\succ_m$ be a module order on $\mathrm{pp}(\bar{X})^r$, $F \subset K[\bar{A}][\bar{X}]^r$, G a Gröbner basis for a submodule $\langle F\rangle$ of $K[\bar{A}][\bar{X}]^r$ with respect to $\succ_m$. Suppose that $\{h_1,\dots,h_l\} = \{\mathrm{lc}_{\bar{A}}(g) \in K[\bar{A}]|\,\mathrm{lc}_{\bar{A}}(g) \notin K, g \in G\}$ and $h = \mathsf{LCM}(h_1,\dots,h_k)$.

Then for all $\bar{a} \in L^m \setminus \mathbb{V}(h)$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}]^r$ with respect to $\succ_m$.

Proof. For all $\bar{a} \in L^m \setminus \mathbb{V}(h)$, we have $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) \neq 0$ for each $g \in G$. By Theorem 9.2.6, $2 \iff 3$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ in $K[\bar{X}]^r$ with respect to $\succ_m$. □

Corollary 9.2.8. Let $\succ_m$ be a module order on $\mathrm{pp}(\bar{X})^r$, $F = \{f_1, \ldots, f_l\}$ be a set of vectors in $K[\bar{A}][\bar{X}]^r$, S a set of polynomials in $K[\bar{A}]$ and $T = \{se_i | s \in S, 1 \leq i \leq r\}$. Furthermore, G be a Gröbner basis of a submodule $\langle F \cup T \rangle \subseteq K[\bar{A}][\bar{X}]^r$ with respect to $\succ_m$. Suppose that $B =: \{q | q \in G \cap K[\bar{A}]e_i, \mathrm{lc}_{\bar{A}}(g) \in \langle S \rangle, 1 \leq i \leq r\}$, $\{h_1, \ldots, h_l\} = \{\mathrm{lc}_{\bar{A}}(g) \,|\, \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G \setminus B\} \subseteq K[\bar{A}]$ and $h = \mathsf{LCM}(h_1, \ldots, h_k)$.
Then, for $\bar{a} \in \mathbb{V}(S) \setminus \mathbb{V}(h)$, $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}]^r$ with respect to $\succ_m$. Actually, we have $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G \setminus B)$.

Proof. Note that for all $q \in B \cap G$, $\sigma_{\bar{a}}(q) = 0$ for all $\bar{a} \in \mathbb{V}(S) \setminus \mathbb{V}(h)$. Thus, $\langle G \setminus B \rangle$ is stable under $\sigma_{\bar{a}}$. Therefore, $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G \setminus B)$. Clearly, for all $\bar{a} \in \mathbb{V}(S) \setminus \mathbb{V}(h)$, we have $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) \neq 0$ for each $g \in G \setminus B$. By Theorem 9.2.6, $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}]^r$ with respect to $\succ_m$. □

By using the two corollaries above, we can construct an algorithm for computing comprehensive Gröbner systems in $K[\bar{A}][\bar{X}]^r$. Before describing the algorithm, we have to define comprehensive Gröbner systems in $K[\bar{A}][\bar{X}]^r$.

Definition 9.2.9. Let F be a set of vectors in $K[\bar{A}][\bar{X}]^r$, $\mathcal{A}_1, \ldots, \mathcal{A}_l$ algebraically constructible subsets of L^m and $G_1, \ldots, G_l$ subsets of $K[\bar{A}][\bar{X}]^r$. Let $\mathcal{S}$ be a subset of L^m such that $\mathcal{S} \subseteq \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_l$.
A finite set $\mathcal{G} = \{(\mathcal{A}_1, G_1), \ldots, (\mathcal{A}_l, G_l)\}$ of pairs is called a **comprehensive Gröbner system** on $\mathcal{S}$ for $\langle F \rangle$ if $\sigma_{\bar{a}}(G_i)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ in $L[\bar{X}]^r$ for each $i = 1, \ldots, l$ and $\bar{a} \in \mathcal{A}_i$. Each $(\mathcal{A}_i, G_i)$ is called a segment of $\mathcal{G}$. We simply say $\mathcal{G}$ is a comprehensive Gröbner system for F if $\mathcal{S} = L^m$.

Let F be a subset of $K[\bar{A}][\bar{X}]^r$. Then, by the Corollary 9.2.7, we obtain one of segments of a comprehensive Gröbner system for $\langle F \rangle$ as $(\emptyset, h, G)$ on $L^m \setminus \mathbb{V}(h)$ where the notations are from Corollary 9.2.7. We have to consider a comprehensive Gröbner system on the whole space L^m. Therefore, by Corollary 9.2.8, we compute other segments recursively.

In the following algorithm, we assume the algorithm `factorize` and LCM. The algorithm $\mathsf{factorize}(h)$ outputs a set of all irreducible factors of h in $K[\bar{A}]$ where $h \in K[\bar{A}]$, and the algorithm $\mathsf{LCM}(h_1, \ldots, h_l)$ outputs the least common multiple of $h_1, \ldots, h_l$ in $K[\bar{A}]$ where $h_1, \ldots, h_l \in K[\bar{A}]$.

Algorithm 9.2.10. $\mathsf{CGSM}(F, \succ_m)$ (Comprehensive Gröbner Systems for Modules)

Input F: a finite set of vectors in $K[\bar{A}][\bar{X}]^r$,
  $\succ_m$: a module order $\mathrm{pp}(\bar{X})^r$,
Output H: comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ_m$ in $K[\bar{A}][\bar{X}]^r$.
begin
  $G \leftarrow \mathsf{Gr\ddot{o}bnerBasisM}(F, \succ_m)$
if $e_1, \ldots, e_r \in G$ **then**
  $\mathsf{return}(\{(\emptyset, \{1\}, G)\})$
end-if
  $S \leftarrow \{h_1, \ldots, h_l\} := \{q | q \in \mathsf{factorize}(\mathrm{lc}_{\bar{A}}(g)), \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G\}$
if $S \neq \emptyset$ **then**

$h \leftarrow \mathsf{LCM}(h_1, \ldots, h_l)$
$H \leftarrow \{(\emptyset, h, G)\} \cup$ $\mathsf{CGSMainM}(G \cup \{h_1e_1, \ldots, h_1e_r\}, \{h_1\}, \succ_m) \cup$
$\cdots \cup \mathsf{CGSMainM}(G \cup \{h_le_1, \ldots, h_le_r\}, \{h_l\}, \succ_m)$
else
$H \leftarrow \{(\emptyset, \{1\}, G)\}$
end-if
`return`(H)
end

Algorithm 9.2.11. $\mathsf{CGSMainM}(F, Z, \succ_m)$ (CGS Main for Modules)

Input F: a finite set of vectors in $K[\bar{A}][\bar{X}]^r$,
Z : a finite set of polynomials in $K[\bar{A}]$,
$\succ_m$: a module order $\mathrm{pp}(\bar{X})^r$,
Output H: comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ_m$ on $\mathbb{V}(Z)$ in $K[\bar{A}][\bar{X}]^r$.

begin
$G \leftarrow \mathsf{Gr\ddot{o}bnerBasisM}(F, \succ_m)$
if $e_1, \ldots, e_r \in G$ **then**
$C \leftarrow$ the reduced Gröbner basis for $\langle Z \rangle$ in $K[\bar{A}]$
if $1 \in C$ **then**
$H \leftarrow \emptyset$
else
$H \leftarrow \{(C, \{1\}, \{e_1, \ldots, e_r\})\}$
end-if
else
$B \leftarrow \{g | g \in G \cap K[\bar{A}]e_i, \mathrm{lc}_{\bar{A}}(g) \in \langle Z \rangle,$ for some $i \in \{1, \ldots, r\}\}$
$S \leftarrow \{h_1, \ldots, h_l\} := \{q | q \in \mathsf{factorize}(\mathrm{lc}_{\bar{A}}(g)), \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G \backslash B\}$
if $S \neq \emptyset$ **then**
$h \leftarrow \mathsf{LCM}(h_1, \ldots, h_l)$
$H \leftarrow \{(Z, h, G \backslash B)\} \cup \mathsf{CGSMainM}(G \cup \{h_1e_1, \ldots, h_1e_r\}, Z \cup \{h_1\}, \succ_m) \cup$
$\cdots \cup \mathsf{CGSMainM}(G \cup \{h_le_1, \ldots, h_le_r\}, Z \cup \{h_l\}, \succ_m)$
else
$H \leftarrow \{(Z, \{1\}, G \backslash B)\}$
end-if
end-if
return(H)
end

Remark: As we saw in Algorithm 4.4.3, we can apply a lot of optimization techniques for getting nice and small comprehensive Gröbner systems. See the remark of Algorithm 4.4.3.

Theorem 9.2.12. The algorithm `CGSM` (Comprehensive Gröbner Systems for Modules) terminates for any input of a finite subset F of $K[\bar{A}][\bar{X}]^r$. If H is the output of $\mathsf{CGSM}(F, \succ_m)$, then H is a comprehensive Gröbner system for $\langle F \rangle$ on L^m.

Proof. First we show the termination. Obviously, the algorithms GröbnerBasisM, LCM and `factorize` terminate. We have to prove the termination of CGSMainM.
We suppose that $\mathsf{CGSMainM}(F, Z)$ does not terminate, then there exists an infinite sequence $F_0, F_1, \ldots$, such that $F_0 = F$ and $F_i \neq F_{i+1}$ for $i \in \mathbb{N}$. By the algorithm,

$F_{n+1} = F_s \cup \{h_s\}$ for some $h_s \in K[\bar{A}]$ such that $h_s \notin \langle F_s \rangle$. Hence we have $\langle F_s \rangle \subsetneq \langle F_{s+1} \rangle$ for each s. We know that every infinite ascending chain $M_1 \subseteq M_2 \subseteq \cdots$ of submodules of $K[\bar{A}][\bar{X}]^r$ stabilizes. That is, there exists N such that $M_N = M_{N+1} = \cdots = M_{N+l} = \cdots$ for all $0 \leq l$. Therefore, $\langle F_s \rangle \subsetneq \langle F_{s+1} \rangle$ for each s contradicts by the fact. CGSMainM terminates.

We next show that, if $(Z, h, G) \in H$, then the triple (Z, h, G) forms a segment of a comprehensive Gröbner system for $\langle F \rangle$, i.e., $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ for each $\bar{a} \in \mathbb{V}(Z) \setminus \mathbb{V}(h)$.

Let G be a Gröbner basis of the ideal $\langle F' \rangle$ with respect to $\succ_m$ in $K[\bar{A}][\bar{X}]^r$, $B := \{g | g \in G \cap K[\bar{A}]e_i, \mathrm{lc}_{\bar{A}}(g) \in \langle Z \rangle$, for some $i \in \{1, \ldots, r\}\}$ and $\{h_1, \ldots, h_l\} := \{q | q \in \texttt{factorize}(\mathrm{lc}_{\bar{A}}(g)), \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G \backslash B\}$ and $h = \mathsf{LCM}(h_1, \ldots, h_l)$. Then by Corollary 9.2.8, $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F') \rangle$ for each $\bar{a} \in \mathbb{V}(Z) \setminus \mathbb{V}(h)$. In fact $\bar{a} \in \mathbb{V}(Z) \setminus \mathbb{V}(h)$ implies $\sigma_{\bar{a}}(G \backslash B) = \sigma_{\bar{a}}(G)$ and $\sigma_{\bar{a}}(F') = \sigma_{\bar{a}}(F)$. This means that $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$.

We have to finally prove that the conditions in H covers the entire L^m, i.e.,

$$L^m = \bigcup_{(P,h,G) \in H} \mathbb{V}(P) \setminus \mathbb{V}(h).$$

In the algorithm, if the first "`if`" of `CGSM` is true, then the output is $\{(\emptyset, \{1\}, G)\}$. The condition is $\mathbb{V}(\emptyset) \setminus \mathbb{V}(1) = L^m$.

If the second "`if`" of `CGSM` is false, then the output is $\{(\emptyset, \{1\}, G)\}$. The condition is L^m.

If the second "`if`" of `CGSM` is true, then we have to consider $\{(\emptyset, h, G)\} \cup \mathsf{CGSMainM}(G \cup \{h_1 e_1, .., h_1 e_r\}, \{h_1\}) \cup \cdots \cup \mathsf{CGSMainM}(G \cup \{h_l e_1, .., h_l e_r\}, \{h_l\})$.

Let us consider a subalgorithm CGSMainM. We assume that one of inputs of CGSMainM is (F, Z) where $F \subseteq K[\bar{A}][\bar{X}]^r$ and $Z \subseteq K[\bar{A}]$. Let G' be a Gröbner basis of $\langle F \rangle$ with respect to $\succ_m$ in $K[\bar{A}][\bar{X}]^r$ and let $h' = h'_1 \cdots h'_l$ in $K[\bar{A}]$. Then, the following equation always holds.

$$\mathbb{V}(Z) = (\mathbb{V}(Z) \setminus \mathbb{V}(h')) \cup \bigcup_{i=1}^{l} \mathbb{V}(Z \cup h'_i).$$

The equation above follows by the induction on the well-founded tree of the algorithm. Therefore, the condition of $\{(\emptyset, h', G)\} \cup$ $\mathsf{CGSMain}(G \cup \{h'_1 e_1, \ldots, h'_1 e_r\}, \{h'_1\}) \cup \cdots \cup \mathsf{CGSMain}(G \cup \{h'_l e_1, \ldots, h'_l e_r\}, \{h'_l\})$ is L^m. $\square$

This algorithm `CGSM` has been implemented in the computer algebra systems singular and Risa/Asir by the author. In the following examples, we see some examples of comprehensive Gröbner systems, and how the algorithm works.

Example 9.2.13. Let $F = \left\{ \begin{pmatrix} ax+1 \\ 0 \end{pmatrix}, \begin{pmatrix} bx+ay \\ x+1 \end{pmatrix} \right\} \subseteq \mathbb{Q}[a,b][x,y]^2$ and let x, y be variables and a, b parameters. We have a module order $\succ_m = (POT, \succ_{lex})$ such that $x \succ_{lex} y$. We ready to compute a comprehensive Gröbner system for $\langle F \rangle$ with respect to $\succ_m$.

1. First, we have to compute a Gröbner basis for $\langle F \rangle$ in $\mathbb{Q}[a,b][x,y]^2$ with respect to $\succ_m$. By the algorithm GröbnerBasisM, we compute the reduced Gröbner basis for $\langle F \rangle$ in $Q[a,b,x,y]^2$ with respect to $\succ_{hm1} = (\succ_m, \succ_{lex})$ such that $a \succ b$. Then the reduced Gröbner basis is the following

$$F_1 = \left\{ \begin{pmatrix} 0 \\ ax^2+ax+x+1 \end{pmatrix}, \begin{pmatrix} a^2y-b \\ ax+a \end{pmatrix}, \begin{pmatrix} bx+ay \\ x+1 \end{pmatrix}, \begin{pmatrix} ax+1 \\ 1 \end{pmatrix} \right\}.$$

The set of every leading coefficient of F_1 is $\mathrm{lc}_{\bar{A}}(F_1) = \{a, a^2, b\}$. By Corollary 9.2.7, when all elements of $\mathrm{lc}_{\{a,b\}}(F_1)$ is not zero under specialization $\sigma_{\bar{\alpha}}$ where $\bar{\alpha} \in \mathbb{Q}^2$ (i.e. $ab \neq 0$), then $\sigma_{\bar{\alpha}}(F_1)$ is a Gröbner basis for $\langle \sigma_{\bar{\alpha}}(F) \rangle$. Therefore, $(\{0\}, \{ab\}, F_1\})$ is one of segments of a comprehensive Gröbner system for $\langle F \rangle$.

2. By the first step, we have one segment of a comprehensive Gröbner system for $\langle F \rangle$. Actually, we need to consider the whole parametric spaces parameters (i.e. we need to consider the case $\{ab = 0\}$). We have `factorize`$(ab)= \{a, b\}$. We need to consider two cases $\{a = 0\}$ and $\{b = 0\}$. First we consider the case $a = 0$. By Corollary 9.2.8, we need to compute a reduced Gröbner basis of $H1 = \left\langle F \cup \left\{ \begin{pmatrix} a \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ a \end{pmatrix} \right\} \right\rangle$ in $\mathbb{Q}[a, b, x, y]^2$ with respect to $\succ_{hm1}$. Then the algorithm GröbnerBasisM outputs the following
$$F_2 = \left\{ \begin{pmatrix} 0 \\ x+1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ a \end{pmatrix} \right\}.$$
Since we are considering the case $\{a = 0\}$, we can eliminate the vector $\begin{pmatrix} 0 \\ a \end{pmatrix}$ from F_2. Therefore, we can have
$$F_2' = \left\{ \begin{pmatrix} 0 \\ x+1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\}.$$
A triple $(\{a\}, \{1\}, F_2')$ is one of segments of a comprehensive Gröbner system for $\langle F \rangle$.

3. Similarly, we have to consider the case $\{b = 0\}$, i.e., $H2 = \left\langle F \cup \left\{ \begin{pmatrix} b \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ b \end{pmatrix} \right\} \right\rangle$. Then a Gröbner basis of $H2$ is
$$F_3 = \left\{ \begin{pmatrix} 0 \\ ax^2 + ax + x + 1 \end{pmatrix}, \begin{pmatrix} ax+1 \\ 0 \end{pmatrix}, \begin{pmatrix} b \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ b \end{pmatrix}, \begin{pmatrix} y \\ -x^2 - x \end{pmatrix} \right\}.$$
Since we do not need $\begin{pmatrix} b \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ b \end{pmatrix}$ in F_3, we eliminate them from F_3. Hence, we have
$$F_3' = \left\{ \begin{pmatrix} 0 \\ ax^2 + ax + x + 1 \end{pmatrix}, \begin{pmatrix} ax+1 \\ 0 \end{pmatrix}, \begin{pmatrix} y \\ x^2 - x \end{pmatrix} \right\}.$$
Since $\mathrm{lc}_{\{a,b\}}(F_3') = \{a\}$, one of segments of the comprehensive Gröbner system is $(\{b\}, \{a\}, F_3')$.

4. Finally, we need to consider the case $\{a = 0, b = 0\}$. That is, we compute a Gröbner basis for $H3 = \left\langle F \cup \left\{ \begin{pmatrix} a \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ a \end{pmatrix}, \begin{pmatrix} b \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ b \end{pmatrix} \right\} \right\rangle$. The algorithm GröbnerBasisM outputs the following Gröbner basis for $H3$
$$F_4 = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ a \end{pmatrix}, \begin{pmatrix} 0 \\ b \end{pmatrix}, \begin{pmatrix} 0 \\ x+1 \end{pmatrix} \right\}.$$
We omit $\begin{pmatrix} a \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ b \end{pmatrix}$ from F_4 because of $a = b = 0$. Hence, we obtain
$$F_4' = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ x+1 \end{pmatrix} \right\},$$

and $(\{a, b\}, \{1\}, F'_4)$ is one of the segments.

5. **Solution :** By the steps 1,2,3 and 4, a comprehensive Gröbner system for $\langle F \rangle$ is

$$\{(\{0\}, \{ab\}, F_1), (\{a\}, \{1\}, F'_2), (\{b\}, \{a\}, F'_3), (\{a, b\}, \{1\}, F'_4)\}.$$

That is,

$$\begin{cases} F_1 & \text{if } ab \neq 0, \\ F'_2 & \text{if } a = 0, \\ F'_3 & \text{if } b = 0, a \neq 0, \\ F'_4 & \text{if } a = b = 0. \end{cases}$$

See Figure 9.1

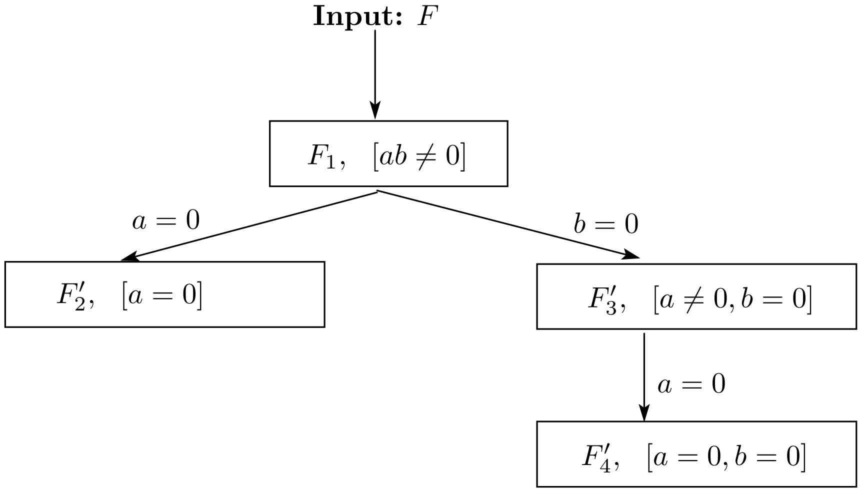

Figure 9.1

Remark: In fact, the segment $(\{a\}, \{1\}, F'_2)$ includes the segment $(\{a, b\}, \{1\}, F'_4)$. Therefore, we can eliminate the segment $(\{a, b\}, \{1\}, F'_4)$ from the list. In the remarks of Algorithm 4.4.3, we mentioned optimization techniques for getting a small and nice comprehensive Gröbner system. One of the techniques is this technique "checking each parameter space". In this example, we just followed the algorithm. See the remark of Algorithm 4.4.3.

Example 9.2.14. Let x, y be variables and a, b parameters.
We have $f_1 = \begin{pmatrix} ax - bx + 1 \\ ax^2y + ax + b \end{pmatrix}$ and $f_2 = \begin{pmatrix} by + a \\ bx^2 + bx + 2 \end{pmatrix}$ in $\mathbb{Q}[a, b][x, y]^2$. Then, our program of singular outputs the following list which is a comprehensive Gröbner system for $\langle f_1, f_2 \rangle$ with respect to $\succ_m = (POT, \succ_{lex})$ such that $x \succ_{lex} y$.

```
[b]==0, [a]!=0
_[1]=[0,x2ya2+xa2-2xa-2]
_[2]=[a,2]

[a,b]==0, [1]!=0
_[1]=[0,1]
_[2]=[1]

[a-b]==0, [b]!=0
```

```
_[1]=[0,x2y2b2+x2yb2-x2b+xyb2+xb2-xb+yb2+b2-2]
_[2]=[1,x2yb+xb+b]

[0]==0, [(a-b)*b]!=0
_[1]=[0,x3ab-x3b2-x2y2ab-x2ya2+x2ab-x2b2+x2b-xyab-xa2+2xa-xb-yb2-ab+2]
_[2]=[yb+a,x2b+xb+2]
_[3]=[xa-xb+1,x2ya+xa+b]
```

The output means the following:

If $b = 0, a \neq 0$, then
$\left\{ \begin{pmatrix} a \\ 2 \end{pmatrix}, \begin{pmatrix} 0 \\ a^2x^2y + a^2x - 2ax - 2 \end{pmatrix} \right\}$.
If $a = b = 0$, then
$\left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$.
If $a - b = 0, b \neq 0$, then
$\left\{ \begin{pmatrix} 1 \\ bx^2y + bx + b \end{pmatrix}, \right.$
$\left. \begin{pmatrix} 0 \\ b^2x^2y^2 + b^2x^2 - bx^2 + b^2xy + b^2x - bx + b^2y + b^2 - 2 \end{pmatrix} \right\}$.
If $(a - b)b \neq 0$, then
$\left\{ \begin{pmatrix} 0 \\ (*1) \end{pmatrix}, \begin{pmatrix} by + a \\ bx^2 + bx + 2 \end{pmatrix}, \begin{pmatrix} ax - bx + 1 \\ ax^2y + ax + b \end{pmatrix} \right\}$,
where $(*1) = abx^3 - b^2x^3 - abx^2y^2 - a^2x^2y + abx^2 - b^2x^2 + bx^2 - abxy - a^2x + 2ax - bx - b^2y - ab + 2$.

9.3 Comprehensive Gröbner bases for modules

Here we present an algorithm that provides an explicit construction of a comprehensive Gröbner basis for $\langle F \rangle$ from any finite set $F \subset K[\bar{A}][\bar{X}]^r$ via the intermediate concept of a Gröbner system.

Definition 9.3.1 (Comprehensive Gröbner Bases). Let F and G be sets of vectors in $K[\bar{A}][\bar{X}]^r$. $G \subset \langle F \rangle$ is called a **comprehensive Gröbner basis** for $\langle F \rangle$ if $\sigma_{\bar{a}}(G)$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F) \rangle$ for each $\bar{a} \in L^m$.

We already saw comprehensive Gröbner systems in previous section which has conditions of parameters. However, comprehensive Gröbner bases do not have conditions of parameters (parametric spaces). A comprehensive Gröbner basis is a set of vectors. In this point, comprehensive Gröbner bases are different from comprehensive Gröbner systems.

In this section we consider an algorithm for computing comprehensive Gröbner bases. Then, we need the following concept.

Definition 9.3.2. Let F be a set of vectors in $K[\bar{A}][\bar{X}]^r$, $s_1, \ldots, s_l, t_1, \ldots, t_l \subset K[\bar{A}]$ and $G_1, \ldots, G_l \subset K[\bar{A}][\bar{X}]^r$. Then a comprehensive Gröbner system $\{(s_1, t_1, G_1), \ldots, (s_l, t_l, G_l)\}$ for F is call **faithful** if $G_i \subset \langle F \rangle$ for each $i = 1, \ldots, l$.

Actually, in this section we describe an algorithm for computing **faithful** comprehensive

Gröbner systems. If $\{(\mathbb{V}(s_1)\setminus\mathbb{V}(t_1), G_1), \ldots, (\mathbb{V}(s_l)\setminus\mathbb{V}(t_l), G_l)\}$ is a faithful comprehensive Gröbner system for $\langle F\rangle$, then by the definition of comprehensive Gröbner basis, $G_1 \cup \cdots \cup G_l$ is a comprehensive Gröbner basis for $\langle F\rangle$. Therefore, we modify the algorithm CGSMainM to compute a faithful comprehensive Gröbner system. The key idea which is from [SS06], is to apply a new variable U. In chapter 4, we introduced a new auxiliary variable U besides $\bar{X}$ and $\bar{A}$ in order to compute comprehensive Gröbner bases. We follow this technique to compute comprehensive Gröbner bases for modules.

We define homomorphisms σ^0 and σ^1 from $K[\bar{A}][U,\bar{X}]^r$ to $K[\bar{A}][\bar{X}]^r$ as a specialization of U with 0 and 1 respectively, i.e. $\sigma^0(f(U,\bar{A},\bar{X})) = f(0,\bar{A},\bar{X})$ and $\sigma^1(f(U,\bar{A},\bar{X})) = f(1,\bar{A},\bar{X})$. Before we introduce the algorithm for computing comprehensive Gröbner bases, we need the following lemma which is also from [SS06].

Lemma 9.3.3. Let F and S be subsets of $K[\bar{A}][\bar{X}]^r$. For any $g \in \langle (U\cdot F)\cup(U-1)\cdot S)\rangle \subseteq K[\bar{A}][U,\bar{X}]^r$, then $\sigma^0(g) \in \langle S\rangle \subseteq K[\bar{A}][\bar{X}]^r$ and $\sigma^1(g) \in \langle F\rangle \subseteq K[\bar{A}][\bar{X}]^r$.

Proof. See [SS06] Lemma 3.1. □

In order to construct an algorithm for computing comprehensive Gröbner bases in $K[\bar{A}][\bar{X}]^r$, we need the following special mix order. This order is very important for our main result of this section. Actually, this special order always tells us what the leading monomial of a vector is under specializations. Moreover, even if we apply the special order for computing Gröbner bases, we can still hold a simple algorithm for computing comprehensive Gröbner bases like algorithm CGSM.

Definition 9.3.4 (Hybrid module order 2). Let $\succ_m$ be a module order on $K[\bar{X}]^r$. Then, a **hybrid module order 2** $\succ_{hm2}$ on $\mathrm{pp}(U,\bar{X})^r$ is defined as follows

$$U^{\alpha_1}x^{\alpha_2}e_i \succ_{hm2} U^{\beta_1}x^{\beta_1}e_j \iff \alpha_1 > \beta_2 \text{ or } (\alpha_1 = \beta_2 \text{ and } x^{\alpha_2}e_i \succ_m x^{\beta_2}e_j),$$

for all $\alpha_1, \beta_1 \in \mathbb{N}$, $\alpha_2, \beta_2 \in \mathbb{N}^n$, $i, j = 1, \ldots, r$. This **hybrid module order 2** with respect to a variable U is written as $\succ_{hm2} := (U, \succ_m)$.

Remark: If $\succ_m$ is TOP, then we have to consider

$$U \succ \bar{X} \succ e_1 \succ e_2 \succ \cdots \succ e_r.$$

Actually, this order is still TOP, so nothing difficulties.
If $\succ_m$ is POT, then we have to consider

$$U \succ e_1 \succ e_2 \succ \cdots \succ e_r \succ \bar{X}.$$

This order is not $\succ_m$ and POT.
In fact, when we compute a comprehensive Gröbner basis, we need two special module orders **"hybrid module order 1"** and **"hybrid module order 2"**. Since these mix orders are very complicated, we have to be careful when we compute a comprehensive Gröbner basis.

The next theorem is the main result of this section. By the following theorem, we can construct an algorithm for computing comprehensive Gröbner bases.

Theorem 9.3.5. Let F be a subset of $K[\bar{A}][\bar{X}]^r$, S' a subset of $K[\bar{A}]$, $S := \{se_i \mid s \in S', 1 \le i \le r\}$ and $\succ_m$ a module order on $\mathrm{pp}(\bar{X})^r$. Let G be a Gröbner basis of $\langle (U \cdot F) \cup (U-1)\cdot S)\rangle$ in $K[\bar{A}][U,\bar{X}]^r$ with respect to a hybrid module order 2 $\succ_{hm2} = (U, \succ_m)$. (We can compute this Gröbner basis by the algorithm GröbnerBasisM

and a hybrid module order 1 "$\succ_{hm1}= (\succ_m, \succ_1)$" where $\succ_1$ is a term order on $\mathrm{pp}(\bar{A})$.)
Suppose that $B_1 := \{g|g \in G \cap K[\bar{A}][U]e_i, \mathrm{lc}_{\bar{A}}(g) \in \langle S'\rangle$, for some $i \in \{1,\dots,r\}\}$, $B_2 := \{g|g \in G, \deg_U(\mathrm{lpp}_A(g)) = 0\}$, $G' := \{g|g \in G\backslash(B_1 \cup B_2)\}$ and $\{h_1,\dots,h_l\} := \{\mathrm{lc}_{\bar{A}}(g)|\,\mathrm{lc}_{\bar{A}}(g) \notin K, g \in G'\} \subseteq K[\bar{A}]$.
Then for each $\bar{a} \in \mathbb{V}(S')\backslash\,\mathbb{V}(h)$, $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis of $\langle\sigma_{\bar{a}}(F)\rangle$ in $L[\bar{X}]^r$ with respect to $\succ_m$ where $h = \mathsf{LCM}(h_1,\dots,h_l)$. Actually, we have $\sigma_{\bar{a}}(\sigma^1(G)) = \sigma_{\bar{a}}(\sigma^1(G'))$.

Proof. Note that any vector of G' has a linear form of U, i.e., the degree of U is at most 1.
It is clearly that $\sigma^1(G)$ is a basis of $\langle F\rangle$ by Lemma 9.3.3. We prove that $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis of $\langle\sigma_{\bar{a}}(F)\rangle$. For $\bar{a} \in \mathbb{V}(S')\backslash\,\mathbb{V}(h)$, we have $\sigma_{\bar{a}}(\mathrm{lc}_{\bar{A}}(g)) \neq 0$ for each $g \in G'$.
By the definition of G', B_1 and B_2, we have $G = G' \cup B_1 \cup B_2$. For each $f \in B_1$, f can be written as $f = U\cdot f_1e_i + f_2e_i$ where $f_1, f_2 \in K[\bar{A}]$ for some $i \in \{1,\dots,r\}$. By Lemma 9.3.3, $\sigma^0(f) = f_2e_i \in \langle S\rangle$, thus $\sigma_{\bar{a}}(f_2e_i) = 0$. By the definition of B_1, $\mathrm{lc}_{\bar{A}}(f) = f_1 \in \langle S'\rangle$, so $\sigma_{\bar{a}}(f_1) = 0$. Hence, $\sigma_{\bar{a}}(f) = 0$.
For each $q \in B_2$, by Lemma 9.3.3, $\sigma^0(q) = q \in \langle S\rangle$. Thus $\sigma_{\bar{a}}(q) = 0$.
Even if we change a module order $\succ_m$ into a hybrid module order $\succ_{hm2}$ in Theorem 9.2.6 ($R = K[\bar{A}]$), the properties of Theorem 9.2.6 hold. Thus, $\sigma_{\bar{a}}(G) = \sigma_{\bar{a}}(G\backslash(B_1 \cup B_2)) = \sigma_{\bar{a}}(G')$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(U\cdot F \cup (U-1)\cdot S)\rangle$ with respect to the $\succ_{hm2}$ in $L[U,\bar{X}]^r$.
For $g \in G'$, g can be written as $g = U\cdot g_1 + g_2$ where $g_1, g_2 \in K[\bar{A}][\bar{X}]^r$. By Lemma 9.3.3, we have $\sigma^0(g) = g_2 \in \langle S\rangle$, thus $\sigma_{\bar{a}}(g_2) = 0$. Namely, we have $\sigma_{\bar{a}}(g) = \sigma_{\bar{a}}(U\cdot g_1)$. Since every power product of $\sigma_{\bar{a}}(G')$ has a variable U whose degree is 1 and $U \gg \bar{X}$, $\sigma^1(\sigma_{\bar{a}}(G'))$ is a Gröbner basis of $\langle\sigma^1(\sigma_{\bar{a}}(U\cdot F) \cup (U-1)\cdot S\rangle = \langle\sigma^1(\sigma_{\bar{a}}(U\cdot F))\rangle = \langle\sigma_{\bar{a}}(F)\rangle$. Therefore, it follows that $\sigma_{\bar{a}}(\sigma^1(G))$ is a Gröbner basis for $\langle\sigma_{\bar{a}}(F)\rangle$ in $L[\bar{X}]^r$. ☐

Theorem 9.3.5 leads us to have the following algorithm which outputs a faithful comprehensive Gröbner system on L^m for $\langle F\rangle$.

Algorithm 9.3.6. $\mathsf{FCGSM}(F, \succ_m)$ (Faithful CGSs for Modules)

Input F: a finite set of vectors in $K[\bar{A}][\bar{X}]^r$,
$\succ_m$: a module order on $\mathrm{pp}(\bar{X})^r$,
Output G: a faithful comprehensive Gröbner system on L^m for $\langle F\rangle$ with respect to $\succ_m$.
begin
$H \leftarrow \mathsf{Gr\ddot{o}bnerBasisM}(F, \succ_m)$
if $e_1,\dots,e_r \in H$ **then**
$G \leftarrow \{(\emptyset, \{1\}, H)\}$
end-if
$S \leftarrow \{h_1,\dots,h_l\} := \{q|q \in \mathsf{factorize}(\mathrm{lc}_{\bar{A}}(g)), \mathrm{lc}_{\bar{A}}(g) \notin K, g \in H\}$
if $S \neq \emptyset$ **then**
$h \leftarrow \mathsf{LCM}(h_1,\dots,h_l)$
$G \leftarrow \{(\emptyset, h, H)\} \cup \mathsf{CGBMainM}(H, \{h_1e_1,\dots,h_1e_r\}, \{h_1\}, \succ_m) \cup$
$\cdots \cup \mathsf{CGBMainM}(H, \{h_le_1,\dots,h_le_r\}, \{h_l\}, \succ_m)$
else
$G \leftarrow \{(\emptyset, \{1\}, H)\}$
end-if
`return`(G)
end

Algorithm 9.3.7. $\mathsf{CGBMainM}(F, S, Z, \succ_m)$

Input F: a finite set of $K[\bar{A}][\bar{X}]^r$,
S: a finite set of vectors such that $\forall q \in S$, $q \in K[\bar{A}]e_i$,
Z: a finite set of polynomials in $K[\bar{A}]$,
$\succ_m$: a module order on $\mathrm{pp}(\bar{X})^r$,
Output G: a finite set of triples which forms a faithful comprehensive Gröbner system on $\mathbb{V}(Z)$ for $\langle F \rangle$.
begin
$H \leftarrow \mathsf{GröbnerBasisM}(U \cdot F \cup ((U-1) \cdot S, \succ_{hm2}))$ where $\succ_{hm2} := (U, \succ_m)$
$C \leftarrow$ the reduced Gröbner basis for $\langle Z \rangle$ in $K[\bar{A}]$
if $1 \in C$ **then**
$G \leftarrow \emptyset$
end-if
$B_1 \leftarrow \{g | g \in H \cap K[\bar{A}][U]e_i, \mathrm{lc}_{\bar{A}}(g) \in \langle Z \rangle,$ for some $i \in \{1, \ldots, r\}\}$
$B_2 \leftarrow \{g | g \in H, \deg_U(\mathrm{lpp}_A(g)) = 0\}$
$H' \leftarrow H \backslash (B_1 \cup B_2)$
$M \leftarrow \{\mathrm{lc}_{\bar{A}}(g) \mid g \in H'\}$
$L \leftarrow \{\beta_1, \ldots, \beta_l\} := \{q | q \in \mathtt{factorize}(g), g \notin K, g \in M\}$
if $L \neq \emptyset$ **then**
$\beta \leftarrow \mathsf{LCM}(\beta_1, \ldots, \beta_l)$
$G \leftarrow \{(Z, \beta, \sigma^1(H'))\} \cup \mathsf{CGBMainM}(F, S \cup \{\beta_1 e_1, \ldots, \beta_1 e_r\}, Z \cup \{\beta_1\}, \succ_m) \cup$
$\cdots \cup \mathsf{CGBMainM}(F, S \cup \{\beta_l e_1, \ldots, \beta_l e_r\}, Z \cup \{\beta_l\}, \succ_m)$
else
$G \leftarrow \{(Z, \{1\}, \sigma^1(H')\}$
end-if
$\mathsf{return}(G)$
end

Theorem 9.3.8. Let F be a finite set of vectors in $K[\bar{A}][\bar{X}]^r$. Then, the algorithm $\mathsf{FCGSM}(F)$ terminates. The output of FCGSM is a faithful comprehensive Gröbner system on L^m for $\langle F \rangle$.

Proof. In order to show the termination of the algorithm, it suffices to show that any of $\{\beta_j e_1, \ldots, \beta_j e_r\}$ is not in the submodule $\langle S \rangle$ because this algorithm is almost same as algorithm `CGSM` (see Theorem 9.2.12) (and we have $\sigma_{\bar{a}}(B_1) = \sigma_{\bar{a}}(B_2) = 0$ where $\bar{a} \in \mathbb{V}(S') \backslash \mathbb{V}(h)$). All notations of this proof is from the algorithm $\mathsf{CGBMainM}$.
By the construction of β_j, there exists $g \in H$ (which is from $\mathsf{GröbnerBasisM}(UF \cup (U - 1)S)$)) such that $\beta_j = \mathrm{lc}_{\bar{A}}(g)$, $\mathrm{lpp}_{\bar{A}}(g) \notin \mathrm{pp}(\bar{X})^r$. Therefore g can be written as

$$g = \beta_j UT + g_1,$$

where $T \in \mathrm{pp}(\bar{X})^r$, $\mathrm{lpp}_{\bar{A}}(g) = U \cdot T$ and $g_1 \in K[\bar{A}][U, \bar{X}]^r$. If $\beta_j e_1, \ldots, \beta_j e_r \in \langle S \rangle$, then $\beta_j e_i \cdot (U - 1) \in \langle G \rangle$ where $1 \leq i \leq r$. Hence, $\mathrm{lm}_{\bar{A}}(\beta_j e_i \cdot (U-1)) = \mathrm{lm}_{\bar{A}}(\beta_j e_i \cdot U)$ must be reduced by G. In the algorithm $\mathsf{GröbnerBasisM}$, we compute the reduced Gröbner basis for $\langle U \cdot F \cup (U-1) \cdot S \rangle$ in $K[\bar{A}, U, \bar{X}]^r$ with respect to a hybrid module order 2 "$\succ_{hm2} = (U, \succ_{hm1})$" where $\succ_{hm1} = (\succ_m, \succ_1)$ is a hybrid module order 1. Since G is the reduced Gröbner basis in $K[\bar{A}, U, \bar{X}]^r$, this is the contradiction. Therefore, $\beta_j e_i$ is not in the submodule $\langle S \rangle$.
It is an easy consequence of Theorem 9.2.12 and Lemma 9.3.3 that the output of FCGSM

is a faithful comprehensive Gröbner system on L^m for $\langle F \rangle$. □

Algorithm 9.3.9. $\mathsf{CGBM}(F, \succ_m)$ (Comprehensive Gröbner Bases for Modules)

Input F: a finite set of vectors in $K[\bar{A}][\bar{X}]^r$,
  $\succ_m$: a module order on $\mathrm{pp}(\bar{X})^r$,
Output G: a comprehensive Gröbner basis for $\langle F \rangle$ with respect to $\succ_m$.
begin
  $G \leftarrow \emptyset$
  $H \leftarrow \mathsf{FCGSM}(F, \succ_m)$
while $H \neq \emptyset$
  select (h_1, h_2, G_1) from H;
  $H \leftarrow H \backslash \{(h_1, h_2, G_1)\}$
  $G \leftarrow G \cup G_1$
end-while
return(G)
end

This algorithm CGBM has been implemented in the computer algebra system Risa/Asir by the author. In the following example, we can see an example of comprehensive Gröbner bases.

Example 9.3.10. Let x, y be variables and a, b parameters. We have $f_1 = \begin{pmatrix} ax - bx + 1 \\ ax^2y + ax + b \end{pmatrix}$ and $f_2 = \begin{pmatrix} by + a \\ bx^2 + bx + 2 \end{pmatrix}$ in $\mathbb{Q}[a, b][x, y]^2$. Then, our program outputs the following list which is a comprehensive Gröbner bases for $\langle f_1, f_2 \rangle$ with respect to $(POT, \succ_{lex})$ such that $x \succ_{lex} y$.

```
[0,(-b*a+b^2)*x^3+(b*a*y^2+a^2*y-b*a+b^2-b)*x^2+(b*a*y+a^2
  -2*a+b)*x+b^2*y+b*a-2]
[b*y+a,b*x^2+b*x+2]
[(a-b)*x+1,a*y*x^2+a*x+b]
[(-b*y-b)*x+1,-b*x^3+(a*y-b)*x^2+(a-2)*x+b]
```

This means;

$$\left\{ \begin{pmatrix} 0 \\ (**) \end{pmatrix}, \begin{pmatrix} by + a \\ bx^2 + bx + 2 \end{pmatrix}, \begin{pmatrix} (a-b)x + 1 \\ ayx^2 + ax + b \end{pmatrix}, \right.$$
$$\left. \begin{pmatrix} (-by - b)x + 1 \\ -bx^3 + (ay - b)x^2 + (a-2)x + b \end{pmatrix} \right\},$$

where $(**) := (-ba + b^2)x^3 + (bay^2 + a^2y - ba + b^2 - b)x^2 + (bay + a^2 - 2a + b)x + b^2y + ba - 2$.

9.4 Applications

Here we treat an application of comprehensive Gröbner systems (or comprehensive Gröbner bases). Especially, we consider **syzygies of parametric polynomials (or vectors)**.

Definition 9.4.1 (Comprehensive Syzygy systems). Let $f_1, \ldots, f_k$ be vectors in indexcomprehensive syzygy systems $K[\bar{A}][\bar{X}]^r$, $\mathcal{A}_1, \ldots, \mathcal{A}_l$ be algebraically constructible subsets of L^m and $G_1, \ldots, G_l$ be subsets of $K[\bar{A}][\bar{X}]^r$. Let $\mathcal{S}$ be a subset of L^m

such that $\mathcal{S} \subseteq \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_l$. A finite set $\mathcal{G} = \{(\mathcal{A}_1, G_1), \ldots, (\mathcal{A}_l, G_l)\}$ of pairs is called a **comprehensive syzygy system** on $\mathcal{S}$ for $\{f_1, \ldots, f_k\}$ if $\sigma_{\bar{a}}(G_i)$ is a basis of a syzygy module of $\{\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k)\}$ for each $i = 1, \ldots, l$ and $\bar{a} \in \mathcal{A}_i$, i.e. $\langle \sigma_{\bar{a}}(G_i) \rangle = \mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k))$ on $\mathcal{A}_i$. Each $(\mathcal{A}_i, G_i)$ is called a segment of $\mathcal{G}$. We simply say $\mathcal{G}$ is a comprehensive syzygy system of F if $\mathcal{S} = L^m$.

Suppose that $\mathcal{G}$ is a comprehensive syzygy of $\{f_1, \ldots, f_k\} \subset K[\bar{A}][\bar{X}]^r$. Then, for a segment $(\mathcal{A}_i, G_i) \in \mathcal{G}$, $\begin{pmatrix} g_1 \\ \vdots \\ g_k \end{pmatrix} \in \langle G_i \rangle \subseteq K[\bar{A}][\bar{X}]^k$ satisfies

$$\sum_{j=1}^{k} \sigma_{\bar{a}}(g_j)\sigma_{\bar{a}}(f_j) = 0,$$

where $\bar{a} \in \mathcal{A}_i$.

Lemma 9.4.2. Let $F = \{f_1, \ldots, f_l\}$ be a set of vectors in $K[\bar{A}][\bar{X}]^r$, S a set of polynomials in $K[\bar{A}]$ and $T = \{se_i | s \in S, 1 \le i \le r\}$. Furthermore, $G = \{g_1, \ldots, g_s\}$ be a Gröbner basis of a submodule $\langle F \cup T \rangle \subseteq K[\bar{A}][\bar{X}]^r$ with respect to $(POT, \succ)$ where $\succ$ is a term order on $\mathrm{pp}(\bar{X})$. Suppose that $B =: \{q | q \in G \cap K[\bar{A}]e_i, \mathrm{lc}_{\bar{A}}(g) \in \langle S \rangle, 1 \le i \le r\}$, $\{h_1, \ldots, h_l\} = \{\mathrm{lc}_{\bar{A}}(g) \,|\, \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G \backslash B\} \subseteq K[\bar{A}]$ and
$h = \mathsf{LCM}(h_1, \ldots, h_k)$. We have, for any $s = 0, \ldots, r-1$,

$$G' := G \cap \bigoplus_{i=s+1}^{r} K[\bar{A}][\bar{X}]e_i, \text{ and } F' := \langle F \rangle \cap \bigoplus_{i=s+1}^{r} K[\bar{A}][\bar{X}]e_i.$$

Then, for $\bar{a} \in \mathbb{V}(S) \backslash \mathbb{V}(h)$, $\sigma_{\bar{a}}(G')$ is a Gröbner basis for $\langle \sigma_{\bar{a}}(F') \rangle$ with respect to $(POT, \succ)$.

Proof. By Theorem 9.2.6 and Corollary 9.2.8, $\sigma_{\bar{a}}(G)$ is a Gröbner basis of $\langle \sigma_{\bar{a}}(F) \rangle$ for $\bar{a} \in \mathbb{V}(S) \backslash \mathbb{V}(h)$. Let $b \in \langle F' \rangle$, then we have to prove that there exists $f \in G'$ such that $\mathrm{lm}(\sigma_{\bar{a}}(f)) | \mathrm{lm}(\sigma_{\bar{a}}(b))$.
Since $\sigma_{\bar{a}}(G \backslash B)$ is a Gröbner basis of $\sigma_{\bar{a}}(F)$ in $K[\bar{A}]^r$, there exists $f \in \sigma_{\bar{a}}(G \backslash B)$ such that $\mathrm{lm}(f)$ divides $\sigma_{\bar{a}}(b)$. In particular,
$\mathrm{lm}(\sigma_{\bar{a}}(f)) \in \bigoplus_{i=s+1}^{r} K[\bar{X}]e_i$. Actually, by the assumption, we have $\mathrm{lm}(\sigma_{\bar{a}}(f)) = \sigma_{\bar{a}}(\mathrm{lm}(f))$ and $\sigma_{\bar{a}}(f) \neq 0$. Therefore, by these facts and the definition of the module order $(POT, \succ)$, we obtain $f \in \bigoplus_{i=s+1}^{r} K[\bar{X}]e_i$. In particular, $f \in G'$. □

Theorem 9.4.3. Let $F = \{f_1, \ldots, f_k\}$ be a set of vectors in $K[\bar{A}][\bar{X}]^r$. Consider the canonical embedding

$$K[\bar{A}][\bar{X}]^r \subseteq K[\bar{A}][\bar{X}]^{r+k}$$

and the canonical projection

$$\pi : K[\bar{A}][\bar{X}]^{r+k} \to K[\bar{A}][\bar{X}]^k.$$

Let S be a set of polynomials in $K[\bar{A}]$ and $T = \{se_i | s \in S, 1 \le i \le r\}$. Furthermore, $G = \{g_1, \ldots, g_s\}$ be a Gröbner basis of a submodule $\langle \{f_1 + e_{r+1}, f_2 + e_{r+2}, \ldots, f_k + e_{r+k}\} \cup T \rangle \subseteq K[\bar{A}][\bar{X}]^r$ with respect to $(POT, \succ)$, $B =: \{q | q \in G \cap K[\bar{A}]e_i, \mathrm{lc}_{\bar{A}}(g) \in \langle S \rangle, 1 \le i \le r\}$, $\{h_1, \ldots, h_l\} = \{\mathrm{lc}_{\bar{A}}(g) \,|\, \mathrm{lc}_{\bar{A}}(g) \notin K, g \in G \backslash B\} \subseteq K[\bar{A}]$ and $h = \mathsf{LCM}(h_1, \ldots, h_k)$ where $\succ$ is a term order on $\mathrm{pp}(\bar{X})$.
Suppose that $\{g_1, \ldots, g_l\} = \{G \backslash B\} \cap \bigoplus_{i=r+1}^{r+k} K[\bar{A}][\bar{X}]e_i$, then for
$\bar{a} \in \mathbb{V}(S) \backslash \mathbb{V}(h)$,

$$\mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k)) = \langle \sigma_{\bar{a}}(\pi(g_1)), \ldots, \sigma_{\bar{a}}(\pi(g_l)) \rangle.$$

Proof. We know that $\sigma_{\bar{a}}(\{f_1 + e_{r+1}, f_2 + e_{r+2}, \ldots, f_k + e_{r+k}\} \cup T) = \sigma_{\bar{a}}(f_1) + e_{r+1}, \ldots, \sigma_{\bar{a}}(f_k) + e_{r+k}$. Let us define $F' := \langle \sigma_{\bar{a}}(f_1) + e_{r+1}, \ldots, \sigma_{\bar{a}}(f_k) + e_{r+k} \rangle$. First, we prove that $\pi(F' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i) = \mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k))$.

($\subseteq$) Let $h \in F' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$, that is,

$$h = \sum_{v=r+1}^{r+k} h_v e_v = \sum_{j=1}^{k} b_j(\sigma_{\bar{a}}(f_j) + e_{r+j}),$$

for suitable $b_j \in K[\bar{X}]$. This implies that $\sum_{j=1}^{k} b_j \sigma_{\bar{a}}(f_j) = 0$ and $b_j = h_{r+j}$. Therefore, $\pi(h) \in \mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k))$.

($\supseteq$) Let $h = (h_1, \ldots, h_k) \in \mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k))$, that is, we have $\sum_{v=1}^{k} h_v \sigma_{\bar{a}}(f_v) = 0$. Then $h' = \sum_{v=1}^{k} h_v(\sigma_{\bar{a}}(f_v) + e_{r+v}) \in F' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]$. Obviously, $h = \pi(h') \in \pi(F' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i)$.

Therefore, $\pi(F' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i) = \mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k))$.

Next, we have to consider generators of $\mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k))$. By Lemma 9.4.2, $\sigma_{\bar{a}}(G \backslash B) \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$ is a Gröbner basis of $F' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i$. Therefore,

$$\begin{aligned}
\mathrm{Syz}(\sigma_{\bar{a}}(f_1), \ldots, \sigma_{\bar{a}}(f_k)) &= \pi(F' \cap \bigoplus_{i=r+1}^{r+k} K[\bar{X}]e_i) \\
&= \pi(\langle \sigma_{\bar{a}}(G) \rangle \cap \bigoplus_{i=r+1}^{r+k} (K[\bar{X}])e_i) \\
&= \langle \pi(\sigma_{\bar{a}}(g_1)), \ldots, \pi(\sigma_{\bar{a}}(g_l)) \rangle \\
&= \langle \sigma_{\bar{a}}(\pi(g_1)), \ldots, \sigma_{\bar{a}}(\pi(g_l)) \rangle.
\end{aligned}$$

□

By this theorem, we can apply the algorithm `CGSM` for constricting an algorithm for computing comprehensive syzygy systems. Note that in Lemma 9.4.2 the module order is not a hybrid module order.

Algorithm 9.4.4. $\mathsf{CSS}(\{f_1, \ldots, f_k\}, \succ_m)$ (Comprehensive Syzygy Systems)

Input: $f_1, \ldots, f_k$: vectors in $K[\bar{A}][\bar{X}]^r$,
 $\succ_m$: a module order on $\mathrm{pp}(\bar{X})^r$,
Output: H: a comprehensive syzygy system for $\langle f_1, \ldots, f_k \rangle$ on L^m.
begin
 $H \leftarrow \emptyset$
$F \leftarrow \{f_1 + e_{r+1}, \ldots, f_k + e_{r+k}\}$
 $D \leftarrow \mathsf{CGSM}(F, \succ_m)$
while $D \neq \emptyset$ **do**
 Select $(h1, h2, G)$ from D
 $D \leftarrow D \backslash \{(h1, h2, G1)\}$
 $H \leftarrow H \cup \left\{ \left(h1, h2, \pi \left(G \cap \bigoplus_{i=r+1}^{r+k} K[\bar{A}][\bar{X}]e_i \right) \right) \right\}$ (see below (∗))
end-while
`return`(H)

end
((*) π is the canonical projection $\pi : K[\bar{A}][\bar{X}]^{r+k} \to K[\bar{A}][\bar{X}]^k$, $(a_1, ., a_r, a_{r+1}, .., a_{r+k}) \mapsto (a_1, \ldots, a_r)$.)

Theorem 9.4.5. The algorithm CSS (Comprehensive Syzygy Systems) terminates for any input of a finite subset F of $K[\bar{A}][\bar{X}]^r$. If H is the output of CSS$(F, \succ_m)$, then H is a comprehensive syzygies system for $\langle F \rangle$ on L^m.

Proof. Since the algorithm CGSM terminates by Theorem 9.2.12, the algorithm CSS terminates. By Theorem 9.4.3 and algorithm CGSM, H is a comprehensive syzygies system for F on L^m. ☐

The algorithm CSS has been implemented in the computer algebra systems singular [GMPS05] and Risa/Asir by the author.

Example 9.4.6. Let x, y be variables and a, b parameters. We have $f_1 = ay^2 + x + 1$, $f_2 = bx + b$ and $f_3 = xy + a$. Then, our program in singular outputs a comprehensive syzygy system of $\{f_1, f_2, f_3\}$ as follows:

```
[b]==0, [a]!=0
_[1]=[0,1]
_[2]=[xy+a,0,-y2a-x-1]

[a,b]==0, [1]!=0
_[1]=[0,1]
_[2]=[xy,0,-x-1]

[a]==0, [b]!=0
_[1]=[0,xy,-xb-b]
_[2]=[b,-1]

[0]==0, [a,b]!=0
_[1]=[0,xy+a,-xb-b]
_[2]=[yb-ab,-y3a-y+a,y2ab]
_[3]=[xb+b,-y2a-x-1]
```

This meaning is the following:

$$\begin{cases} \{[0,1,0], [xy+a, 0, -ay^2-x-1]\} & \text{if } b=0, a\neq 0, \\ \{[0,1,0], [xy, 0, -x-1]\} & \text{if } a=b=0, \\ \{[0, xy, -bx-b], [b, -1, 0]\} & \text{if } a=0, b\neq 0, \\ \{[0, xy+a, -bx-b], [by-ab, -ay^3-y+a, aby^2], & \text{if } ab\neq 0. \\ \quad [bx+b, -ay^2-x-1, 0]\} & \end{cases}$$

That is;

When $b = 0$ and $a \neq 0$, then two vectors $\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} xy+a \\ 0 \\ -ay^2-x-1 \end{pmatrix}$ cover all syzygies of $\{f_1, f_2, f_3\}$. Obviously, we have the following.

$$\begin{cases} 0 = 1 \cdot f_2, \\ 0 = (xy+a)f_1 + (-ay^2-x-1)f_3. \end{cases}$$

When $a=b=0$, then two vectors $\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} xy \\ 0 \\ -x-1 \end{pmatrix}$ cover all syzygies of $\{f_1, f_2, f_3\}$, and these two vectors satisfy

$$\begin{cases} 0 = 1 \cdot f_2, \\ 0 = xyf_1 + (-x-1)f_3. \end{cases}$$

When $a=0$ and $b \neq 0$, then two vectors $\begin{pmatrix} 0 \\ xy \\ -bx-b \end{pmatrix}, \begin{pmatrix} b \\ -1 \\ 0 \end{pmatrix}$ cover all syzygies of $\{f_1, f_2, f_3\}$, and these two vectors satisfy

$$\begin{cases} 0 = xyf_2 + (-bx-b)f_3, \\ 0 = bf_1 + (-1)f_2. \end{cases}$$

When $ab \neq 0$, then vectors $\begin{pmatrix} 0 \\ xy+a \\ -bx-b \end{pmatrix}, \begin{pmatrix} by-ab \\ -ay^3-y+a \\ aby^2 \end{pmatrix}, \begin{pmatrix} bx+b \\ -ay^2-x-1 \\ 0 \end{pmatrix}$ cover all syzygies of $\{f_1, f_2, f_3\}$, and these three vectors satisfy

$$\begin{cases} 0 = (xy+a)f_2 + (-bx-b)f_3, \\ 0 = (by-ab)f_1 + (-ay^3-y+a)f_2 + (aby^2)f_3 \\ 0 = (bx+b)f_1 + (-ay^2-x-1)f_2. \end{cases}$$

Example 9.4.7. Let $f_1 = x^2+ay, f_2 = x+b, f_3 = bx+y$ be polynomials in $\mathbb{C}[a,b][x,y]$ where a, b are parameters. We have the lexicographic order $\succ$ such that $x \succ y$. Then, the program in Risa/Asir outputs the following as a comprehensive syzygy system for $\{f_1, f_2, f_3\}$.

```
[0]==0,  (b)*(a+1)!=0,
[-b^2*x-b^3,y^2+b^2*a*y,b*x^2+(-y+b^2)*x-b*y]
[-b*y+b^3,-y^2-b^2*a*y,(y-b^2)*x+(b*a+b)*y]
[0,b*x+y,-x-b]

[b]==0,  (1)!=0,
[1,-x,-a]
[0,y,-x]

[a+1]==0,  (b)!=0,
[b,y,-x]
[0,b*x+y,-x-b]
```

This output means the following

$$\begin{cases} \left\{ \begin{pmatrix} -b^2x-b^3 \\ y^2+b^2ay \\ bx^2+(-y+b^2)x-by \end{pmatrix}, \begin{pmatrix} -by+b^3 \\ -y^2-b^2ay \\ (y-b^2)x+(ba+b)y \end{pmatrix}, \begin{pmatrix} 0 \\ bx+y \\ -x-b \end{pmatrix} \right\}, & \text{if } b(a+1) \neq 0, \\ \left\{ \begin{pmatrix} 1 \\ -x \\ -a \end{pmatrix}, \begin{pmatrix} 0 \\ y \\ -x \end{pmatrix} \right\} & \text{if } b=0, \\ \left\{ \begin{pmatrix} b \\ y \\ -x \end{pmatrix}, \begin{pmatrix} 0 \\ bx+y \\ -x-b \end{pmatrix} \right\}, & \text{if } a+1=0, b \neq 0. \end{cases}$$

In the algorithm CSS, we apply the algorithm CGSM for computing comprehensive syzygy systems. The algorithm CGSM computes comprehensive Gröbner **systems**.

Probably, one has the following question.

Question:

Can we change algorithm CGSM into CGBM (or FCGS) in the algorithm CSS for computing "a module of syzygy" (not system) like comprehensive Gröbner bases as follows?

Algorithm 9.4.8. CSB($\{f_1,\ldots,f_k\},\succ_m$) (Comprehensive Syzygy bases)

Input: $f_1,\ldots,f_k$: vectors in $K[\bar{A}][\bar{X}]^r$,
$\succ_m$ a module order on $\mathrm{pp}(\bar{X})^r$,
begin
$F \leftarrow \{f_1+e_{r+1},\ldots,f_k+e_{r+k}\}$
$\boxed{H \leftarrow \mathsf{CGBM}(F,\succ_m)} \quad \Leftarrow ?$
$G \leftarrow \pi(H \cap \bigoplus_{i=r+1}^{r+k} K[\bar{A}][\bar{X}]e_i)$
end-while
return(G)
end

The answer is **"NO"**.
Actually, we can obtain some of (parametric) syzygies from the algorithm CSB. However, the outputs of the algorithm can not cover all syzygies of the input F under specialization $\sigma_{\bar{a}}$ for any $a \in L^m$. Consider the following.

Let a,b be parameters and x,y variables and $f_1 = x^2+ax+b, f_2 = 2x+2, f_3 = xy+a$ in $\mathbb{Q}[a,b][x,y]$. We have the graded reverse lexicographic order $\succ$ such that $x \succ y$. Then our program which computes a comprehensive Gröbner basis, returns the following list G as a comprehensive Gröbner basis G of $\langle (x^2+ax+b)e_1+e_2, (2x+2)e_1+e_3, (xy+a)e_1+e_4\rangle$ (i.e., CGBM($\{(ax^2+x+b)e_1+e_2, (x+b)e_1+e_3, (xy+a)e_1+e_4\}), (POT,\succ)$);
$G =$

```
[[      2*x+2,       0,           1,            0],
 [  2*y-2*b-2,      -2,     x+y+a-1,           -2],
 [ -2*a+2*b+2,       2,      -x-a+1,            0], (**)
 [   -2*y+2*a,       0,          -y,            2],
 [          0,   2*x+2,  -x^2-a*x-b,            0],
 [          0, 2*y-2*a, a*x-b*y+a^2, -2*x-2*a+2*b],
 [          0,       0,      -y*x-a,        2*x+2]].
```

(Note that $[x1,x2,x3,x4]$ means $x1e_1+x2e_2+x3e_3+x4e_4$.)

By the output $\pi(G \cap \bigoplus_{i=2}^{4} \mathbb{Q}[\bar{A}][\bar{X}]e_i)$, we can obtain the following list SY of vectors.
$SY =$

```
[[   2*x+2,  -x^2-a*x-b,             0],
```

```
 [ 2*y-2*a, a*x-b*y+a^2, -2*x-2*a+2*b],
 [       0,      -y*x-a,        2*x+2]].
```

Does this list SY cover the all syzygies of $\{\sigma_{\bar{a}}(f_1), \sigma_{\bar{a}}(f_2), \sigma_{\bar{a}}(f_3)\}$ where $\bar{a} \in \mathbb{Q}^2$?
If we take $a = 1, b = 0$, then
$\sigma_{\{1,0\}}(SY) =$

```
[[ 2*x+1,  -x^2-x,       0],
 [ 2*y-2,     x+1,  -2*x-2],
 [     0,  -x*y-1,    2*x+2]].
```

However, a Gröbner basis for a module of syzygies of
$\{\sigma_{\{1,0\}}(f_1), \sigma_{\{1,0\}}(f_2), \sigma_{\{1,0\}}(f_3)\}$ (with respect to $(POT, \succ)$) is the following list H.
$H =$

```
[[ 2,   -x,      0],
 [ 0,    y,     -2],
 [ 0, x*y+1, -2*x-2]].
```

Obviously, the submodule $\langle \sigma_{\{1,0\}}(SY) \rangle$ can not cover an element $[2, -x, 0]$. That is, we have $\langle \sigma_{\{1,0\}}(SY) \rangle \neq \langle H \rangle$.
Hence, the answer is "**NO**".

Why was this happened?
In the algorithm `CGBM`, we have to apply a hybrid module order. Therefore, even if the first coordinates of vectors of G become zero after substituting values into parameters, the first coordinates of G are not zero (by the order). Look at `(**)`, we have $[-2*a+2*b+2, 2, -x-a+1, 0]$=$(-2a+3b+2)e_1 + 2e_2 + (-x-a+1)e_3$. If we take $\{a = 1, b = 0\}$, then we obtain $[0, 2, -x, 0]$ from `(**)`. This vector can be a basis of syzygies of $\{\sigma_{\{1,0\}}(f_1), \sigma_{\{1,0\}}(f_2), \sigma_{\{1,0\}}(f_3)\}$. However, we can not compute this basis (vector) by algorithm the `CSB`.

In fact, if we apply a hybrid module order in Lemma 9.4.2 for computing Gröbner bases in $K[\bar{A}][\bar{X}]^r$, then Lemma 9.4.2 does not hold. In the proof of Lemma 9.4.2 we use the property of $(POT, \succ)$ which is not a hybrid module order. Therefore, Theorem 9.4.3 also does not hold if we apply a hybrid module order. We can not apply algorithm `CGBM` (and FCGSM) for computing syzygies of parametric vectors (or polynomials).

Remark: The computation of parametric syzygies cannot be computed by an immediate generalization to modules of a comprehensive Gröbner basis algorithm for syzygies; the remark that syzygies for a special case cannot be deduced from global syzygies is straightforward: consider any generic system, such that the generic element is regular (hence the global syzygies are the trivial ones) and a special case is non-regular (hence the trivial syzygies are a proper submodule). The additional syzygies have support in a proper non-dense subset of the parameter space, hence cannot be computed globally. The non-extensibility of syzygies is why comprehensive Gröbner bases are non-trivial (and would be useful in case that they could be computed in practice).

9.5 Reduced Gröbner bases in $K[\bar{A}][\bar{X}]^r$

In chapter 3, we saw the theory of reduced Gröbner bases in $K[\bar{A}][\bar{X}]$ and it's algorithms. By this algorithms, we can obtain reduced Gröbner bases in $K[\bar{A}][\bar{X}]$. Now we are considering $K[\bar{A}][\bar{X}]^r$. In the algorithm GröbnerBasisM, we need to compute a reduced Gröbner basis with respect to a hybrid module order 1 $\succ_{hm1}=(\succ_m,\succ_1)$ in $K[\bar{A},\bar{X}]^r$ where $\succ_m$ is a module order on $\mathrm{pp}(\bar{X})^r$ and $\succ_1$ is a term order on $\mathrm{pp}(\bar{A})$. However, the output of GröbnerBasisM is not always a reduced Gröbner basis in $K[\bar{A}][\bar{X}]^r$. That is, the output has sometimes some unnecessary vectors.

For example, let a,x,y be variables and $f_1=\begin{pmatrix} ax-x+y \\ 1 \end{pmatrix}$, $f_2=\begin{pmatrix} ay+a \\ x+a \end{pmatrix}$ vectors in $\mathbb{Q}[a][x,y]^2$. Then, the output of GröbnerBasisM$(\{f_1,f_2\},\succ_m)$ where $\succ_m=(TOP, x\succ_{lex} y\gg a)$, is the following;

$$g_1=\begin{pmatrix} xy+2x-y^2 \\ x^2+ax-y \end{pmatrix}, g_2=\begin{pmatrix} (a-1)x+y \\ 1 \end{pmatrix}, g_3=\begin{pmatrix} ax+2 \\ x+a \end{pmatrix},$$

$$g_4=\begin{pmatrix} 0 \\ (a-1)x^2+xy+(a^2-a)x-2 \end{pmatrix}.$$

By Lemma 9.2.3, we know that $\{g_1,g_2,g_3,g_4\}$ is a Gröbner basis for $\langle f_1,f_2\rangle$ in $\mathbb{Q}[a][x,y]^2$. However, there exists a nicer Gröbner basis, because we have

$$\mathrm{lm}_{\{a\}}(g_1)=xye_1\in\langle \mathrm{lm}_{\{a\}}(g_2),\mathrm{lm}_{\{a\}}(g_3)\rangle=\langle (a-1)xe_1, aye_1\rangle.$$

That is, g_1 can be written as

$$g_1=yg_2-xg_3.$$

This means that g_1 is an unnecessary vector. That is, $\{g_2,g_3,g_4\}$ is a Gröbner basis too. However we can not obtain this Gröbner basis $\{g_2,g_3,g_4\}$ by the algorithm GröbnerBasisM.

What is a reduced Gröbner basis in $K[\bar{A}][\bar{X}]^r$? The definition of reduced Gröbner bases in $K[\bar{A}][\bar{X}]^r$ is the following.

Definition 9.5.1 (Reduced Gröbner bases). Let $\succ_m$ be a module order $\mathrm{pp}(\bar{X})^r$, $\succ_1$ a term order on $\mathrm{pp}(\bar{A})$ and $\succ_{hm1}=(\succ_m,\succ_1)$ a hybrid module order 1 on $\mathrm{pp}(\bar{A},\bar{X})^r$. Let I be an ideal in $K[\bar{A}][\bar{X}]^r$. Then, a reduced Gröbner basis G for I with respect to $(\succ_m,\succ_1)$ is a Gröbner basis for I in $K[\bar{A}][\bar{X}]^r$ such that

1. $\mathrm{lc}(p)=1$ for all $p\in G$ with respect to $\succ_{hm1}$,
2. for all $p\in G$, no monomial in $\mathrm{Mono}_{\bar{A}}(p)$ lies in $\langle \mathrm{lm}_{\bar{A}}(G\setminus\{p\})\rangle$ in $K[\bar{A}][\bar{X}]^r$ with respect to $\succ_m$,
3. G is the reduced Gröbner basis for an ideal generated by itself $\langle G\rangle$ in $K[\bar{A},\bar{X}]^r$ with respect to $\succ_{hm1}$.

In order to compute reduced Gröbner bases in $K[\bar{A}][\bar{X}]^r$ we need two reduction systems which are the following.

Definition 9.5.2. We have a module order with a "hybrid module order 1" $\succ_{hm1}=(\succ_m,\succ_1)$ where $\succ_m$ is a module order on $\mathrm{pp}(\bar{X})^r$ and $\succ_1$ is a term order on $\mathrm{pp}(\bar{A})$. Let $f=a\alpha+f_1, g=b\alpha\beta+g_1$ with $\mathrm{lm}(f)=a\alpha$ in $K[\bar{A},\bar{X}]^r$ where $a,b\in K$, $\alpha,\beta\in\mathrm{pp}(\bar{A},\bar{X})^r$ and $f_1,g_1\in K[\bar{A},\bar{X}]^r$. Then a reduction $\xrightarrow{r1}_f$ is defined as follows:

$$g\xrightarrow{r1}_f b\alpha\beta+g_1-ba^{-1}\beta(a\alpha+f_1),$$

where $b\alpha\beta$ need not be the leading monomial of g. In this paper we call this reduction "Reduce1 ". A reduction $\xrightarrow{r1}_F$ by a set F of polynomials is also natural defined.

In order to compute reduced Gröbner bases in $K[\bar{A}][\bar{X}]^r$ we also need the following reduction system.

Definition 9.5.3. Let F be a finite set of vectors in $K[\bar{A}][\bar{X}]^r$ and $g = a\beta + g' \in K[\bar{A}][\bar{X}]^r$ where $a \in K[\bar{A}]$, $\beta \in \mathrm{pp}(\bar{X})^r$ and $g' \in K[\bar{A}][\bar{X}]^r$ (i.e. $a\beta \in \mathrm{Mono}_{\bar{A}}(g)$). Moreover, let

$$F' := \{f \in F \mid \mathrm{lpp}_{\bar{A}}(f) \text{ divides } \beta\}.$$

If $a \in \langle \mathrm{lc}_{\bar{A}}(F')\rangle$, a can be written as $a = \sum_{f_i \in F'} h_i \,\mathrm{lc}_{\bar{A}}(f_i)$ where $h_i \in K[\bar{A}]$. Then a reduction $\xrightarrow{r2}_f$ is defined as follows:

$$g \xrightarrow{r2}_{F'} g - \sum_{f_i \in F'} h_i \frac{\beta}{\mathrm{lpp}_{\bar{A}}(f_i)} f_i.$$

In this chapter, we define this reduction system as Reduce2 (written: $\xrightarrow{r2}$).

By the two reduction systems and the algorithm GröbnerBasisM, we can construct an algorithm for computing reduced Gröbner bases in $K[\bar{A}][\bar{X}]^r$.

Algorithm 9.5.4. RGB$(F, \succ_m)$ (Reduced Gröbner bases)

Input: F: a set of vectors in $K[\bar{A}][\bar{X}]^r$,
  $\succ_m$: a module order on $\mathrm{pp}(\bar{X})^r$,
Output: G: a reduced Gröbner basis for $\langle F\rangle$ with respect to $\succ_m$.
begin
  $G \leftarrow$ GröbnerBasisM$(F, \succ_m)$
  $E1 \leftarrow 0$
  $E2 \leftarrow 0$
while $E1 \neq 1$ **do**
  while $E2 \neq 2$ **do**
   if there exists $p \in G$ such that
    $\left(p \xrightarrow{r1}_{\{G\backslash\{p\}\}} p_1 \text{ and } p \neq p_1\right)$ or $\left(p \xrightarrow{r2}_{\{G\backslash\{p\}\}} p_1 \text{ and } p \neq p_1\right)$
   then
    $G \leftarrow \{G\backslash\{p\}\} \cup \{p_1\}$
   else
    $E2 \leftarrow 2$
   end-if
  if (G is **NOT** the reduced Göbner basis for $\langle G\rangle$ with respect to $\succ_{hm1}$ in $K[\bar{A}, \bar{X}]^r$)
  then
   $G \leftarrow$ Compute the reduced Gröbner basis for $\langle G\rangle$ with respect to $\succ_{hm1}$ in $K[\bar{A}, \bar{X}]^r$
   $E2 \leftarrow 0$
  else
   $E1 \leftarrow 1$
  end-if
  end-while
end-while
return(G)
end

Theorem 9.5.5. The algorithm RGB$(F, \succ_m)$ terminates. The output forms a reduced Gröbner basis for $\langle F\rangle$ with respect to a module order $\succ_m$ in $K[\bar{A}][\bar{X}]^r$.

Proof. In the first step, we compute a Gröbner basis for $\langle F \rangle$ with respect to $\succ_m$ in $K[\bar{A}][\bar{X}]^r$. This step obviously terminates.
In the second, if there exists an element $q \in G$ which can be reduced to q_1 by $\xrightarrow{r1}_{\{G\setminus\{p\}\}} p_1$ or $\xrightarrow{r2}_{\{G\setminus\{p\}\}} p_1$, then we have $\mathrm{lpp}_{\bar{A}}(q) \succeq_m \mathrm{lpp}_{\bar{A}}(q_1)$ ($\mathrm{lpp}_{\bar{A}}(q_1)$ is smaller or equal than $\mathrm{lpp}_{\bar{A}}(q)$ with respect to $\succ_m$). That is, the result of applying $\xrightarrow{r1}$ and $\xrightarrow{r2}$ to q in $K[\bar{A}][\bar{X}]^r$ or $K[\bar{A}, \bar{X}]^r$ has a leading power product which cannot be greater than $\mathrm{lpp}_{\bar{A}}(q)$ with respect to $\succ_m$. Therefore, iterated application of $\xrightarrow{r1}$ and $\xrightarrow{r2}$ to G will eventually terminate.
Next we check whether G is the reduced Gröbner basis for $\langle G \rangle$ with respect to $\succ_{hm1}$ in $K[\bar{A}][\bar{X}]^r$ or not. If G is the reduced Gröbner basis for $\langle G \rangle$ with respect to $\succ_{hm1}$ in $K[\bar{A}][\bar{X}]^r$, then this algorithm terminates. If not, we repeat the same procedure. We have the fact that "every infinite ascending chain $M_1 \subset M_2 \subset \cdots$ of submodules of $K[\bar{A}][\bar{X}]^r$ stabilizes." (see books [CLO97, GMP02]) Therefore, this algorithm terminates and outputs a reduced Gröbner basis with respect to $(\succ_m, \succ_1)$ in $K[\bar{A}][\bar{X}]^r$. □

In chapter 3, we defined two reduced Gröbner bases in $K[\bar{A}][\bar{D}]$. In the module $K[\bar{A}][\bar{X}]^r$, we can define two reduced Gröbner bases, too. In fact, we can regard Definition 9.5.1 as a definition of weak Gröbner bases in $K[\bar{A}][\bar{X}]^r$. In this chapter, we do not treat strong reduced Gröbner bases in $K[\bar{A}][\bar{X}]^r$. One can easily construct them by using the same way of chapter 3.

9.6 Concluding Remarks

In this chapter we have extended comprehensive Gröbner bases to modules, and gave algorithms to compute comprehensive Gröbner bases and comprehensive Gröbner systems for modules. Furthermore, these algorithms have been implemented in the computer algebra systems singular [GMPS05] and Risa/Asir [NT92] by the author. (In fact, we described the generalization of Suzuki-Sato algorithm (Algorithm 4.4.3 in chapter4).) It is possible to apply other existing algorithms for computing comprehensive Gröbner bases and comprehensive Gröbner systems. For example, Nabeshima's new approach for computing comprehensive Gröbner systems (in chapter 5), and the approach of ACGB. One can easily follow the algorithms for computing them.

In general, since comprehensive Gröbner bases are huge, comprehensive Gröbner bases in $K[\bar{A}][\bar{X}]^r$ are huge too. This means that we need high speed machines and a lot of memory (RAM) in the machines. However, our programs in the both the computer algebra systems singular and Risa/Asir still work for a lot of (easy) examples in $K[\bar{A}][\bar{X}]^r$ where $r \leq 3$, $|\bar{X}| \leq 3$ and $|\bar{A}| \leq 3$ (OS: WindowsXP, CPU: Pentium M 1.73GHz, Memory: 512MB RAM).

We can solve a lot of parametric problems by applying comprehensive Gröbner bases and comprehensive Gröbner systems, like Gröbner bases in $K[\bar{X}]$. One of the applications "syzygies of parametric polynomials (or vectors)" was shown. In the future work, we consider optimization techniques for getting small comprehensive Gröbner bases and applications of comprehensive Gröbner bases and comprehensive Gröbner systems more.

Chapter 10

Implementation: PGB package

We present a new software package in the computer algebra system Risa/Asir, named PGB, for computing parametric Gröbner bases in various domains and related objects. The purpose of this chapter is to illustrate how to use the package and solve problems using the package.

In the author's paper [Nab07b], this package was introduced.

One can download the package from the following website.

`http://www.risc.uni-linz.ac.at/people/knabeshi/pgb/`

One can also download the computer algebra system Risa/Asir from the following website.

`http://www.math.kobe-u.ac.jp/Asir/asir.html`

In this chapter, we describe the package PGB Version 1.0 (20070305). (The author will improve and update it on the website.)

When one loads the package, then Risa/Asir gives the following message;

```
----------------------------------------------------------
PGB, Version 1.0 (20070305)
Copyright (C) Katsusuke Nabeshima.
Research Institute for Symbolic Computation (RISC-Linz),
Johannes Kepler University Linz.
A-4040, Linz, Austria.
http://www.risc.uni-linz.ac.at/people/knabeshi/pgb/
----------------------------------------------------------
Welcome to PGB. Enjoy!
```

10.1 CGBs and CGSs in commutative polynomial rings

In this section we treat the commands of PGB for computing comprehensive Gröbner systems and comprehensive Gröbner bases in commutative polynomial rings.

10.1.1 Comprehensive Gröbner systems

Here, we introduce the commands for computing comprehensive Gröbner systems in commutative polynomial rings.

The package PGB has the following commands for computing comprehensive Gröbner systems in a commutative polynomial ring. The following commands which have the Suzuki-Sato algorithm, output a comprehensive Gröbner system. Each of them has different optimization techniques for getting comprehensive Gröbner systems.

Commands:

cgs(polylist,plist,vlist,option,*order*),
cgs1(polylist,plist,vlist,option,*order*),
cgs2(polylist,plist,vlist,option,*order*),
cgs_re(polylist,plist,vlist,option,*order*),
cgs_re1(polylist,plist,vlist,option,*order*).

Output:

a comprehensive Gröbner system for an ideal generated by polylist with respect to *order*.

- polylist : a list of polynomials.
- plist : a list of parameters.
- vlist : a list of variables.
- *order :* a term order on the set of power product of vlist, (see "**Setting term ordering**" of the manual [NST03]. Matrix orders are available).
- option : 1 or 0. This package PGB has two kinds of form for comprehensive Gröbner systems. One have to select **0** or **1**.

When we input "**0**" into option, then the programs output a list of segments of a comprehensive Gröbner basis as follows.

```
cgs1([a*x^2*y^2+b*x*y+2,b*x+a*y+2],[a,b],[x,y],0,2);
[[[b],[a],[-2*x^2-a,a*y+2]],[[b,a],[1],[1]],[[a],[b],[b*x+2,-y+1]],[[0],[
 b*a],[-a^3*y^4-4*a^2*y^3+(b^2-4)*a*y^2+2*b^2*y-2*b^2,b*x+a*y+2]]]
```

If we input "**1**" into option, then the programs output a list of segments of a comprehensive Gröbner system, too. However, this output form is more intuitive than the previous one "**0**". An example of this case is the following.

```
cgs1([a*x^2*y^2+b*x*y+2,b*x+a*y+2],[a,b],[x,y],1,2);
[b]==0,  [a]!=0,
[-2*x^2-a,a*y+2]
[b,a]==0,  [1]!=0,
[1]
[a]==0,  [b]!=0,
[b*x+2,-y+1]
[0]==0,  [b*a]!=0,
[-a^3*y^4-4*a^2*y^3+(b^2-4)*a*y^2+2*b^2*y-2*b^2,b*x+a*y+2]
Number of segments is 4
```

A part of outputs `[b,a]` means $\mathbb{V}(a,b)$. That is, `[b]==0, [a]!=0` means $\mathbb{V}(b)\setminus\mathbb{V}(a)$. Two outputs above are a comprehensive Gröbner system for $\langle ax^2y^2+bxy+2, bx+ay+2\rangle$

with respect to the lexicographic order $\succ$ such that $x \succ y$ where x, y are variables and a, b are parameters. The outputs mean the following;

$$\begin{cases} \{ay+2, -2x^2-a\}, & \text{if } \mathbb{V}(b)\setminus\mathbb{V}(a)\ [b=0, a\neq 0],\\ \{1\}, & \text{if } \mathbb{V}(a,b)\ [a=b=0],\\ \{-y+1, bx+2\} & \text{if } \mathbb{V}(a)\setminus\mathbb{V}(b)\ [a=0, b\neq 0]\\ \{-a^3y^4-4a^2y^3+(b^2-4)ay^2+2b^2y-2b^2, bx+ay+2\} & \text{if } \mathbb{C}^2\setminus\mathbb{V}(ab)\ [ab\neq 0]. \end{cases}$$

All the commands "cgs", "cgs1", "cgs2" "cgs_re" and "cgs_re",output a comprehensive Gröbner system. **What are differences?** The differences are the optimization techniques. In general, the outputs of comprehensive Gröbner basis or comprehensive Gröbner systems are very big. As we saw in chapter 4, we need a lot of optimization techniques for getting small and nice ones. Some optimization techniques have been implemented in the commands. However, concerning speed, some of the techniques are expensive. Hence, one can select a command from the list, which has some optimization techniques for computing comprehensive Gröbner systems. Main differences are the following three optimization techniques;

(1) computing the reduced Gröbner basis in each segment,
(2) checking parametric spaces (conditions of parameters) of all segments at the last step,
(3) computing a reduced Gröbner basis in $\mathbb{Q}[\bar{A}][\bar{X}]$ instead of computing the reduced Gröbner basis with respect to a block order with $\bar{X} \gg \bar{A}$ in $\mathbb{Q}[\bar{A}, \bar{X}]$. (See chapter 3.)

The commands "cgs1" and "cgs_re1" have the technique **(1)**. That is, the commands output the reduced Gröbner basis in each segment.

In fact, all of the commands check whether there exists any redundant segment or not. If there exist redundant segments, the programs remove the segments. However, this check system is not complete, this is rough. Therefore, when the commands finish computing, we can check again for getting a small and nice comprehensive Gröbner system. This technique is **(2)**. The commands "cgs", "cgs1", "cgs_re" and "cgs_re1" have this technique **(2)**.

All the commands follow the Suzuki-Sato algorithm [SS06]. In the Suzuki-Sato algorithm, we need to compute Gröbner bases in $\mathbb{Q}[\bar{A}][\bar{X}]$. As we saw in chapter 3, the algorithm Insa-Pauer (Algorithm 3.2.8) and the algorithm GröbnerBasisB (Algorithm 3.3.2) can not output a reduced Gröbner basis in $\mathbb{Q}[\bar{A}][\bar{X}]$. Therefore, we can apply the algorithm WRGB (Algorithm 3.5.2) for computing reduced Gröbner bases in $\mathbb{Q}[\bar{A}][\bar{X}]$. The commands "cgs_re" and "cgs_re1" have this technique **(3)**.

The following table shows us what techniques all the commands have.

command	techniques
cgs	**(2)**
cgs1	**(1), (2)**
cgs2	**none**
cgs_re	**(1), (2), (3)**

Concerning speed, the author recommends to use cgs (or cgs2) because sometimes computing reduced Gröbner bases in each segments and computing reduced Gröbner bases in $\mathbb{Q}[\bar{A}][\bar{X}]$ are very expensive.

In the following examples, we see how the commands work, and we compare some commands.

We compare the commands cgs and cgs1.

```
cgs([a*x^2*y^2+x+y,x*y+y^2,b*x^2+y],[a,b],[x,y],1,1);
[b]==0,  [1]!=0,
[x,y]
[a]==0,  [b]!=0,
[b*y^2+y,x+y]
[0]==0,  [b*a]!=0,
[a*y,x+y,b*y^2+y]
Number of segments is 3

cgs1([a*x^2*y^2+x+y,x*y+y^2,b*x^2+y],[a,b],[x,y],1,1);
[b]==0,  [1]!=0,
[x,y]
[a]==0,  [b]!=0,
[b*y^2+y,x+y]
[0]==0,  [b*a]!=0,
[a*y,x+y]
Number of segments is 3
```

We compare the commands cgs1 and cgs_re.

```
cgs1([a*x*z+b*x*z+a,b*z+a,(a^2+a)*x*y],[a,b],[x,y,z],0,0);
[[[b-1],[a^2+a],[(-a^2-a)*x+a,a*y,z+a]],[[b,a],[0],[]],[[a+b],[b,a],[1]],
 [[a+1],[b^2-b],[(b-1)*x-b,b*z-1]],[[a],[b],[b*z]],[[b],[a],[1]],[[0],[(b
 ^2-b)*a^3+(b^3-b)*a^2+(b^3-b^2)*a],[b*z+a,(-a^2-a)*y,(-a^2-b*a)*x+b*a]]]
(7 segments)
cgs_re([a*x*z+b*x*z+a,b*z+a,(a^2+a)*x*y],[a,b],[x,y,z],0,0);
[[[b,a],[0],[]],[[a+b],[b,a],[1]],[[a+1],[b^2-b],[(b-1)*x-b,b*z-1]],[[a],
 [b],[b*z]],[[b],[a],[1]],[[0],[b*a^3+(b^2+b)*a^2+b^2*a],[b*z+a,(-a^2-a)*
 y,(-a^2-b*a)*x+b*a,(a*z-a)*x+a]]]
(6 segments)
```

In chapter 5, we saw the algorithm NEW for computing comprehensive Gröbner systems. The algorithm has been also implemented. The commands for the algorithm NEW is the following.

Commands:

cgs_con(polylist,plist,vlist,num, option,*order*),
cgs_con1(polylist,plist,vlist,num, option,*order*),
cgs_con2(polylist,plist,vlist,num, option,*order*).

Output:

a comprehensive Gröbner system for an ideal generated by polylist. with respect to *order*.

- num : The program select a polynomial whose number of monomials is num (see chapter 5). If one inputs 0, then the program executes the algorithm NEW. If one inputs "-1" in cgs_con, then cgs works (the Suzuki-Sato algorithm). If one inputs -1 in cgs_con1, then cgs1 works (the Suzuki-Sato algorithm). If one inputs s which is a natural number, then the algorithm NEW[s] works (see chapter 5).
- option : 1 or 0. This package PGB has two kinds of form for comprehensive Gröbner systems. One have to select **0** or **1**.

The following table shows us which kinds of techniques three commands have.

command	techniques
cgs_con	**(2)**
cgs_con1	**(1), (2)**
cgs_con2	**none**

In the following examples, we see how the commands work.

```
cgs_con1([x^3*y+x+y,b*x*y+a*y+1],[a,b],[x,y],0,1,2);
[0]==0,  [[a],[a^3-b^3],[b]]!=0,
[b^3*a*x+(-b^2*a^4+b^5*a)*y^2+(-3*b^2*a^3-b^4*a^2)*y-2*b^2*a^2-b^4*a,(-a^
 3+b^3)*y^3+(-3*a^2-b^2*a)*y^2+(-3*a-b^2)*y-1]
[a]==0,  [[a^3-b^3],[b]]!=0,
[x+b^2*y^2-b,b^3*y^3-b^2*y-1]
[a-b]==0,  [[a^2+3*a],[b]]!=0,
[-x+(a^2+3*a)*y+a+2,(a^3+3*a^2)*y^2+(a^2+3*a)*y+1]
[a+3,b+3]==0,  [[b]]!=0,
[1]
[a,b]==0,  [[b]]!=0,
[1]
[a^2+b*a+b^2]==0,  [[a^3-3*a^2+9*a],[b]]!=0,
[(a^4-3*a^3+9*a^2)*x+((-b+3)*a^5+(6*b-9)*a^4+(-18*b+27)*a^3+27*b*a^2)*y+(
 -b+2)*a^4+(5*b-6)*a^3+(-15*b+18)*a^2+18*b*a,(a^4-3*a^3+9*a^2)*y^2+(a^3-3
 *a^2+9*a)*y+b+3]
[a^2-3*a+9,(b+3)*a+b^2-9]==0,  [[b+3],[b]]!=0,
[1]
[b]==0,  [[a]]!=0,
[x^3-a*x+1,-a*y-1]
Number of segments is 8
```

Remark that the meaning of `[F]==0,[[`t_1`], ... ,[`t_l`]]!=0,` is

$$\mathbb{V}(F)\setminus(\mathbb{V}(t_1)\cup\cdots\cup\mathbb{V}(t_l)).$$

In fact, $\mathbb{V}(F)\setminus(\mathbb{V}(t_1)\cup\cdots\cup\mathbb{V}(t_l)) = \mathbb{V}(F)\setminus\mathbb{V}(\mathsf{LCM}(t_1,t_2,\ldots,t_k))$.

The special command "sub4cgs" has been implemented for substituting arbitrary values for parameters of a comprehensive Gröbner system.

> **Commands:**
>
> sub4cgs(a comprehensive Gröbner system,[[a_1,v_1],[a_2,v_2],...,[a_l,v_l]])

Procedure:
(1) The program searches some segments $(\mathcal{A}_1, G_1), \ldots, (\mathcal{A}_l, G_l)$ of a comprehensive Gröbner system such that $(v_1, \ldots, v_l) \in \mathcal{A}_i$, $1 \leq i \leq l$.
(2) The program substitutes the values $v_1, \ldots, v_l$ for the parameters $a_1, \ldots, a_l$ of the set polynomials $G_1, \ldots, G_l$.
(3) The program **outputs** the parametric spaces, the original set of polynomials $G_1, \ldots, G_l$ and the set of polynomials $G_1(v_1, \ldots, v_l), \ldots, G_l(v_1, \ldots, v_l)$ computed.

- $a_1, a_2, \ldots, a_l$: parameters.
- $v_1, v_2, \ldots, v_l$: values.

In this example, we see how the command "sub4cgs" works.

```
B=cgs1([a*x^2*z^2+x*y^2+2*b,x^2+z^2+a,b*z+x],[a,b],[x,y,z],0,2);
[[[a,b^3+b],[0],[(b^2+1)*z^2,-b*z*y^2+2*b,(-2*b^4-2*b^2)*z,x+b*z]],[[a],[
 b^3+b],[1]],[[b],[1],[x,z^2+a]],[[a,b^2+1],[0],[z*y^2-2,x+b*z]],[[b^2+1]
 ,[a],[1]],[[0],[(b^3+b)*a],[(b^2+1)*z^2+a,-b*a*y^2-a^2*z^3+(-a^3-2*b^3-2
 *b)*z,x+b*z]]]

sub4cgs(B,[[a,3/4],[b,1/2]]);

[0]==0,  [(b^3+b)*a]!=0,
[(b^2+1)*z^2+a,-b*a*y^2-a^2*z^3+(-a^3-2*b^3-2*b)*z,x+b*z]
[5/4*z^2+3/4,-3/8*y^2-9/16*z^3-107/64*z,x+1/2*z]

sub4cgs(B,[[a,2],[b,0]]);
[b]==0,  [1]!=0,
[x,z^2+a]
[x,z^2+2]

C=cgs_con([a*x^2+b*y^2,c*x^2+y^2,2*a*x-2*c*y],[a,b,c],[x,y],0,0,2);
[[[0],[[a],[a-c*b]],[a*x-c*y,y^2]],[[a],[[c],[a-c*b]],[c*x^2,y]],[[a-c*b]
 ,[[a],[b^2+c]],[a*x-c*y,y^2]],[[a,c*b],[[c],[b^2+c]],[c*x^2,y]],[[a,c],[
 [b^2+c]],[y^2]],[[b^2+c,a-c*b,b*a+c^2],[[a]],[a*x-c*y]],[[a,b,c],[[1]],[
 y^2]]]

sub4cgs(C,[[a,-1],[b,-1/3],[c,0]]);
[0]==0,  [[a],[a-c*b]]!=0,
[a*x-c*y,y^2]
[-x,y^2]
```

10.1.2 Comprehensive Gröbner bases

The package PGB has the following commands for computing comprehensive Gröbner bases in a commutative polynomial ring. The following commands output a comprehensive Gröbner basis. Each of them has different optimization techniques for getting nice and small comprehensive Gröbner bases. Actually, these commands are related to the commands for computing comprehensive Gröbner systems.

Commands:

cgb(polylist,plist,vlist,*order*),
cgb1(polylist,plist,vlist,*order*),
cgb_re(polylist,plist,vlist,*order*).
cgb_re1(polylist,plist,vlist,*order*).

Output:

a comprehensive Gröbner basis for an ideal generated by polylist with respect to the *order*.

- polylist : a list of polynomials.
- plist : a list of parameters.
- vlist : a list of variables.
- *order :* a term order on the set of power product of vlist(see "**Setting term ordering**" of the manual [NST03]. Matrix orders are available).

All the commands "cgb", "cgb1", "cgb_re" and "cgb_re1" output a comprehensive Gröbner basis. **What are differences?** The differences are the optimization techniques.

The following table shows us what techniques all the commands have.

command	techniques
cgb	**none**
cgb1	**(2)**
cgb_re	**(3)**
cgb_re1	**(2), (3)**

(see section 10.1.1)

In the following examples, we see how the commands work.

```
cgb([a*x*z+x*z+a,b*z^2+a,(a^2+a)*x*y+b^2],[a,b],[x,y,z],2);
[(-z*y-b*a)*x-a*y+b^2*z,(-b^2*y+b^4*z)*x+b^3*z*y+b^4,(-a*y-b^2*z)*x-b^2,(
 -b^2*a-b^2)*x+b^3*z,(-a^2-a)*x+b*a*z,-a^2*y+b^2*z,(-b*z^3+z)*x-b*z^2,b*z
 ^2+a,-b*a*z*y-b^2,(a+1)*z*x+a]

cgb1([a*x*z+x*z+a,b*z^2+a,(a^2+a)*x*y+b^2],[a,b],[x,y,z],2);
[(-z*y-b*a)*x-a*y+b^2*z,(-b^2*y+b^4*z)*x+b^3*z*y+b^4,(-a*y-b^2*z)*x-b^2,(
 -b^2*a-b^2)*x+b^3*z,(-a^2-a)*x+b*a*z,-a^2*y+b^2*z,(-b*z^3+z)*x-b*z^2,b*z
 ^2+a,-b*a*z*y-b^2,(a+1)*z*x+a]

cgb_re([a*x*z+x*z+a,b*z^2+a,(a^2+a)*x*y+b^2],[a,b],[x,y,z],2);
[(-b^2*a-b^2)*x+b^3*z,(-a^2-a)*x+b*a*z,-a^2*y+b^2*z,(-b*z^3+z)*x-b*z^2,b*
 z^2+a,-b*a*z*y-b^2,(a+1)*z*x+a]
```

The special command "sub4cgb" has been implemented for substituting arbitrary values for parameters of a comprehensive Gröbner basis.

Commands:

sub4cgb(polylist,[[a_1,v_1],[a_2,v_2],...,[a_l,v_l]])

Output:
the set of polynomials which is substituted $(v_1, v_2, \ldots, v_l)$ for parameters $(a_1, a_2, \ldots, a_l)$ of polylist.

- polylist : a list of polynomials.
- $a_1, a_2, \ldots, a_l$: parameters.
- $v_1, v_2, \ldots, v_l$: values.

In the following example, we see how the command "sub4cgb" works.

```
A=cgb1([a*x^2*z^2+x*y^2+2*b,c*x^2+z^2+a,b*z+x],[a,b,c],[x,y,z],2)$
sub4cgb(A,[[a,-1],[b,0],[c,0]]);
[z^2-1,(-z^2+1)*y^2,x]

sub4cgb(A,[[a,2],[b,3],[c,2]]);
[-6*y^2+36*z^3-114*z,-18*z*y^2+57*z^6+114*z^4+36,-18*z*y^2-6*z^4-12*z^2+3
 6,6*y^4-216*z^6-144*z^2+228,19*z^2+2,-3*z*y^2+18*z^4+6,(-z^2-2)*y^2+6*z^
 5+12*z^3-36*z,x+3*z]

sub4cgb(A,[[a,0],[b,2/5],[c,2]]);
[-132/125*z,-8/25*z*y^2+66/125*z^6+16/25,-8/25*z*y^2+16/25,264/125,33/25*
 z^2,-2/5*z*y^2+4/5,-z^2*y^2-16/25*z,x+2/5*z]
```

10.1.3 Faithful comprehensive Gröbner systems

Here, we treat the commands for computing faithful comprehensive Gröbner systems in commutative polynomial rings. These commands are the main part of the commands for computing comprehensive Gröbner bases. The following four commands output a faithful comprehensive Gröbner system. Each of them has different optimization methods for getting nice and small comprehensive Gröbner systems. (These commands are included in the programs of the commands for computing computing comprehensive Gröbner bases (cgb, cgb1, cgb_re and cgb_re1).)

Commands:

fcgs(polylist,plist,vlist,option,*order*),
fcgs1(polylist,plist,vlist,option,*order*),
fcgs_re(polylist,plist,vlist,option,*order*),
fcgs_re1(polylist,plist,vlist,option,*order*).

Output:

a faithful comprehensive Gröbner basis for an ideal generated by polylist with respect to *order*.

All the commands output a faithful comprehensive Gröbner system. **What are differences?** The differences are the optimization techniques. These differences are the same as the commands of comprehensive Gröbner basis (see section 10.1.2).

The following table shows us what techniques all the commands have.

command	techniques
fcgs	**none**
fcgs1	**(2)**
fcgs_re	**(3)**
fcgs_re1	**(2), (3)**

In the following example, we see how the commands work.

```
fcgs1([a*x^2*y^2+x+y,a*x*y^2+y,b*x^2+b*y],[a,b],[x,y],1,0);
[a]==0,  [1]!=0,
[a*x+a*y^2+(a+1)*y,a*x^2+(a+1)*x+a*y]
[a+1]==0,  [1]!=0,
[a*x+a*y^2+(a+1)*y,(-y+1)*x+y,a*x^2+(a+1)*x+a*y]
[b]==0,  [a]!=0,
[a*x+a*y^2+(a+1)*y,(-y+1)*x+y,a*x^2+(a+1)*x+a*y]
[0]==0,  [b*a^2+b*a]!=0,
[(b*a+b)*y,(b*a+b)*x,b*x+b*y^2,a*x+a*y^2+(a+1)*y,(-y+1)*x+y,b*x^2+b*y,a*x
 ^2+(a+1)*x+a*y]
Number of segments is 4

fcgs([a*x^2*y^2+x+y,a*x*y^2+y,b*x^2+b*y],[a,b],[x,y],1,0);
[a]==0,  [1]!=0,
[a*x+a*y^2+(a+1)*y,a*x^2+(a+1)*x+a*y]
[a+1]==0,  [1]!=0,
[a*x+a*y^2+(a+1)*y,(-y+1)*x+y,a*x^2+(a+1)*x+a*y]
[b,a]==0,  [1]!=0,
[a*x+a*y^2+(a+1)*y,a*x^2+(a+1)*x+a*y]
[b]==0,  [a]!=0,
[a*x+a*y^2+(a+1)*y,(-y+1)*x+y,a*x^2+(a+1)*x+a*y]
[0]==0,  [b*a^2+b*a]!=0,
[(b*a+b)*y,(b*a+b)*x,b*x+b*y^2,a*x+a*y^2+(a+1)*y,(-y+1)*x+y,b*x^2+b*y,a*x
 ^2+(a+1)*x+a*y]
Number of segments is 5

fcgs_re([a*x^2*y^2+x+y,a*x*y^2+y,b*x^2+b*y],[a,b],[x,y],1,0);
[a]==0,  [1]!=0,
[a*x+a*y^2+(a+1)*y,a*x^2+(a+1)*x+a*y]
[a+1]==0,  [1]!=0,
[a*x+a*y^2+(a+1)*y,(-y+1)*x+y,a*x^2+(a+1)*x+a*y]
[b]==0,  [a]!=0,
[a*x+a*y^2+(a+1)*y,(-y+1)*x+y,a*x^2+(a+1)*x+a*y]
[0]==0,  [b*a^2+b*a]!=0,
[(b*a+b)*y,(b*a+b)*x,a*x+a*y^2+(a+1)*y,(-y+1)*x+y,a*x^2+(a+1)*x+a*y]
Number of segments is 4
```

10.2 CGBs and CGSs in rings of differential operators

In this section, we introduce the commands for computing comprehensive Gröbner bases and systems in rings of differential operators. The approaches and optimization techniques of all the commands, are the same as the previous section. We do not describe the details. We just show the commands, remarks and examples.

10.2.1 Comprehensive Gröbner systems

Here, we treat the commands for computing comprehensive Gröbner systems in $\mathbb{Q}[\bar{A}][\bar{X}, \bar{D}]$.

The package PGB has the following commands for computing comprehensive Gröbner systems in $\mathbb{Q}[\bar{A}][\bar{X}, \bar{D}]$. The following commands which have the Suzuki-Sato algorithm, output a comprehensive Gröbner system. With respect to speed, each of them has different optimization techniques for getting comprehensive Gröbner systems.

Commands:

cgsw(polylist,plist,[[$x_1,.,x_p$],[$d_1,.,d_p$]],option,*order*),
cgsw2(polylist,plist,[[$x_1,.,x_p$],[$d_1,.,d_p$]],option,*order*),
cgsw_re(polylist,plist,[[$x_1,.,x_p$],[$d_1,.,d_p$]],option,*order*).

Output:

a comprehensive Gröbner system for an ideal generated by polylist with respect to *order*.

- polylist : a list of differential operators (polynomials).
- plist : a list of parameters.
- [[$x_1,.,x_p$],[$d_1,.,d_p$]] : a list of variables such that $x_ix_j = x_jx_i$, $d_id_j = d_jd_i$, $x_id_j = d_jx_i$, for $i \neq j$ and $d_ix_i = x_id_i + 1$.
- *order :* a term order on the set of power product of $\mathrm{pp}(x_1, \ldots, x_p, d_1, \ldots, d_p)$ (see "**Setting term ordering**" of the manual [NST03]. Matrix orders are available).
- option : 1 or 0. This package PGB has two kinds of form for comprehensive Gröbner systems. One have to select **0** or **1**.

The following table shows us what techniques all the commands have.

command	techniques
cgsw	**(2)**
cgsw2	**none**
cgsw_re	**(2), (3)**

(see section 10.1.1)

Note that the technique **(1)** "computing the reduced Gröbner basis in each segment" has not been implemented, yet. It the author implement it, then the author will upload the program on the PGB website.

In the following examples, we see how the commands work, and we compare some commands.

```
cgsw([a*x1*d1^2*d2+(a+1)*x1*x2*d2,x2^2*d2+b*x1,d1*d2^2],[a,b],[[x1,x2],[d
 1,d2]],1,2);
[a+1]==0,  [b]!=0,
[x1,d2]
[b,a+1]==0,  [1]!=0,
[x2^2*d2,d1*d2]
[b]==0,  [a+1]!=0,
[x2*d2,d1*d2]
[0]==0,  [b*a+b]!=0,
[x1,d2]
Number of segments is 4

A=newmat(4,4,[[0,0,0,1],[0,0,1,0],[0,1,0,0],[1,0,0,0]]);
[ 0 0 0 1 ]
[ 0 0 1 0 ]
[ 0 1 0 0 ]
[ 1 0 0 0 ]

cgsw_re([a*x1*d1^2*d2+(a+1)*x1*x2*d2,x2^2*d2+b*x1^2],[a,b],[[x1,x2],[d1,d
 2]],0,A);
[[[a+1],[b],[x1^2,d2]],[[b],[a],[x2^2*d2,(a*x1*d1^2+(a+1)*x1*x2)*d2]],[[b
 ,a],[1],[x1*x2*d2,x2^2*d2]],[[a],[b],[x1,x2^2*d2]],[[0],[b*a^2+b*a],[x1,
 d2]]]
```

10.2.2 Comprehensive Gröbner bases

The package PGB has the following commands for computing comprehensive Gröbner bases in $\mathbb{Q}[\bar{A}][\bar{X},\bar{D}]$. The following commands output a comprehensive Gröbner basis. Each of them has different optimization techniques for getting nice and small comprehensive Gröbner bases.

Commands:

cgbw(polylist,plist,$[[x_1,.,x_p],[d_1,.,d_p]]$,$order$),
cgbw1(polylist,plist,$[[x_1,.,x_p],[d_1,.,d_p]]$,$order$),
cgbw_re(polylist,plist,$[[x_1,.,x_p],[d_1,.,d_p]]$,$order$).
cgbw_re1(polylist,plist,$[[x_1,.,x_p],[d_1,.,d_p]]$,$order$).

Output:

a comprehensive Gröbner basis for an ideal generated by polylist with respect to the *order*.

- polylist : a list of polynomials.
- plist : a list of parameters.
- $[[x_1,.,x_p],[d_1,.,d_p]]$: a list of variables such that $x_ix_j = x_jx_i$, $d_id_j = d_jd_i$, $x_id_j = d_jx_i$, for $i \neq j$ and $d_ix_i = x_id_i + 1$.

- *order :* a term order on the set of power product of vlist (see "**Setting term ordering**" of the manual [NST03]. Matrix orders are available).

The following table shows us what techniques all the commands have.

command	techniques
cgbw	**none**
cgbw1	**(2)**
cgbw_re	**(3)**
cgbw_re1	**(2)**, **(3)**

(see section 10.1.1)

In following examples, we see how the commands work.

```
cgbw([a*x1*d1^2*d2+(a+1)*x1*x2*d2,x2^2*d2+b*x1^2,d1*d2^2],[a,b],[[x1,x2],
 [d1,d2]],2);
[x2^2*d2+b*x1^2,x2^2*d2^2+2*x2*d2,(a+1)*x2*d2,d1*d2,b*d2]

A=cgbw_re([a*x1*d1^2*d2+(a+1)*x1*x2*d2,x2^2*d2+b*x1^2,d1*d2^2],[a,b],[[x1
 ,x2],[d1,d2]],2);
[b*d2,d1*d2,(a+1)*x2*d2,x2^2*d2^2+2*x2*d2,x2^2*d2+b*x1^2]

sub4cgb(A,[[a,1],[b,2]]);
[2*d2,d1*d2,2*x2*d2,x2^2*d2^2+2*x2*d2,x2^2*d2+2*x1^2]

cgbw([a*x1*d1^2*d2+(c+1)*x1*x2*d2,x2^2*d2+b*x1,c*d1*d2^2],[a,b,c],[[x1,x2
 ],[d1,d2]],2);
[(x1*x2^2+2*b*a)*d2^2+2*x1*x2*d2,(-a*x1*d1^3-a*d1^2-x1*x2*d1+(-c-1)*x2)*d
 2,(a*x1*d1^2+(c+1)*x1*x2)*d2,(a^2*x1*d1^4+2*a^2*d1^3+2*a*x2*d1+2*b*a)*d2
 ,x2^2*d2+b*x1,x2^3*d2^2+(2*b*a*d1+2*x2^2)*d2,a*x2^2*d1*d2,-a*x2^2*d2^2-2
 *a*x2*d2,c*x2^2*d2^2+2*c*x2*d2,(c+1)*a*x2^2*d2,-b*a*x2*d2,(-c^2-c)*x2*d2
 ,c*d1*d2,b*a^2*d2,b^2*a*d2,c*b*d2]
```

10.2.3 Faithful comprehensive Gröbner systems

Here, we introduce the commands for computing faithful comprehensive Gröbner systems in $\mathbb{Q}[\bar{A}][\bar{X},\bar{D}]$. These commands are the main part of the commands for computing comprehensive Gröbner bases.

The following four commands output a faithful comprehensive Gröbner system. Each of them has different optimization methods for getting nice and small comprehensive Gröbner systems. Actually, these commands are included in the commands of comprehensive Gröbner bases (cgb, cgb1, cgb_re and cgb_re1).

Commands:

fcgsw(polylist,plist,vlist,option,*order*),
fcgsw1(polylist,plist,vlist,option,*order*),
fcgsw_re(polylist,plist,vlist,option,*order*),
fcgsw_re1(polylist,plist,vlist,option,*order*).

Output:

a faithful comprehensive Gröbner system for an ideal generated by polylist with respect to *order*.

The following table shows us what techniques all the commands have.

command	techniques
fcgsw	**none**
fcgsw1	**(2)**
fcgsw_re	**(3)**
fcgsw_re1	**(2), (3)**

(see section 10.1.1)

In following examples, we see how the commands work.

```
fcgsw1([a*x1*d1^2*d2+(a+1)*x1*x2*d2,x2^2*d2+b*x1^2,d1*d2^2],[a,b],[[x1,x2
 ],[d1,d2]],1,2);
[a+1]==0,  [b]!=0,
[x2^2*d2+b*x1^2,x2^2*d2^2+2*x2*d2,d1*d2,b*d2]
[a+1,b]==0,  [1]!=0,
[x2^2*d2+b*x1^2,d1*d2]
[b]==0,  [a+1]!=0,
[x2^2*d2+b*x1^2,(a+1)*x2*d2,d1*d2]
[0]==0,  [b*a+b]!=0,
[x2^2*d2+b*x1^2,x2^2*d2^2+2*x2*d2,(a+1)*x2*d2,d1*d2,b*d2]
Number of segments is 4

fcgsw([a*x1*d1^2*d2,b*x1*x2*d2,a*x2^2*d2+x1^2],[a,b],[[x1,x2],[d1,d2]],0,
 1);
[[[b],[a],[a*d2,x1^2]],[[b,a],[1],[x1^2]],[[a],[b],[x1^2,b*d2*x2*x1]],[[0
 ],[b*a],[a*d2,x1^2,b*x1*x2*d2]]]
```

10.2.4 CGBs and CGSs in $(\mathbb{Q}[\bar{A}][\bar{X}])[\bar{D}]$

In chapter 8, we described a method for computing comprehensive Gröbner systems and comprehensive Gröbner bases in $(K[\bar{A}][\bar{X}])[\bar{D}]$. If one wants to compute comprehensive Gröbner systems and comprehensive Gröbner bases in $(K[\bar{A}][\bar{X}])[\bar{D}]$, then one need to input a block order on the position "*order*" of the commands for computing comprehensive Gröbner systems and comprehensive Gröbner bases in $K[\bar{A}][\bar{X},\bar{D}]$.

10.3 CGBs and CGSs for modules

In this section, we introduce the commands for computing comprehensive Gröbner Gröbner bases and comprehensive Gröbner systems in $\mathbb{Q}[\bar{A}][\bar{X}]^r$. The approaches and optimization techniques of all the commands are the same as section 10.1. We do not describe the details. We just give the commands, remarks and examples.

10.3.1 Comprehensive Gröbner systems

Here, we treat the commands for computing comprehensive Gröbner systems in $\mathbb{Q}[\bar{A}][\bar{X}]^r$ where $r \in N$.

The package PGB has the following commands for computing comprehensive Gröbner systems in $\mathbb{Q}[\bar{A}][\bar{X}]^r$. Concerning speed and output, each of them has different optimization techniques for getting comprehensive Gröbner systems.

Commands:

cgs_m(veclist,plist,vlist,option, t or p, *order*),
cgs_m1(veclist,plist,vlist,option, t or p, *order*).

Output:

a comprehensive Gröbner system for a submodule generated by veclist with respect to a module order ([TOP or PTO] and *order*).

- veclist : a list of vectors such that all vectors have the **same length**.
- plist : a list of parameters.
- vlist : a list of variables.
- t or p: t means TOP and p means POT. One have to select one of both.
- *order :* a term order on the set of power product of vlist.

The following table shows us what techniques all commands have.

command	techniques
cgs_m	**none**
cgs_m1	**(2)**

Note that the techniques **(1)** and **(2)** have not been implemented yet.

There exists the following command in the package for substituting arbitrary values for parameters of a comprehensive Gröbner system in $\mathbb{Q}[\bar{A}][\bar{X}]^r$.

Commands:

sub4cgsmodule(a comprehensive Gröbner system,[[a_1,v_1],[a_2,v_2],...,[a_l,v_l]])

Procedure:
(1) The program searches some segments $(\mathcal{A}_1, G_1), \ldots, (\mathcal{A}_l, G_l)$ of a comprehensive Gröbner system such that $(v_1, \ldots, v_l) \in \mathcal{A}_i$, $1 \leq i \leq l$.
(2) The program substitutes the values $v_1, \ldots, v_l$ for the parameters $a_1, \ldots, a_l$ of the set polynomials $G_1, \ldots, G_l$.
(3) The program **outputs** the parametric spaces, the original set of polynomials $G_1, \ldots, G_l$ and the set of polynomials $G_1(v_1, \ldots, v_l), \ldots, G_l(v_1, \ldots, v_l)$ computed.

- $a_1, a_2, \ldots, a_l$: parameters.
- $v_1, v_2, \ldots, v_l$: values.

In following examples, we see how the commands work.

```
cgs_m([[a*x^2*y+b*x,x*y],[b*x*y^3+a*y,(b+1)*y*x+1]],[a,b],[x,y],1,t,2);
[a]==0,  (b)!=0,
[b*x,y*x]
[(b*y^3-b^2-b)*x,1]

[b+1,a]==0,  (1)!=0,
[-x,y*x]
[-y^3*x,1]

[b+1]==0,  (a)!=0,
[(y^2-a^2*y)*x,(-y^3-a)*x]
[-y^3*x+a*y,1]
[a*y*x^2-x,y*x]

[b,a]==0,  (1)!=0,
[0,1]

[b]==0,  (a)!=0,
[a*y,y*x+1]
[a*y*x^2-a*y,-1]

[0]==0,  (b)*(b+1)*(a)!=0,
[b*y^3*x+a*y,(b+1)*y*x+1]
[(b^2*y^2-a^2*y)*x,(-b-1)*a*y*x^2+(b*y^3-a)*x]
[a*y*x^2+b*x,y*x]
6 segments

A=cgs_m1([[a*x^2*y+b*x,x*y+x],[x+a*y,0]],[a,b],[x,y],0,p,2);
[[[0],[a],[[0,(-y-1)*x^2+(-a*y^2-a*y)*x],[a^3*y^3-b*a*y,(y+1)*x],[x+a*y,0
]]],[[a],[1],[[0,(y+1)*x],[x,0]]]]

sub4cgsmodule(A,[[a,3],[b,0]]);
[0]==0, (a)!=0,
[[0,(-y-1)*x^2+(-a*y^2-a*y)*x],[a^3*y^3-b*a*y,(y+1)*x],[x+a*y,0]]
[[0,(-y-1)*x^2+(-3*y^2-3*y)*x],[27*y^3,(y+1)*x],[x+3*y,0]]
```

10.3.2 Comprehensive Gröbner bases

The package PGB has the following commands for computing comprehensive Gröbner bases in $\mathbb{Q}[\bar{A}][\bar{X}]^r$. The following commands output a comprehensive Gröbner basis. Each of them has different optimization techniques for getting nice and small comprehensive Gröbner bases.

Commands:

cgb_m(veclist,plist,vlist, t or p, *order*),
cgb_m1(veclist,plist,vlist, t or p, *order*).

Output:

a comprehensive Gröbner basis for an ideal generated by veclist with respect to a module order ([TOP or PTO] and *order*).

- veclist : a list of vectors such that all vectors have the same length.
- plist : a list of parameters.
- vlist : a list of variables.
- t or p: t means TOP and p means POT. One have to select one of both.
- *order :* a term order on the set of power product of vlist.
- option : 1 or 0. This package PGB has two kinds of form for comprehensive Gröbner systems. One have to select **0** or **1**.

The following table shows us which kinds of techniques all the commands have.

command	techniques
cgs_m	**none**
cgs_m1	**(2)**

(see section 10.1.1)

There exists the following command in the package for substituting any value for parameters of a comprehensive Gröbner system in $\mathbb{Q}[\bar{A}][\bar{X}]^r$.

Commands:

sub4cgsbmodule(veclist,[[a_1,v_1],[a_2,v_2],...,[a_l,v_l]])

Output:
a set of vectors which is substituted $(v_1, \ldots, v_l)$ for parameters $(a_1, \ldots, a_l)$ of veclist.

- veclist: a list of vectors,

- $a_1, a_2, \ldots, a_l$: parameters.
- $v_1, v_2, \ldots, v_l$: values.

Commands:

ret_list(veclist)

The command arranges each vector of veclist up lengthways.

In following examples, we see how the commands work.

```
B=cgb_m([[a*x*y^2+b,x*y+y^2],[x*y^3+a*y,0]],[a,b],[x,y],1,p,2);
[[-b*a^2+b^2,-a*y^3*x^2+(-a*y^4+(-a^2+b)*y)*x+(-a^2+b)*y^2],[b*y^2*x+b*a,
 y^3*x^2+(y^4+a*y)*x+a*y^2],[(a^2-b)*y,-y^2*x-y^3],[-a^2*y^2*x-b*a,-a*y*x
 -a*y^2],[0,-y^4*x^2+(-y^5-a*y^2)*x-a*y^3],[a*y^2*x+b,y*x+y^2],[y^3*x+a*y
 ,0]]

C=sub4cgbmodule(B,[[a,0],[b,-1/2]]);
[[y^3*x,0],[-1/2,y*x+y^2],[0,-y^4*x^2-y^5*x],[1/2*y,-y^2*x-y^3],[-1/2*y^2
 *x,y^3*x^2+y^4*x],[1/4,-1/2*y*x-1/2*y^2]]

ret_list(B);
[-b*a^2+b^2,-a*y^3*x^2+(-a*y^4+(-a^2+b)*y)*x+(-a^2+b)*y^2]
[b*y^2*x+b*a,y^3*x^2+(y^4+a*y)*x+a*y^2]
[(a^2-b)*y,-y^2*x-y^3]
[-a^2*y^2*x-b*a,-a*y*x-a*y^2]
[0,-y^4*x^2+(-y^5-a*y^2)*x-a*y^3]
[a*y^2*x+b,y*x+y^2]
[y^3*x+a*y,0]

ret_list(C);
[y^3*x,0]
[-1/2,y*x+y^2]
[0,-y^4*x^2-y^5*x]
[1/2*y,-y^2*x-y^3]
[-1/2*y^2*x,y^3*x^2+y^4*x]
[1/4,-1/2*y*x-1/2*y^2]
```

10.3.3 Faithful comprehensive Gröbner systems

Here, we describe the commands for computing faithful comprehensive Gröbner systems in $\mathbb{Q}[\bar{A}][\bar{X}]^r$. These commands are the main part of the commands for computing comprehensive Gröbner bases.

The package PGB has the following commands.

Commands:

fcgs_m(veclist,plist,vlist,option, t or p, *order*),
fcgs_m1(veclist,plist,vlist,option, t or p, *order*).

Output:

a faithful comprehensive Gröbner basis for an ideal generated by veclist with respect to a module order ([TOP or PTO] and *order*).

The following table shows us what techniques all the commands have.

command	techniques
fcgs_m	**none**
fcgs_m1	**(2)**

(see section 10.1.1)

In the following examples, we see how the commands work.

```
fcgs_m1([[x*y^2+b,a*y+y^2],[x*y^2+2,a*x]],[a,b],[x,y],1,p,2);
[0]==0,  (a)*(b-2)!=0,
[0,-a*y^2*x^2+(y^4+a*y^3-b*a)*x+2*y^2+2*a*y]
[-b+2,a*x-y^2-a*y]
[-y^2*x-2,-a*x]

[a]==0,  (b-2)!=0,
[0,-a*y^2*x^2+(y^4+a*y^3-b*a)*x+2*y^2+2*a*y]
[-b+2,a*x-y^2-a*y]
[-y^2*x-2,-a*x]

[b-2,a]==0,  (1)!=0,
[b-2,-a*x+y^2+a*y]
[-y^2*x-2,-a*x]

[b-2]==0,  (a)!=0,
[b-2,-a*x+y^2+a*y]
[y^2*x+b,y^2+a*y]
4 segments
```

10.4 Related objects

In this section, we introduce very useful commands which are not included in the computer algebra system Risa/Asir.

10.4.1 Gröbner bases

First we introduce a command for computing Gröbner bases in $\mathbb{Q}[\bar{X}]^r$ which is the following.

Commands:

groebner_module(veclist,plist,vlist, t or p, *order*).

Output:
the reduced Gröbner basis for an submodule generated by veclist.

- veclist : a list of vectors such that all vectors have **same length**.
- plist : a list of parameters.
- vlist : a list of variables.
- *order :* a term order on the set of power product of vlist.
- t or p: t means TOP, and p means POT. One have to select one of both.

Second, we introduce a command for computing a Gröbner basis in $\mathbb{Q}[\bar{A}][\bar{X}]$ (the coefficient domain is the polynomial ring). In chapter 3, we introduced the algorithms for computing Gröbner bases in polynomial rings over a polynomial ring. The following command computes a Gröbner basis in $\mathbb{Q}[\bar{A}][\bar{X}]$.

Commands:

regb_pp(polylist,cvlist,vlist, *order*).

Output:
a (weak) reduced Gröbner bases in $\mathbb{Q}$[cvlist][vlist] where $\bar{A}$:=cvlist and $\bar{X}$:=vlist.

- cvlist : a list of variable.
- vlist : a list of variables such that cvlist $\cap$ vlist$\neq \emptyset$.
- *order :* a term order on the set of power product of vlist.

In the following examples, we see how the commands work.

```
A=groebner_module([[x*y+x^2,y+1],[y^2+x,0]],[x,y],t,2);
[[y^4-y^3,y+1],[x+y^2,0],[0,(-y-1)*x-y^3-y^2]]

ret_list(A);
[y^4-y^3,y+1]
[x+y^2,0]
[0,(-y-1)*x-y^3-y^2]

regb_pp([a*x^2*z+a*y+a,a*x*z+b,(a+1)*x*z+a*b],[a,b],[x,y,z],1);
[-b*a^2+b*a+b,(a^3-a^2-a)*y+a^3-a^2-a,-b*x+a*y+a,a*z*y+a*z+b^2*a-b^2,-z*x
 -b*a+b]
```

10.4.2 Syzygies

The computer algebra system Risa/Asir does not have a command for computing a basis of syzygy module. However, the package PGB has the command "syzygy" for computing a basis of syzygy module.

Commands:

syzygy([[p_1],...,[p_l]],[vlist],*order*).

Output: a basis of syzygy module between $p_1,\ldots, p_l$.

- $p_1,\ldots,p_l$: polynomials in $\mathbb{Q}$[vlist].
- vlist : a list of variables.
- *order :* a term order on the set of power product of vlist.

In following examples, we see how the command works.

```
syzygy([[x^2*y+x],[3*y^2+y+2],[x^2+x*y+3]],[x,y],2);
[[x^2+y*x+3,0,-y*x^2-x],[3*y^2+y+2,(y^2-1)*x+3*y,-3*y^3-y^2-2*y],[0,-x^2-
 y*x-3,3*y^2+y+2]]

syzygy([[a^2+b+c],[a*c+1],[b^2+a*b+3*c]],[a,b,c],0);
[[c*b^2-b+3*c^2,-3*c*a+b^3+b^2+4*c*b,(-c*b+1)*a+(-c-1)*b-c^2],[b*a+b^2+3*
 c,0,-a^2-b-c],[-c*a-1,a^2+b+c,0],[0,b*a+b^2+3*c,-c*a-1]]
```

In chapter 9, we saw the algorithm for computing comprehensive syzygy systems. The following command is for computing comprehensive syzygy systems, i.e., the command outputs a basis of parametric syzygy for parametric polynomials.

Commands:

p_syzygy([[p_1],...,[p_l]],[plist],[vlist],Option,*order*).

Output:
a comprehensive syzygy system between $p_1,\ldots,p_l$.

- $p_1,\ldots,p_l$: polynomials in $\mathbb{Q}$[plist][vlist].
- plist : a list of parameters.
- vlist : a list of variables.
- *order :* a term order on the set of power product of vlist.
- option : 1 or 0. This package PGB has two kinds of form for comprehensive Gröbner systems. One have to select **0** or **1**.

In following examples, we see how the command works.

```
p_syzygy([[a*y^2+x+1],[b*x+b],[x*y+a]],[a,b],[x,y],1,2);
[0]==0,  (b)*(a)!=0,
[b*x+b,-x-a*y^2-1,0]
[b*y-b*a,-a*y^3-y+a,b*a*y^2]
[0,y*x+a,-b*x-b]

[b]==0,  (a)!=0,
```

```
[-y*x-a,0,x+a*y^2+1]
[0,1,0]

[a,b]==0,  (1)!=0,
[-y*x,0,x+1]
[0,1,0]

[a]==0,  (b)!=0,
[-b,1,0]
[0,y*x,-b*x-b]
```

This output means the following.

$$\begin{cases} \{[bx+b,-x-ay^2-1,0],[by-ba,-ay^3-y+a,bay^2], & \text{if } ab \neq 0, \\ \quad [0,yx+a,-bx-b]\}, & \\ \{[-yz-a,0,x+ay^2+1],[0,1,0]\}, & \text{if } b=0, a\neq 0, \\ \{[-yx,0,x+1],[0,1,0]\} & \text{if } a=b=0, \\ \{[-b,1,0],[0,yx,-bx-b]\} & \text{if } a=0, b\neq 0. \end{cases}$$

10.5 Concluding remarks

In chapter 5, we saw the algorithm NEW for computing comprehensive Gröbner systems. We can easily extend the algorithm to rings of differential operators and modules. The author has not implemented it in their domains yet. If the algorithm is implemented in the domains, the author will upload the programs on the website `http://www.risc.uni-linz.ac.at/people/knabeshi/pgb/` (or author's website).
Since the author likes logic programing language, the author used a lot of the "list" structure of Risa/Asir in the package. With respect to speed, this is not good. Therefore, the author will change the data structure to "*module*" which is the special data structure of Risa/Asir. All new programs will be uploaded in the PGB website.

Bibliography

[Bec94] Becker, T. On Gröbner bases under specialization. *Applicable Algebra in Engineering, Communication and Computing*, 5:1–8, 1994.

[BS80] Burris, S. and Sankappanavar, H.P. *A Course in Universal Algebra*. Springer-Verlag, 1980.

[Buc65] Buchberger, B. *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenrings nach einem nulldimensionalen Polynomideal*. Universität Innsbruck, Austria, 1965. Ph.D. Thesis.

[Buc70] Buchberger, B. Ein algorithmisches Kriterium für die Lösbarkeit eines algebraischen Gleichungssystems. *Aequationes Math*, 4:374–383, 1970.

[Buc79] Buchberger, B. A criterion for detecting unnecessary reductions in the construction of Gröbner bases. In Ng, E.W., editor, *EUROSAM'79*, pages 3–21. Springer, 1979.

[Buc85] Buchberger, B. Gröbner bases: An Algorithmic Method in Polynomial Ideal Theory. In Bose, N., editor, *Multidimensional Systems Theory*, pages 184–232. Reidel Publishing Company, 1985.

[Buc06] Buchberger, B. Bruno Buchberger's PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal. *Journal of Symbolic Computation*, 41(3-4):475–511, 2006. English translation.

[BW93] Becker, T. and Weispfenning, V. *Gröbner Bases, a computational Approach to Commutative Algebra*. Springer New York, 1993.

[BW98] Buchberger, B. and Winkler, F. Ed. *Gröbner Bases and Applications*. Cambridge University Press, 1998.

[Cas86] Castro, F. Calcul de la dimension et des multiplicités d'un D-module monogĕne. *C.R. Acad. Sci. Paris*, t. 302, Série I:184–232, 1986.

[Cas87] Castro, F. Calculs effectifs pour les ideaux d'opérateurs differentiels. In J.M. Aroca, Sánchez Giralda, and J.L. Vicente, editors, *Géométrie algébrique et applications*, pages 1–20. Hermann, Paris, 1987.

[CLO92] Cox, D., Little, J., and O'Shea, D. *Ideals, Varieties and Algorithms*. Springer, 1992.

[CLO97] Cox, D., Little, J., and O'Shea, D. *Using Algebraic Geometry*. Springer, 1997.

[DS97] Dolzmann, A. and Sturm, T. Redlog: Computer algebra meets computer logic. *ACM SIGSAM Bulletin*, 31(2):2–9, 1997.

[DSS06] Dolzmann, A., Strurm, T., and Seidl, A. *Redlog User Mabual Version 3.06*. Universität Passau, 2006.

[EGT01] Elisabetta, F., Gianni, P., and Trager, B.M. Degree reduction under specialization. *Journal of Pure and Applied Algebra*, 164:153–163, 2001.

[FSK86] Furukawa, A., Sasaki, T., and Kobayashi, H. Gröbner Basis of a Module over $\mathrm{K}[x_1, \ldots, x_n]$ and Polynomial Solutions of a System of Linear Equations. In *SYMSAC*, pages 222–224. ACM Press, 1986.

[Gal85] Galligo, A. Some algorithmic questions on ideals of differential operators. In Caviness, B.F., editor, *European Conference on Computer Algebra (EURO-*

CAL 85), VoL. II, LNCS 204, pages 413–421. Springer, 1985.

[Gia87] Gianni, P. Properties of Gröbner bases under specializations. In Davenport, J., editor, *EUROCAL'87*, pages 293–297. ACM Press, 1987.

[GMP02] Greuel, G-M. and Pfister, G. *A Singular Introduction to Commutative Algebra*. Springer, 2002.

[GMPS05] Greuel, G-M., Pfister, G., and Schönemann, H. *SINGULAR, Version 3-0-1 October 2005*. Technische Universität Kaiserslautern, 2005.

[IP98] Insa, M. and Pauer, F. Gröbner Bases in Rings of Differential Operators. In Buchberger, B. and Winkler, F., editors, *Gröbner Bases and Applications*, pages 367–380. Cambridge University Press, 1998.

[Kal97] Kalkbrener, M. On the Stability of Gröbner Bases Under Specializations. *Journal of Symbolic Computation*, 24:51–58, 1997.

[KR00] Kreuzer, M. and Robbiano, L. *Computational Commutative Algebra 1*. Springer, 2000.

[Kre92] Kredel, H. *Solvable Polynomial Rings*. Universität Passau, Germany, 1992. Ph.D. Thesis.

[KRK88] Kandri-Rody, A. and Kapur, D. Computing a Gröbner basis of a polynomial ideal over a euclidean domain. *Journal of Symbolic Computation*, 6:37–57, 1988.

[KRW90] Kandri-Rody, A. and Weispfenning, V. Non-commutative Gröbner bases in algebras of of solvable type. *Journal of Symbolic Computation*, pages 1–26, 1990.

[KW91] Kredel, H. and Weispfenning, V. Parametric Gröbner bases for noncommutative Polynomials. In *IV. Int. Cov. Computer Algebra in Physical Research 1990*, pages 236–244. World Scientific, 1991.

[Li,02] Li, H. *Noncommutative Gröbner Bases and Filtered-Graded Transfer*. Springer, 2002.

[Lou79] Loullis, G. Sheaves and boolean valued model theory. *Journal of Symbolic Logic*, 44 (2):153–183, 1979.

[MM86] Möller, M. and Mora, F. New Constructive Methods in Classical Ideal Theory. *Journal of Algebra*, 100:138–178, 1986.

[MM05] Manubens, M. and Montes, A. Improving DISPGB algorithm using the discriminant ideal (extended abstract). In Dolzmann, A., Seidl, A., and Thomas, S., editors, *the A3L 2005, conference in Honor of the 60th Birthday of Volker Weispfenning*, pages 159–166. BOD Norderstedt, 2005.

[MM06] Manubens, M. and Montes, A. Improving DISPGB algorithm using the discriminant ideal. *Journal of Symbolic Computation*, 41:1245–1263, 2006.

[Mon02] Montes, A. A new algorithm for discussing Gröbner basis with parameters. *Journal of Symbolic Computation*, 33/1-2:183–208, 2002.

[Mor86] Mora, T. Gröbner basis for noncommutative polynomial rings. In Calmet, J., editor, *Algebraic Algorithms and Error-Correcting Codes (AAECC 3), LNCS 229*, pages 353–362. Springer, 1986.

[Mor88] Mora, T. Gröbner basis in non-commutative algebras. In Gianni, P., editor, *International Symposium on Symbolic and Algebraic Computation (ISSAC 88), LNCS 358*, pages 150–161. Springer, 1988.

[Mor94] Mora, T. An introduction to commutative and noncommutative Gröbner bases. *Theoretical Computer Science*, 134:131–173, 1994.

[Nab05a] Nabeshima, K. A computation method for ACGB-V. In Dolzmann, A., Seidl, A., and Thomas, S., editors, *the A3L 2005, conference in Honor of the 60th Birthday of Volker Weispfenning*, pages 171–180. BOD Norderstedt, 2005.

[Nab05b] Nabeshima, K. A Direct Products of Fields Approach to Comprehensive

Gröbner Bases over Finite Fields. In Zaharie, D., Petcu, D., Negru, V., Jebelean, T., Ciobanu, G., Cicortas, A., Abraham, A., and Paprzycki, M., editors, *International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, pages 39–47. IEEE Computer Society, 2005.

[Nab06] Nabeshima, K. Reduced Gröbner bases in polynomial rings over a polynomial ring. In Wang, D. and Zheng, Z., editors, *International Conference on Mathematical Aspects of Computer and Information Sciences*, pages 15–32, 2006.

[Nab07a] Nabeshima, K. Comprehensive Gröbner Bases for Modules. 2007. preprint.

[Nab07b] Nabeshima, K. PGB: A Package for Computing Parametric Gröbner Bases and Related Objects. 2007. preprint `http://www.risc.uni-linz.ac.at/people/knabeshi/pgb/`.

[Nab07c] Nabeshima, K. Reduced Gröbner bases in polynomial rings over a polynomial ring. 2007. submitted for publication.

[Nab07d] Nabeshima, K. A Speed-Up of the Algorithm for Computing Comprehensive Gröbner Systems. In Brown, C., editor, *International Symposium on Symbolic and Algebraic Computation*. ACM-Press, 2007. To appear.

[Nab07e] Nabeshima, K. Tow kinds of Gröbner bases in rings of Differential Operators and Their Comprehensive Gröbner bases. 2007. submitted for publication.

[NG94] Nakos, G. and Glinos, N. Computing Gröbner Bases over the integers. *The Mathematica Journal*, 4-3:70–75, 1994.

[NST03] Noro, M., Shimoyama, T., and Takeshima, T. Risa/Asir committers, 2003. `http://www.math.kobe-u.ac.jp/Asir/asir.html`.

[NT92] Noro, M. and Takeshima, T. Risa/Asir- A Computer Algebra System. In Wang, P., editor, *International Symposium on Symbolic and Algebraic Computation*, pages 387–396. ACM-Press, 1992.

[Oak02] Oaku, T. *D-module and Computer Mathematics*. Asakura Syoten, 2002. in Japanese.

[OS94] Oaku, T. and Shimoyama, T. A Gröbner Basis Method for Modules over Rings of Differential Operators. *Journal of Symbolic Computation*, 18:223–248, 1994.

[PW06] Pan, W. and Wang, D. Uniform Gröbner Bases for Ideals Generated by Polynomials with Parametric Exponebts. In Dumas, J-G., editor, *International Symposium on Symbolic and Algebraic Computation*, pages 269–276. ACM Press, 2006.

[Rob85] Robbiano, L. Term orderings on the polynomial ring. In Caviness, B.F., editor, *European Conference on Computer Algebra (EUROCAL 85), VoL. II, LNCS 204*, pages 513–517. Springer, 1985.

[Sat98] Sato, Y. A new type of canonical Gröbner bases in polynomial rings over von Neumann regular rings. In Gloor, O., editor, *International Symposium on Symbolic and Algebraic Computation*, pages 317–321. ACM-Press, 1998.

[Sat05] Sato, Y. Stability of Gröbner basis and ACGB. In Dolzmann, A., Seidl, A., and Thomas, S., editors, *the A3L 2005, conference in Honor of the 60th Birthday of Volker Weispfenning*, pages 223–228. BOD Norderstedt, 2005.

[SS02] Suzuki, A. and Sato, Y. An alternative approach to Comprehensive Gröbner bases. In Mora, T., editor, *International Symposium on Symbolic and Algebraic Computation*, pages 255–261. ACM Press, 2002.

[SS03] Suzuki, A. and Sato, Y. An alternative approach to Comprehensive Gröbner bases. *Journal of Symbolic Computation*, 36/3-4:649–667, 2003.

[SS04] Suzuki, A. and Sato, Y. Comprehensive Gröbner Bases via ACGB. In Tran, Q-N., editor, *The 10th Internatinal Conference on Applications of Computer Algebra*, pages 65–73. Lamar University, 2004.

[SS06] Suzuki, A. and Sato, Y. A Simple Algorithm to compute Comprehensive Gröbner Bases using Gröbner bases. In *International Symposium on Symbolic and Algebraic Computation*, pages 326–331. ACM Press, 2006.

[SSN02] Sato, Y., Suzuki, A., and Nabeshima, K. Generalized discrete Comprehensive Gröbner Bases. In *CA-ALIAS*, pages 105–110. RIMS, Kyoto University, 2002.

[SSN03a] Sato, Y., Suzuki, A., and Nabeshima, K. ACGB on Varieties. In Ganzha, V.F., Mayr, E.W., and Vorozhtsov, E.V., editors, *The 6th International Workshop on Computer Algebra in Scientic Computing (CASC)*, pages 313–318. Technische Universität München, 2003.

[SSN03b] Sato, Y., Suzuki, A., and Nabeshima, K. Discrete Comprehensive Gröbner bases II. In Li, Z. and Sit, W., editors, *Computer Mathematics, Lecture Notes Series on Computing*, pages 240–247. World Scientific, 2003.

[SST99] Saito, M., Sturmfels, B., and Takayama, N. *Gröbner Deformations of Hypergeometric Differential Equations*. Springer, 1999.

[SW75] Saracino, D. and Weispfenning, V. On algebraic curves over commutative regular rings. In Saracino, D. and Weispfenning, V., editors, *Model Theory and Algebra, a memorial tribute to A. Robinson*, pages 307–387. Springer, 1975.

[Tak89] Takayama, N. Gröbner bases and the problem of contiguous relations. *Japan Journal of Applied Mathematics*, 6:147–160, 1989.

[Wan05] Wang, D. The projection property of regular systems and its application to solving parametric polynomial systems. In Dolzmann, A., Seidl, A., and Thomas, S., editors, *the A3L 2005, conference in Honor of the 60th Birthday of Volker Weispfenning*, pages 269–274. BOD Norderstedt, 2005.

[Wei87] Weispfenning, V. Gröbner bases for polynomial ideals over commutative regular rings. In Davenport, J., editor, *EUROCAL '87, LNCS378*, pages 336–347. Springer, 1987.

[Wei92] Weispfenning, V. Comprehensive Gröbner bases. *Journal of Symbolic Computation*, 14/1:1–29, 1992.

[Wei02a] Weispfenning, V. Canonical Comprehensive Gröbner bases. In Mora, T., editor, *International Symposium on Symbolic and Algebraic Computation*, pages 270–278. ACM Press, 2002.

[Wei02b] Weispfenning, V. Comprehensive Gröbner bases and regular rings. In Nakagawa, K., editor, *Symposium in Honor of Bruno Buchberger's 60th Birthday*, pages 256–264. RISC-Linz, 2002.

[Wei03] Weispfenning, V. Canonical Comprehensive Gröbner bases. *Journal of Symbolic Computation*, 36/3-4:669–683, 2003.

[Wei04] Weispfenning, V. Gröbner bases for binomials with parametric exponents. In Ganzha, V.F., Mayr, E.W., and Vorozhtsov, E.V., editors, *International Workshop on Computer Algebra in Scientic Computing (CASC)*, pages 467–478. Technische Universität München, 2004.

[Wei06] Weispfenning, V. Comprehensive Gröbner bases and regular rings. *Journal of Symbolic Computation*, 41(3-4):285–296, 2006.

[Win86] Winkler, F. Solution of Equations I: Polynomial Ideals and Gröbner bases. In Jenks, R.D., Chudnovsky, D., and Dekker, M., editors, *Computers Mathematics (Lecture Notes in Pure and Applied Mathematics, Vol. 125)*, pages 383–407, 1986.

[Win96] Winkler, F. *Polynomial Algorithms in Computer Algebra*. Springer-Verlag Wien New York, 1996.

[Yok04] Yokoyama, K. On Systems of Algebraic Equations with Parametric Exponents. In Gutierrez, J., editor, *International Symposium on Symbolic and Algebraic*

Computation, pages 312–317. ACM Press, 2004.

[Yok07] Yokoyama, K. On Systems of Algebraic Equations with Parametric Exponents II. 2007. submitted for publication.

[ZW06] Zhou, M. and Winkler, F. On computing Gröbner bases in rings of differential operators with coefficients in a ring. In Wang, D. and Zheng, Z., editors, *International Conference on Mathematical Aspects of Computer and Information Sciences*, pages 45–56, 2006.

Index

$\bar{A}$, 17
$\bar{X}$, 6, 17
BC, 70
bc, 70
br, 70
$\mathbb{C}$, 5
$\mathbb{N}$, 5
$\mathbb{Q}$, 5
$\mathbb{R}$, 5
rad, 6
SP, 71
Spec, 68
St, 68
$\xrightarrow{r1}$
  rings of diff. ope., 96
  commutative polynomial ring, 9
  module, 143
$\xrightarrow{r2}$
  commutative polynomial ring, 19
  module, 144
  rings of diff. ope. , 98
ter_T, 77
$\mathbb{V}$, 6
$\mathbb{Z}$, 5
a^*, 65
$K[\bar{X}]$, 6

ACGB, 65, 76
ACGB-V, 80
admissible order, 6
affine variety, 6
algebraically constructible set, 35
Algorithm
  CGBM, 136
  CSS, 138
  $\bar{D}$-GröbnerBasis, 101
  ACGB, 78
  BC , 70
  Buchberger, 10
  CGB, 45
  CGBMain, 44
  CGBMainM, 135
  CGBMainW, 115
  CGBooverF, 88
  CGBW, 115
  CGSM, 127
  CGSMain, 41
  CGSMainM, 128
  CGSMainW, 108
  CGSW, 107
  CSB, 141
  DCGB, 82
  Div-zero, 84
  EGA, 15
  factorize, 57
  FCGS, 44
  FCGSM, 134
  GBovN , 71
  GröbnerBasisB, 22
  GröbnerBasisM, 125
  GröbnerBasisW, 96
  ImproveGB , 75
  InpaD, 99
  Insa-Pauer, 21
  LCM, 40
  NEW, 57
  NEWCGSMain, 57
  RGB, 103, 144
  RGröbnerBasisW, 97
  SRGB, 28
  Suzuki-Sato, 40
  SYZ, 14
  Transform, 58
  UPDATE, 74
  WRGB, 25
alternative comprehensive Gröbner basis, *see* comprehensive Gröbner basis

block order, *see* term order
Boolean algebra, 66
boolean closed, 69
boolean closure, 70
boolean remainder, 70
Buchberger, 10, 72
Buchberger's algorithm, 10

carrier set, 66
coefficients, 5
Command
  bases2exp Suzuki, 49
  cgb_m PGB, 162
  cgb_m1 PGB, 162
  cgb_re PGB, 153
  cgb_re1 PGB, 153
  cgb PGB, 153
  cgb REDLOG, 47
  cgb1 PGB, 153
  cgbw_re PGB, 157
  cgbw_re1 PGB, 157
  cgbw PGB, 157
  cgbw1 PGB, 157
  cgs_con PGB, 150
  cgs_con1 PGB, 150
  cgs_con2 PGB, 150
  cgs_m PGB, 160
  cgs_m1 PGB, 160
  cgs_re PGB, 148
  cgs_re1 PGB, 148
  cgs PGB, 148
  cgs Suzuki, 49
  cgs1 PGB, 148

cgs2 PGB, 148
cgsw_re PGB, 156
cgsw PGB, 156
cgsw2 PGB, 156
dispgb Montes, 48
fcgb_re PGB, 154
fcgb_re1 PGB, 154
fcgb PGB, 154
fcgb1 PGB, 154
fcgbw_re PGB, 158
fcgbw_re1 PGB, 158
fcgbw PGB, 158
fcgbw1 PGB, 158
fcgs_m PGB, 164
fcgs_m1 PGB, 164
finalcases Montes , 48
groebner_module PGB, 165
gsys REDLOG, 47
newmat Risa/Asir, 157
p_syzygy PGB, 166
regb_pp PGB, 165
ret_list PGB, 163
std singular, 15
sub4cgb PGB, 154
sub4cgs PGB, 151
sub4cgsbmodule PGB, 162
sub4cgsmodule PGB, 160
syz singular, 15
syzygy PGB, 166
torder REDUCE, 47
tplot Montes , 48
complex numbers, 5
comprehensive Gröbner basis
commutative polynomial ring, 36
module, 132
rings of diff. ope., 113
alternative, 76
alternative comprehensive Gröbner basis on variety, 80
discrete, 81
comprehensive Gröbner system
commutative polynomial rings, 34
rings of diff. ope., 106
modules, 127
computer algebra system
Magma, 26
Maple, 34
Mathematica, 34
Reduce, 34
Risa/Asir, 15, 26, 34, 129, 147
singular, 15, 26, 34, 129
Cox, 5

DCGB, 81
differential operator, 93
dimension, 81
discrete comprehensive Gröbner basis, *see* comprehensive Gröbner basis
Dolzman, 33
DPGB, 48

extended Gröbner bases algorithm, 14

faithful, 37, 114
module, 132
first criterion, 72
free module, 11

generic case, 36
Gröbner basis
reduced $K[\bar{A}][\bar{X}]^r$, 143
reduced, von Neumann regular rings, 71
$K[\bar{A}][\bar{X}, \bar{D}]$, 105
$K[\bar{A}][\bar{X}]$, 20
$K[\bar{A}][\bar{X}]^r$, 124
$K[\bar{X}]$, 10
$K[\bar{X}][\bar{D}]$, 98
$K[\bar{X}]^r$, 13
reduced, 11
reduced $K[\bar{X}][\bar{D}]$, 103
stratified, 72
von Neumann regular rings, 71
grade, 99
graded lexicographic order, *see* term order
graded reverse lexicographic order, *see* term order
Greuel, 13

hybrid module order 1, 125
hybrid module order 2, 133

ideal, 6
Boolean algebra, 66
idempotent, 65
initial form, 94
Insa, 17, 97
integers, 5

Kalkbrener, 38
Kredel, 105

leading coefficient
$K[\bar{A}][\bar{X}]$, 17
$K[\bar{A}][\bar{X}]^r$, 123
$K[\bar{X}]$, 8
$K[\bar{X}]^r$, 12
von Neumann regular ring, 69
leading monomial
$K[\bar{A}][\bar{X}]$, 17
$K[\bar{A}][\bar{X}]^r$, 124
$K[\bar{X}]$, 8
$K[\bar{X}][\bar{D}]$, 94
$K[\bar{X}]^r$, 12
von Neumann regular ring, 69
leading power product
$K[\bar{A}][\bar{X}]$, 17
von Neumann regular ring, 69
$K[\bar{A}][\bar{X}]^r$, 123
$K[\bar{X}]$, 8
$K[\bar{X}][\bar{D}]$, 94
$K[\bar{X}]^r$, 12
least common multiple
$K[\bar{X}]$, 8
$K[\bar{X}]^r$, 12
lexicographic order, *see* term order

Maple, 48, 49
Mathematica, 49

matrix order, *see* term order
maximal ideal, 67
maximally independent, 81
module order, 12
module power product, 11
monic, 72
monomial, 5
Montes, 33, 48

natural numbers, 5
Noetherian ring, 42
normally ordered expression, 93

parametric Gröbner bases, 34
partial derivative, 93
Pauer, 17, 97
Pfister, 13
PGB, 147
polynomial, 5
POT (position-over-term), 12
power product, 5
  set of power products, 6
preterrace, 78
prime ideal, 66
prime spectrum, 68

quasi inverse, 65

radical, 6
rational numbers, 5
real numbers, 5
REDLOG, 34, 47
Reduce, 34, 47
reduced
  see Gröbner basis 11
reduction
  Reduce1 module, 143
  Reduce1 rings of diff. ope., 96
  Reduce2 module, 144
  Reduce2 rings of diff. ope., 98
  Reduction1 $K[\bar{X}]$, 9
  Reduction2, 19
  von Neumann regular ring, 69
reverse lexicographic order, *see* term order
Risa/Asir, 15, 49
Robbiano, 7

S-polynomial
  Spoly1, 9
  Spoly1 rings of diff. ope., 95
  Spoly2, 20
  Spoly2 ring of diff. ope., 98
  von Neumann regular rings, 71
Saracino, 65
Sato, 33
second criterion, 72
segment, 106
  commutative polynomial ring, 34
set of initial form, 94
set of monomials
  $K[\bar{A}][\bar{X}]$, 18
  $K[\bar{A}][\bar{X}]^r$, 124
  $K[\bar{X}]$, 8
  $K[\bar{X}]^r$, 12
set of normally ordered monomials, 94
set of power products, *see* power product
  von Neumann regular ring, 69
SICStus Prolog, 83
singular, 15, 34, 49
singular case, 36
Socrates, 1
specialization, 34, 38
stable
  $K[\bar{A}][\bar{X}, \bar{D}]$, 106
  $K[\bar{A}][\bar{X}]$, 38
  $K[\bar{A}][\bar{X}]^r$, 126
stratified, *see* Gröbner basis
strong reduced Gröbner basis, 27
Sturm, 33
support, 78
Suzuki, 33, 49
Suzuki-Sato algorithm, *see* Algorithm
syzygy, 11

term, 5
term order, 6
  block order, 7
  graded lexicographic order, 6
  graded reverse lexicographic order, 7
  lexicographic order, 6
  matrix order, 7
  reverse lexicographic order, 6
  weight order, 7
terrace, 78
TOP (term-over-position), 12
total degree, 5

variable, 5
von Neumann regular ring, 65

weak reduced Gröbner basis, 24
weight order, *see* term order
Weispfenning, 33, 65
Weyl algebra, 93
Winkler, 21, 99

zero-dimensional, 81
Zhou, 21, 99
Zhou-Winkler's criterion, 99

Curriculum Vitae

Katsusuke Nabeshima

Born on August 5, 1977, Kochi, Japan
Japanese Nationality
Single
E-Mail: Katsusuke.Nabeshima@risc.uni-linz.ac.at

Education

1997–2001	Bachelor Studies at Department of Mathematics and Physics (Mathematics course), Ritsumeikan University, Japan (B.Sc.)
2001–2003	Master Studies at Graduate School of Science and Engineering (Mathemat ical Science course), Ritsumeikan University, Japan (M.Sc.)
2003–	Ph.D. Studies at Research Institute for Symbolic Computation, Johannes Kepler University Linz, Austria

Publications

1. Nabeshima, K. (2007) Reduced Gröbner Bases in Polynomial Rings over a Polynomial Ring. *Mathematics in Computer Science.* To appear.
2. Nabeshima, K. (2007) A Speed-Up of the Algorithm for computing Comprehensive Gröbner Systems. In Brown, C. ed., *proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC'07).* ACM-Press. To appear.
3. Nabeshima, K. (2006) Reduced Gröbner Bases in Polynomial Rings over a Polynomial Ring. In Wang, D. and Zheng, Z. eds., *proceedings of the International Conference on Mathematical Aspects of Computer and Information Sciences (MACIS '06).* pp.15-32.
4. Nabeshima, K. (2005) A Direct Products of Fields Approach to Comprehensive Gröbner Bases over Finite Fields. In Zaharie, D., Petcu, D., Negru, V., Jebelean, T., Ciobanu, G., Cicortas, A., Abraham, A. and Paprzycki, M. eds., *proceedings of the International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC '05).* pp. 39-47, IEEE Computer Society.
5. Nabeshima, K. (2005) A computation method for ACGB-V. In Dlozmann, A., Seidl, A. and Thomas, S. eds., *proceedings of the A3L 2005, conference in Honor of the 60th Birthday of Volker Weispfenning.* pp. 171-180, BOD Norderstedt.
6. Sato, Y., Suzuki, A. and Nabeshima, K. (2003) ACGB on Varieties. In Ganzha, V.F., Mayr, E.W. and Vorozhtsov, E.V. eds., *proceedings of the International Workshop on Computer Algebra in Scientific Computing (CASC '03).* pp 313-318, Universität München.
7. Sato, Y., Suzuki, A. and Nabeshima, K. (2003) Discrete Comprehensive Grb̈ner

Bases II. In Li, Z. and Sit, W. eds., *proceedings of the Asian Symposium on Computer Mathematics (ASCM 03), Computer Mathematics III, Lecture Notes Series on Computimg.* pp 240-247, World Scientific.
8. Sato, Y., Suzuki, A. and Nabeshima, K. (2002) Generalized Discrete Comprehensive Gröber Bases. *Proceedings of the CA-ALIAS.* pp 105-110. RIMS, Kyoto University.

Technical reports and preprints

1. Nabeshima, K. (2007) Comprehensive Gröbner Bases for modules.
2. Nabeshima, K. (2007) Two Kinds of Gröbner Bases in rings of Differential Operators and Their Comprehensive Gröbner Bases.
3. Nabeshima, K. (2007) PGB: A Package for Computing Parametric Gröbner bases and Related Objects.
`http://www.risc.uni-linz.ac.at/people/knabeshi/pgb/`

Invited talks

1. Nabeshima, K. (2006) Comprehensive Gröbner Bases for modules. KIAS-RIMS Joint Workshop on Computer Algebra, July 27-29, Korea Institute for Advanced Study (KIAS), Seoul, South Korea

Translations

1. Nabeshima, K. (2006) Derive 6.1 Japanese version. (Mathematical Software) Texas Instruments.