数理解析研究所講究録 1295 巻 2002 年 161-170

媒介変数を係数に含む多項式に対する Gröbner bases について

佐藤 洋祐*　　　　鈴木 晃†
立命館大学理工学部　　神戸大学自然科学研究科

Abstract

We give an alternative definition of comprehensive Gröbner bases in terms of Gröbner bases in polynomial rings over commutative Von Neumann regular rings. Our comprehensive Gröbner bases are defined as Gröbner bases in polynomial rings over certain commutative Von Neumann regular rings, hence they have two important properties which do not hold in standard comprehensive Gröbner bases. One is that they have canonical forms. Another one is that we can define monomial reductions which are compatible with any instantiation.

1 Introduction

Let R be a commutative ring and S be any non-empty set. Then the set of all functions from S to R denoted by R^S becomes a commutative ring by naturally defining an addition and a multiplication of functions. Furthermore, this ring becomes a commutative Von Neumann regular ring if R is a commutative Von Neumann regular ring. Therefore, in case it is computable, we can construct Gröbner bases in polynomial rings over R^S. For such Gröbner bases, we have the following theorem.

Theorem. Let $G = \{g_1, \dots, g_k\}$ be a reduced Gröbner basis of an ideal $\langle f_1, \dots, f_l\rangle$ in a polynomial ring $R^S[\bar{X}]$, then for each element a of S, $\{g_1(a), \dots, g_k(a)\}$ becomes a reduced Gröbner basis of the ideal $\langle f_1(a), \dots, f_l(a)\rangle$ in a polynomial ring $R[\bar{X}]$. Where $h(a)$ denotes a polynomial in $R[\bar{X}]$ given from a polynomial h of $R^S[\bar{X}]$ with replacing its each coefficient c by $c(a)$. (see theorem 2.3 of [5])

This observation leads us to have an alternative definition of comprehensive Gröbner bases. Let K be a field and $f_1(A_1, \dots, A_m, \bar{X})$, ..., $f_k(A_1, \dots, A_m, \bar{X})$ be polynomials in $K[A, \dots, A_m, \bar{X}]$ with parameters $A_1, \dots, A_m$. Considering each polynomial $f(A_1, \dots, A_m)$ in $K[A_1, \dots, A_m]$ as a function from K^m to K, $f_1(A_1, \dots, A_m, \bar{X})$, ..., $f_k(A_1, \dots, A_m, \bar{X})$ become polynomials in $K^{(K^m)}[\bar{X}]$. If we can construct a reduced Gröbner basis G of the ideal $\langle f_1(A_1, \dots, A_m, \bar{X}), \dots, f_k(A_1, \dots, A_m, \bar{X})\rangle$ in a polynomial ring $K^{(K^m)}[\bar{X}]$ over a commutative Von Neumann regular ring $K^{(K^m)}$ somehow, we can consider G as a kind of comprehensive Gröbner basis of $\langle f_1(A_1, \dots, A_m, \bar{X}), \dots, f_k(A_1, \dots, A_m, \bar{X})\rangle$ with parameters $A_1, \dots, A_m$, since an instantiation of $A_1, \dots, A_m$ with any elements $a_1, \dots, a_m$ of K becomes a reduced Gröbner basis of the ideal $\langle f_1(a_1, \dots, a_m, \bar{X}), \dots, f_k(a_1, \dots, a_m, \bar{X})\rangle$ in $K[\bar{X}]$ by the theorem above.

In order to enable the above computation, it suffices to establish a way to handle the smallest commutative Von Neumann regular ring that includes $K[A_1, \dots, A_m]$. If the quotient field $K(A_1, \dots, A_m)$ corresponds to it, the situation would be very nice. Unfortunately, however, it does not work. Consider

*ysato@theory.cs.ritsumei.ac.jp
†sakira@kobe-u.ac.jp

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 $\varphi(0, a_2, \ldots, a_m) = 0$ and $\varphi(a_1, \ldots, a_m) = 1/a_1$ if $a_1 \neq 0$. Certainly φ is not a member of $K(A_1, \ldots, A_m)$.

In order to overcome this situation, we define a new algebraic structure called a *terrace*, which enables us to handle the smallest commutative Von Neumann regular ring that includes $K[A_1, \ldots, A_m]$. Using terraces we can compute a Gröbner basis in a polynomial ring over $K^{(K^m)}$. We call it an *ACGB*(Alternative Comprehensive Gröbner Basis). ACGB have the following two nice properties, which do not hold in standard comprehensive Gröbner bases.

1. There is a canonical form of an ACGB.

   Since an ACGB 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 of an ACGB.

2. We can use monomial reductions of an ACGB.

   Because of the same reason above, we can use monomial reductions of an ACGB. Moreover, it will be shown that monomial reductions are compatible with any instantiation of parameters.

In this paper we introduce our work on ACGB. We concentrate on the case that K is algebraically closed. We give some algorithms to handle terraces using classical Gröbner bases technique.

Our plan is as follows. In section 2, we give a definition of terraces with several algorithms to handle them. In section 3, we give a definition of ACGB. We prove several nice properties they have. In section 4, we give some computation examples we got through our implementation.

We assume the reader is familiar with Gröbner bases of polynomial rings over commutative Von Neumann regular rings. The reader is referred to [5], [2], or [3].

2 Terrace

In this section, we define a computable ring T and operations on T which witness that T forms a Von Neumann regular ring. For an arbitrary polynomial $f \in K[A_1, \ldots, A_n]$, we can consider it as a mapping $f \colon K^n \to K$, i.e., $f \in K^{(K^n)}$. So we can define the canonical embedding

$$\varphi \colon K[A_1, \ldots, A_n] \to K^{(K^n)}.$$

Let T be the closure of the image $\varphi[K[A_1, \ldots, A_n]]$ under addition, multiplication, and inverse in the Von Neumann regular ring $K^{(K^n)}$, hence T becomes a Von Neumann regular ring. We show a way to describe each element of T and define computable operations on T.

In the rest of this section, we fix an algebraically closed field K and a natural number n. We use the symbols $A_1, \ldots, A_n$ as variables. For each finite set of polynomials $\{f_1, \ldots, f_l\}$ in $K[A_1, \ldots, A_n]$, we denote the affine variety by $V(\{f_1, \ldots, f_l\})$, i.e.,

$$V(\{f_1, \ldots, f_l\}) = \{(a_1, \ldots, a_n) \in K^n : f_1(a_1, \ldots, a_n) = \cdots = f_l(a_1, \ldots, a_n) = 0\}.$$

We set $V(\emptyset) = K^n$ and $V(\{1\}) = \emptyset$ for convenience.

In order to handle elements of T such as $t \cdot t^{-1}$, we define an algebraic structure called a terrace.

2.1 Definition of preterraces

Definition 1

A triple $\langle s,t,r\rangle$ is called a preterrace *on $K[A_1,\ldots,A_n]$ if s and t are finite sets of polynomials in $K[A_1,\ldots,A_n]$ and $r = g/h$ for some $g,h \in K[A_1,\ldots,A_n]$ which satisfy*

1. $V(s) \subseteq V(t)$,
2. $(V(\{g\}) \cup V(\{h\})) \cap (V(t) \setminus V(s)) = \emptyset$, *i.e.,* $g(a_1,\ldots,a_n) \neq 0$ *and* $h(a_1,\ldots,a_n) \neq 0$ *for any* $(a_1,\ldots,a_n) \in V(t) \setminus V(s)$.

For a given preterrace $p = \langle s,t,r\rangle$, the support *of p (*supp*(p)) is the set $V(t)\setminus V(s) \subseteq K^n$. For a preterrace $p = \langle s,t,g/h\rangle$ on $K[A_1,\ldots,A_n]$ and $(a_1,\ldots,a_n) \in K^n$, we define $p(a_1,\ldots,a_n) \in K$ by*

$$p(a_1,\ldots,a_n) = \begin{cases} \dfrac{g(a_1,\ldots,a_n)}{h(a_1,\ldots,a_n)}, & \text{if } (a_1,\ldots,a_n) \in \mathrm{supp}(p), \\ 0, & \text{otherwise.} \end{cases}$$

(p can be considered as a member of T).

For an arbitrary polynomial $f \in K[A_1,\ldots,A_n]$, we define the corresponding preterrace pre(f) as follows:

$$\mathrm{pre}(f) = \langle \{f\}, \emptyset, f/1\rangle.$$

Note that $\mathrm{supp}(\mathrm{pre}(f)) = V(\emptyset) \setminus V(\{f\}) = \{(a_1,\ldots,a_n) \in K^n \ : \ f(a_1,\ldots,a_n) \neq 0\}$. Then we can easily see that $f(a_1,\ldots,a_n) = \mathrm{pre}(f)(a_1,\ldots,a_n)$ for any $(a_1,\ldots,a_n) \in K^n$.

Next we define the inverse and multiplicative operations on preterraces. The inverse p^{-1} of a preterrace $p = \langle s,t,g/h\rangle$ is defined by $p^{-1} = \langle s,t,h/g\rangle$ without changing the support. Note that we have

$$\begin{cases} p(a_1,\ldots,a_n)^{-1} = p^{-1}(a_1,\ldots,a_n), & \text{if } (a_1,\ldots,a_n) \in \mathrm{supp}(p) = \mathrm{supp}(p^{-1}) \\ p(a_1,\ldots,a_n) = p^{-1}(a_1,\ldots,a_n) = 0, & \text{otherwise.} \end{cases}$$

Hence p^{-1} represents the inverse of p in T.

In order to define the multiplication $p_1 \cdot p_2$ of preterraces $p_1 = \langle s_1,t_1,r_1\rangle$ and $p_2 = \langle s_2,t_2,r_2\rangle$ to represent the multiplication as elements of T, we need that

$$(p_1 \cdot p_2)(a_1,\ldots,a_n) = \begin{cases} p_1(a_1,\ldots,a_n)\cdot p_2(a_1,\ldots,a_n), & \text{if } (a_1,\ldots,a_n) \in \mathrm{supp}(p_1) \cap \mathrm{supp}(p_2), \\ 0, \ \ \text{otherwise.} \end{cases}$$

Note that we have $\mathrm{supp}(p_1) \cap \mathrm{supp}(p_2) = V(t_1 \cup t_2) \setminus V(\mathbf{Prod}(s_1 \cup t_2, s_2 \cup t_1))$, where, for finite set s,t of polynomials, $\mathbf{Prod}(s,t) = \{f\cdot g \ : \ f \in s, g \in t\}$. So we define the multiplication by $p_1 \cdot p_2 = \langle \mathbf{Prod}(s_1 \cup t_2, s_2 \cup t_1), t_1 \cup t_2, r_1 \cdot r_2\rangle$.

We can easily check that $p_1 \cdot p_2 = p_2 \cdot p_1$, $(p_1 \cdot p_2)\cdot p_3 = p_1 \cdot (p_2 \cdot p_3)$, and $p_1 \cdot \langle\{1\},\emptyset,1\rangle = p_1$ for any preterraces p_1, p_2, and p_3. Note that, for a preterrace $p = \langle s,t,r\rangle$, we have $p\cdot p^{-1} = \langle s,t,1\rangle$, which might not be equal to $\langle\{1\},\emptyset,1\rangle$ in general.

2.2 Definition of terraces

A sum of two preterraces as an element of T is not generally represented by a preterrace. We need another definition.

Definition 2
A finite set $\{p_1, \ldots, p_l\}$ *is called a* terrace *on* $K[A_1, \ldots, A_n]$ *if each* p_i $(i = 1, \ldots, l)$ *is a preterrace on* $K[A_1, \ldots, A_n]$ *such that* $\mathrm{supp}(p_i) \neq \emptyset$ *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*

$$\mathrm{supp}(t) = \bigcup_{p \in t} \mathrm{supp}(p) \subseteq K^n.$$

For a given terrace t and a sequence $(a_1, \ldots, a_n) \in K^n$, we define

$$t(a_1, \ldots, a_n) = \begin{cases} r(a_1, \ldots, a_n), & \text{if } (\exists p = \langle s, t, r \rangle \in t)\ (a_1, \ldots, a_n) \in \mathrm{supp}(p), \\ 0, & \text{otherwise.} \end{cases}$$

(The well-definedness is derived from the definition of terraces.) Hence, we consider t as an element of $K^{(K^m)}$, actually it is an elements of T since t represents $p_1 + \cdots + p_l$ in T. Intuitively a terrace is a representation of an element of T as a finite set of pairs of a rational function and a partition of K^m such that the rational function is not equal to 0 everywhere on its partition.

For a given finite set of preterraces, we can judge whether it forms a terrace or not by using the following algorithm **PreterraceIsZERO**. Indeed, for given two preterraces p and q, $\mathrm{supp}(p) \cap \mathrm{supp}(q) = \emptyset$ iff **PreterraceIsZERO**$(p \cdot q)$ returns True.

Algorithm **PreterraceIsZERO**

Specification: PreterraceIsZERO(P)
check whether a preterrace P satisfies $\mathrm{supp}(P) = \emptyset$ or not
Input: P is a preterrace on $K[A_1, \ldots, A_n]$
Output: return True if $\mathrm{supp}(P) = \emptyset$
return False otherwise
$\langle S, T, R \rangle := P$
IF $V(S) = V(T)$ THEN
RETURN True
ELSE
RETURN False

For a given preterrace p, we see that $p(a_1, \ldots, a_n) \neq 0$ for some $(a_1, \ldots, a_n) \in K^n$ if and only if $\mathrm{supp}(p) \neq \emptyset$ by the definition of preterraces. So the previous algorithm works as we desire.

The addition $t_1 + t_2$, the multiplication $t_1 \cdot t_2$, and the inverse t_1^{-1} of terraces t_1 and t_2 as elements of T are given as follows:

1. $(t_1 + t_2)(a_1, \ldots, a_n) = t_1(a_1, \ldots, a_n) + t_2(a_1, \ldots, a_n)$,
2. $(t_1 \cdot t_2)(a_1, \ldots, a_n) = t_1(a_1, \ldots, a_n) \cdot t_2(a_1, \ldots, a_n)$,
3. $t_1^{-1}(a_1, \ldots, a_n) = \begin{cases} 1/t_1(a_1, \ldots, a_n), & \text{if } t_1(a_1, \ldots, a_n) \neq 0, \\ 0, & \text{if } t_1(a_1, \ldots, a_n) = 0. \end{cases}$

We will define $t_1 + t_2$, $t_1 \cdot t_2$, and t_1^{-1} as terraces to preserve the above properties

We first concentrate on the case that t_1 and t_2 are singletons of preterraces, say $t_1 = \{p_1\}$ and $t_2 = \{p_2\}$ where $p_1 = \langle s_1, t_1, r_1 \rangle$ and $p_2 = \langle s_2, t_2, r_2 \rangle$. Note that $\mathrm{supp}(t_1) = \mathrm{supp}(p_1)$ and $\mathrm{supp}(t_2) =$

$\mathrm{supp}(p_2)$. Present $r_1 + r_2$ as an irreducible form g/h as an element of $K(A_1, \ldots, A_n)$. Let $p^{\cap}_{p_1,p_2} = \langle \mathbf{Prod}(\mathbf{Prod}(s_1 \cup t_2, s_2 \cup t_1), g), t_1 \cup t_2, g/h \rangle$, $p^{\setminus,(1)}_{p_1,p_2} = \langle \mathbf{Prod}(s_1, t_1 \cup t_2), t_1, r_1 \rangle$, $p^{\setminus,(2)}_{p_1,p_2} = \langle s_1 \cup s_2, t_1 \cup s_2, r_1 \rangle$. $p^{\setminus,(1)}_{p_2,p_1} = \langle \mathbf{Prod}(s_2, t_1 \cup t_2), t_2, r_2 \rangle$, and $p^{\setminus,(2)}_{p_2,p_1} = \langle s_1 \cup s_2, t_2 \cup s_1, r_2 \rangle$. Then the finite set $t = \{p \in \{p^{\cap}_{p_1,p_2}, p^{\setminus,(1)}_{p_1,p_2}, p^{\setminus,(2)}_{p_1,p_2}, p^{\setminus,(1)}_{p_2,p_1}, p^{\setminus,(2)}_{p_2,p_1}\} \ : \ \mathrm{supp}(p) \neq \emptyset\}$ of preterraces forms a terrace and satisfy $t(a_1, \ldots, a_n) = t_1(a_1, \ldots, a_n) + t_2(a_1, \ldots, a_n)$ for any $(a_1, \ldots, a_n) \in K^n$.

Using these notations, we define an additive operation on the set of the terraces. The following algorithm compute the addition of two terraces,

Algorithm **TerraceAdd**

Specification: $T \leftarrow$**TerraceAdd**(T_1, T_2)
Input: T_1, T_2 are terraces on $K[A_1, \ldots, A_n]$
Output: T is a terrace on $K[A_1, \ldots, A_n]$

```
  T := ∅
FOR each pair (p1, p2) ∈ T1 × T2 DO
  IF PreterraceIsZERO(p1 · p2) does not hold THEN
    S := {p∩_{p1,p2}, p\,(1)_{p1,p2}, p\,(2)_{p1,p2}, p\,(1)_{p2,p1}, p\,(2)_{p2,p1}}
    FOR each p ∈ S DO
      IF PreterraceIsZERO(p) does not hold THEN
        T := T ∪ {p}
      ENDIF
    END
  ENDIF
END
RETURN T
```

We define a terrace $t_1 + t_2$ as an output of **TerraceAdd**(t_1, t_2). It is easy to check the property 1 holds.

The definition of multiplication is rather simpler. The following algorithm compute the multiplication of two terraces.

Algorithm **TerraceMul**

Specification: $T \leftarrow$**TerraceMul**(T_1, T_2)
Input: T_1, T_2 are terraces on $K[A_1, \ldots, A_n]$
Output: T is a terrace on $K[A_1, \ldots, A_n]$

```
  T := ∅
FOR each p1 ∈ T1 and p2 ∈ T2 DO
  p := p1 · p2
  IF PreterraceIsZERO(p) does not hold THEN
    T := T ∪ {p}
  ENDIF
END
RETURN T
```

We define a terrace $t_1 \cdot t_2$ as an output of **TerraceMul**(t_1, t_2). It is easy to check the property 2 holds.

For an arbitrary terrace t, the inverse t^{-1} of t is defined by $t^{-1} = \{p^{-1} : p \in t\}$. It is trivial that t^{-1} forms a terrace and the property 3 holds. Now we have defined computable algorithms to compute operations on the terraces satisfying property 1, 2, 3.

We let $\text{TER} = \text{TER}(K[A_1, \ldots, A_n])$ be the set of terraces on $K[A_1, \ldots, A_n]$. We should note that, for a terrace $t \in \text{TER}$, there are infinitely many terraces $t' \in \text{TER}$ such that $t(a_1, \ldots, a_n) = t'(a_1, \ldots, a_n)$ for any $(a_1, \ldots, a_n) \in K^n$.

We define a binary relation $\sim$ on TER by

$$t \sim t' \iff t + \{\text{pre}(-1)\} \cdot t' = \emptyset.$$

Then the relation $\sim$ is a computable equivalence relation on TER.

Proposition 3

For arbitrary two terraces t and t' on $K[A_1, \ldots, A_n]$, $t \sim t'$ if and only if $t(a_1, \ldots, a_n) = t'(a_1, \ldots, a_n)$ for any $(a_1, \ldots, a_n) \in K^n$,.

It should be noted that there is only one terrace namely $\{\}$ which represents 0. We denote the set of the equivalence class $\text{TER}(K[A_1, \ldots, A_n])/\sim$ by $T_{(A_1,\ldots,A_n)}$. For a equivalence class $[t]_\sim \in T_{(A_1,\ldots,A_n)}$ and a sequence $(a_1, \ldots, a_n) \in K^n$, we define $[t]_\sim(a_1, \ldots, a_n) = t(a_1, \ldots, a_n) \in K$. The previous Proposition witnesses the well-definedness of $[t]_\sim(a_1, \ldots, a_n) \in K$. Moreover, using the Proposition, we can define addition, multiplication, and inverse on $T_{(A_1,\ldots,A_n)}$ by $[t]_\sim + [t']_\sim = [t + t']_\sim$, $[t]_\sim \cdot [t']_\sim = [t \cdot t']_\sim$, and $[t]_\sim^{-1} = [t^{-1}]_\sim$ for $t, t' \in \text{TER}(K[A_1, \ldots, A_n])$.

We can easily check that $T_{(A_1,\ldots,A_n)}$ is a Von Neumann regular ring, actually it is isomorphic to T which defined at the beginning of this section.

For a given polynomial $f \in K[A_1, \ldots, A_n]$, we define the corresponding equivalence class on terraces $\text{ter}_T(f) \in T_{(A_1,\ldots,A_n)}$ by

$$\text{ter}_T(f) = \begin{cases} [\{\text{pre}(f)\}]_\sim, & \text{if } f \in K[A_1, \ldots, A_n] \setminus \{0\}, \\ [\emptyset]_\sim, & \text{if } f = 0. \end{cases}$$

Note that $f(a_1, \ldots, a_n) = \text{ter}_T(f)(a_1, \ldots, a_n)$ for any $(a_1, \ldots, a_n) \in K^n$.

3 ACGB

In this section, we give an alternative comprehensive Gröbner bases. Let K be an algebraically closed field, TER be the set of the terraces on $K[A_1, \ldots, A_m]$ where $A_1, \ldots, A_m$ are variables, $T = \text{TER}/\sim$, and $\text{ter}_T\colon K[A_1, \ldots, A_m] \to T$ be the corresponding embedding. As we have seen in section 2, T is a commutative Von Neumann regular ring.

Definition 4

We extend ter_T to the embedding $\text{ter}_T\colon K[A_1, \ldots, A_m, X_1, \ldots, X_n] \to T[X_1, \ldots, X_n]$. by $\text{ter}_T(f_1\alpha_1 + \cdots + f_l\alpha_l) = \text{ter}_T(f_1)\alpha_1 + \cdots + \text{ter}_T(f_l)\alpha_l$ where $f_1, \ldots, f_l \in K[A_1, \ldots, A_m]$ and $\alpha_1, \ldots, \alpha_l$ are terms of $X_1, \ldots, X_n$.

Definition 5

For each $f(X_1, \ldots, X_n) = c_1\alpha_1 + \cdots + c_l\alpha_l \in T[X_1, \ldots, X_n]$ and elements $a_1, \ldots, a_m \in K$, we define $f_{(a_1,\ldots,a_m)}(X_1, \ldots, X_m) \in K[X_1, \ldots, X_m]$ by $f_{(a_1,\ldots a_m)}(X_1, \ldots, X_n) = c_1(a_1, \ldots, a_m)\alpha_1 + \cdots + c_l(a_1, \ldots, a_m)\alpha_l$ where $c_i \in T$ and α_i are terms of $X_1, \ldots, X_n$.

We redescribe some definitions and results which we need for our comprehensive Gröbner bases. The detailed argument is given in [5, 3].

Definition 6
A polynomial f is called boolean closed *if $(lc(f))^* f = f$. (where a^* is an element defined by $a \cdot a^{-1}$)*

Then we have the following property.

Theorem 7
Let G be a reduced Gröbner basis, then any element of G is boolean closed.

Definition 8
A reduced Gröbner basis G in a polynomial ring over a commutative Von Neumann regular ring is called stratified *Gröbner basis, when it satisfies the following two properties:*

- *$lc(g) = lc(g)^*$ for each $g \in G$,*
- *$lt(f) \neq lt(g)$ for any distinct elements f and g in G.*

We can calculate the stratified Gröbner basis for a given finite set of polynomials over a computable commutative Von Neumann regular ring. Now we prove the following theorem.

Theorem 9
For an algebraically closed field K, let T be the canonical set of equivalence classes on the terraces on $K[A_1,\dots,A_m]$, and let $\mathrm{ter}_T\colon K[A_1,\dots,A_m,X_1,\dots,X_n] \to T[X_1,\dots,X_n]$ be the corresponding embedding. For a given set $F = \{f_1(A_1,\dots,A_m,X_1,\dots,X_n),$ $\dots, f_k(A_1,\dots,A_m,X_1,\dots,X_n)\} \subseteq K[A_1,\dots,A_m,X_1,\dots,X_n]$, we let $\mathrm{ter}_T(F) = \{\mathrm{ter}_T(f_i) : i = 1,\dots,k\} \subseteq T[X_1,\dots,X_n]$, and let $G = \{g_1(X_1,\dots,X_n),\dots,g_l(X_1,\dots,X_n)\}$ be a Gröbner basis of $\mathrm{ter}_T(F)$ in $T[X_1,\dots,X_n]$ such that each element g_i is boolean closed. Then we have the following properties:

1. *For each elements $a_1,\dots,a_m \in K$, $G_{(a_1,\dots,a_m)} = \{g_{1(a_1,\dots,a_m)}(X_1,\dots,X_n),\dots,g_{l(a_1,\dots,a_m)}(X_1,\dots,X_n)\}\setminus \{0\}$ is a Gröbner basis of the ideal generated by $F(a_1,\dots,a_m)= \{f_1(a_1,\dots,a_m,X_1,\dots,X_n),\dots, f_k(a_1,\dots,a_m,X_1,\dots,X_n)\}$ in $K[X_1,\dots,X_n]$. Moreover, $G_{(a_1,\dots,a_m)}$ becomes a reduced Gröbner basis, in case G is stratified.*

2. *For any polynomial $h(X_1,\dots,X_n) \in T[X_1,\dots,X_n]$, we have $(h \downarrow_G)_{(a_1,\dots,a_m)}(X_1,\dots,X_n) = h_{(a_1,\dots,a_n)}(X_1,\dots,X_n) \downarrow_{G_{(a_1,\dots,a_m)}}$.*

By property 1, G can be considered as a kind of comprehensive Gröbner basis where $A_1,\dots,A_m$ are parameters, and so we call G an *ACGB* (Alternative Comprehensive Gröbner Basis.) Note that in the standard comprehensive Gröbner bases, we can not define monomial reductions before instantiation. In our algorithm, we can define monomial reductions, furthermore they are preserved by any instantiation.

4 Examples of computation

We implemented the algorithm to compute ACGB in the case K is the field of the complex numbers $\mathbb{C}$. In this section, we give some examples of our implementation.

Example 1.
Find the reduced Gröbner basis for the ideal generated by the following system of polynomials of the variables x, y with parameters a, b:

$$\begin{cases} ax^2y + 1, \\ bxy + abx + b. \end{cases}$$

In order to solve them simultaneously, compute a Gröbner basis of the ideal x in $T_{(a,b)}[x, y]$ where $T_{(a,b)}$ is the Von Neumann regular ring of equivalence classes on the terraces on $\mathbb{C}[a, b]$. Our program written in Risa/Asir [1] produces the following Gröbner basis in the graded reverse lexicographic order with $x > y$:

```
[[(V[a]-V[1],1)],
 [(V[0]-V[b*a],1)]*x+[(V[0]-V[b*a],(1)/(a^2))]*y+
   [(V[0]-V[b*a],(2)/(a))],
 [(V[0]-V[b*a],1)]*y^2+[(V[0]-V[-b*a],3*a)]*y+
   [(V[0]-V[-b*a],a^2)],
 [(V[b*a]-V[a],1)]*y*x^2+[(V[b*a]-V[a],(1)/(a))]]
```

Above output means that the reduced Gröbner basis is

$$\begin{cases} \langle 1 \rangle, & \text{if } a = 0, \\ \langle x + \dfrac{1}{a^2}y + \dfrac{2}{a}, y^2 + 3ay + a^2 \rangle, & \text{if } ab \neq 0, \\ \langle yx^2 + \dfrac{1}{a} \rangle, & \text{if } ab = 0, a \neq 0. \end{cases}$$

Example 2.
Find the reduced Gröbner basis for the ideal generated by the following system of polynomials of the variables x, y with parameters a, b, c:

$$\begin{cases} ax^2y + a + 3b^2, \\ a(b-c)xy + abx + 5c. \end{cases}$$

Then our program produces the following Gröbner basis which has six polynomials:

```
[[(V[a^2+3*b^2*a]-V[a+3*b^2],1),(V[b^2*a,c*b*a,c*a+3*c*b^2,c^2*a,
  a^2]-V[b^2*a,a^2,c],1)],
 [(V[(c*b-c^2)*a]-V[(b^2-c^2)*a*2+(3*b^4-3*c^4)*a,(c*b-c^2)*a],1),
  (V[(c*b^2-c^2*b)*a^2+(3*c*b^4-3*c^2*b^3)*a]-V[(c*b^2-c^2*b)*a],1),
  (V[(b^2-c*b)*a^2+(3*b^4-3*c^3*b)*a,(c*b^2-c^2*b)*a]-
  V[(b^2-c^2)*a^2+(3*b^4-3*c^4)*a,(c*b-c^2)*a],1),
  (V[(b-c)*a]-V[b*a,c*a],1)]*y+[(V[(c*b-c^2)*a]-V[(b^6-c^6)*a^4+
  (9*b^8-9*c^8)*a^3+(27*b^10-27*c^10)*a^2+(27*b^12-27*c^12)*a,
  (c*b-c^2)*a],(b)/(b-c)),(V[(b^2-c*b)*a^2+(3*b^4-3*c^3*b)*a,
  (c*b^2-c^2*b)*a]-V[(b^2-c^2)*a^2+(3*b^4-3*c^4)*a,(c*b-c^2)*a],
  (-25*c^2)/((-b^2+2*c*b-c^2)*a^2+(-3*b^4+6*c*b^3-3*c^2*b^2)*a)),
  (V[(b-c)*a]-V[-c*a^2-3*c^3*a,(b-c)*a],
  (-1/25*b^2*a^2-3/25*b^4*a)/(-c^2))],
 [(V[(b^2-c*b)*a^2+(3*b^4-3*c^3*b)*a,(c*b^2-c^2*b)*a]-
  V[(b^2-c^2)*a^2+(3*b^4-3*c^4)*a,(c*b-c^2)*a],1),
                                      ...
  (c*b^2-c^2*b)*a]-V[(b^2-c^2)*a^2+(3*b^4-3*c^4)*a,(c*b-c^2)*a],
  ((1/5*b-1/5*c)*a+3/5*b^3-3/5*c*b^2)/(-c))],
 [(V[(c*b^4-2*c^2*b^3+c^3*b^2)*a^4+(6*c*b^6-12*c^2*b^5+6*c^3*b^4)
  *a^3+(9*c*b^8-18*c^2*b^7+9*c^3*b^6+25*c^3*b^2-25*c^4*b)*a^2+
  (75*c^3*b^4-75*c^4*b^3)*a]-V[(c*b^2-c^2*b)*a^2+
  (3*c*b^4-3*c^2*b^3)*a],1),
                                       ...
  +(75*c^3*b^4-75*c^4*b^3)*a]-V[(c*b^2-c^2*b)*a^2+
  (3*c*b^4-3*c^2*b^3)*a],((-b^3+c*b^2)*a^2+(-3*b^5+3*c*b^4)*
  a-50*c^2*b)/((b^3-3*c*b^2+3*c^2*b-c^3)*a^2+
  (3*b^5-9*c*b^4+9*c^2*b^3-3*c^3*b^2)*a))],
 [(V[(b^2-c^2)*a^2+(3*b^4-3*c^4)*a,(c*b-c^2)*a]-V[(b-c)*a],1)]*y*x+
  [(V[(b^2-c^2)*a^2+(3*b^4-3*c^4)*a,(c*b-c^2)*a]-V[(b^2-c^2)*a,
  (c*b-c^2)*a],(b)/(b-c))]*x,
```

```
 [(V[(c*b^2-c^2*b)*a]-V[(b^2-c*b)*a],1)]*x^2+
 [(V[(c*b^2-c^2*b)*a]-V[(b^2-c*b)*a^2+(3*b^4-3*c^3*b)*a,
 (c*b^2-c^2*b)*a],((b-c)*a+3*b^3-3*c*b^2)/(-b*a))],
[(V[b*a,c*a]-V[a],1)]*y*x^2+[(V[b*a,c*a]-V[a],(a+3*b^2)/(a))]]
```

Looking at the first line of the output above, we can see that the ideal contains 1 if and only if $a = 0, b \neq 0$ or $a = 0, b = 0, c \neq 0$.

5 Conclusion and Remarks

Our algorithm of ACGB does not have a canonical representation in a completely syntactic form. There are infinitely many forms of equivalent terraces, although there is only one form (i.e. an empty set) to represent 0 as is mentioned in section 2. In this paper we employed rather naive methods to handle terraces. We did not use any sophisticated technique such as polynomial factorizations or computations of radical ideals or prime(primary) ideal decompositions. We did not use even ideal intersections but used ideal products. We need further computational experiments to find the most effective way.

We described our work under the assumption that K is algebraically closed. But this is not indispensable. What we actually need is computability of terrace. If we can compute terraces, then we can define and calculate ACGB. For example, when K is a real closed field, we can handle terraces using standard quantifier elimination technique.

Our ACGB gives us a direct information of a given system of polynomial equations with parameters. If we consider the following system of polynomial equations

$$\begin{cases} f_1(A_1, \ldots, A_m, \bar{X}) = 0 \\ \vdots \\ f_k(A_1, \ldots, A_m, \bar{X}) = 0 \end{cases}$$

in $K^{(K^m)}[\bar{X}]$, that is we consider each X_i as a function from K^m to K, an easy extension of Hilbert Nullstellensatz tells us that it has a solution if and only if $\langle f_1(A_1, \ldots, A_m, \bar{X}), \ldots, f_k(A_1, \ldots, A_m, \bar{X})\rangle \cap K^{(K^m)} = \emptyset$. Our ACGB gives us a direct answer to the question whether the system has a solution. It has a solution if and only if the ACGB of $\langle f_1(A_1, \ldots, A_m, \bar{X}), \ldots, f_k(A_1, \ldots, A_m, \bar{X})\rangle$ does not contain a constant.

References

[1] Noro, M and Takeshima, T. (1992) Risa/Asir – A Computer Algebra System. International Symposium on Symbolic and Algebraic Computation (ISSAC 92), Proceedings, 387–396.

[2] Sato, Y. (1998). A new type of canonical Gröbner bases in polynomial rings over Von Neumann regular rings. International Symposium on Symbolic and Algebraic Computation (ISSAC 98), Proceedings, 317–321

[3] Sato, Y. and Suzuki, A. (2001). Discrete Comprehensive Gröbner Bases. International Symposium on Symbolic and Algebraic Computation (ISSAC 2001), Proceedings, 292–296.

[4] Saracino, D., Weispfening, V. (1975). On algebraic curves over commutative regular rings, Model Theory and Algebra, a memorial tribute to A. Robinson, Springer LNM **498**, 307–387.

[5] Weispfenning, V. (1989). Gröbner bases in polynomial ideals over commutative regular rings, ROCAL '87, J. H. Davenport Ed., Springer LNCS **378**, 336–347.

[6] Weispfenning, V. (1992). Comprehensive Gröbner bases, J. Symb. Comp. 14/1, 1–29.