An Extended GCD Algorithm for Parametric Univariate Polynomials and Application to Parametric Smith Normal Form

Dingkang Wang
[1]KLMM, Academy of Mathematics and Systems Science, Chinese Academy of Sciences
Beijing 100190, China
[2]School of Mathematical Sciences, University of Chinese Academy of Sciences
Beijing, China
dwang@mmrc.iss.ac.cn

Hesong Wang
[1]KLMM, Academy of Mathematics and Systems Science, Chinese Academy of Sciences
Beijing 100190, China
[2]School of Mathematical Sciences, University of Chinese Academy of Sciences
Beijing, China
wanghesong2021@gmail.com

Fanghui Xiao
[1]KLMM, Academy of Mathematics and Systems Science, Chinese Academy of Sciences
Beijing 100190, China
[2]School of Mathematical Sciences, University of Chinese Academy of Sciences
Beijing, China
xiaofanghui@amss.ac.cn

ABSTRACT

An extended greatest common divisor (GCD) algorithm for parametric univariate polynomials is presented in this paper. This algorithm computes not only the GCD of parametric univariate polynomials in each constructible set but also the corresponding representation coefficients (or multipliers) for the GCD expressed as a linear combination of these parametric univariate polynomials. The key idea of our algorithm is that for non-parametric case the GCD of arbitrary finite number of univariate polynomials can be obtained by computing the minimal Gröbner basis of the ideal generated by those polynomials. But instead of computing the Gröbner basis of the ideal generated by those polynomials directly, we construct a special module by adding the unit vectors which can record the representation coefficients, then obtain the GCD and representation coefficients by computing a Gröbner basis of the module. This method can be naturally generalized to the parametric case because of the comprehensive Gröbner systems for modules. As a consequence, we obtain an extended GCD algorithm for parametric univariate polynomials. More importantly, we apply the proposed extended GCD algorithm to the computation of Smith normal form, and give the first algorithm for reducing a univariate polynomial matrix with parameters to its Smith normal form.

CCS CONCEPTS

• **Computing methodologies → Symbolic and algebraic algorithms**; **Algebraic algorithms**;

KEYWORDS

Extended greatest common divisor, Parametric univariate polynomial, Comprehensive Gröbner system, Smith normal form

ACM Reference Format:
Dingkang Wang, Hesong Wang, and Fanghui Xiao. 2020. An Extended GCD Algorithm for Parametric Univariate Polynomials and Application to Parametric Smith Normal Form. In *International Symposium on Symbolic and Algebraic Computation (ISSAC '20), July 20–23, 2020, Kalamata, Greece.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3373207.3404019

1 INTRODUCTION

The computation of polynomial greatest common divisor (GCD) is one of the most primitive computations in computer algebra with a wide range of applications that include simplifying rational expressions, partial fraction expansions, canonical transformations, mechanical geometry theorem proving, hybrid rational function approximation, and decoder implementation for error-correction; see [7, 10, 15, 17, 38]. It has been extensively studied and a crowd of algorithms have been constructed [8, 16, 23, 37]. Among them Euclidean algorithm which is the oldest algorithm for computing the GCD of two univariate polynomials and its variants are the most common algorithms. As an extension of polynomial GCD, parametric GCDs came into being. That is, the parameters space is decomposed into a finite number of constructible sets such that a GCD of the parametric polynomials is given uniformly in each constructible set. Abramov and Kvashenko [1] proposed an algorithm for computing the parametric GCD of two univariate polynomials with one parameter using sub-resultant chain. Ayad [2] studied the parametric GCD of several univariate polynomials with many parameters and mainly introduced two algorithms to compute the parametric GCD. Also with the idea of the comprehensive Gröbner system (CGS) introduced by Weispfenning [36], Nagasaka [27] extended the theories of Gianni and Trager [16], and Sasaki and Suzuki [31] which compute the GCD by Gröbner bases method to multivariate polynomials with parameters. Kapur et al. [19] proposed another algorithm for computing the parametric GCD of parametric multivariate polynomials. Besides, based on triangular set methods, Chen and Maza [9], and Bächler et al. [3] used subresultant chains and regular chains to compute parametric GCDs.

As for the extended polynomial GCD computation, of course it is also an important problem in symbolic algebraic computation and applications. To our knowledge, for non-parametric univariate polynomials, there are two kinds of algorithms to compute the extended GCD. One is the well-known extended Euclidean algorithm,

ISSAC '20, July 20–23, 2020, Kalamata, Greece

ACM ISBN 978-1-4503-7100-1/20/07…$15.00
https://doi.org/10.1145/3373207.3404019

and the other is the algorithm for solving the extended GCD problem by means of Hankel matrix techniques which was proposed by Sendra and Llovet [32]. However, there is currently no algorithm for computing the extended parametric polynomial GCD.

In this paper, we present an algorithm for computing the extended GCD of parametric univariate polynomials. We begin to present our key idea from non-parametric case, then extend the method for computing the extended GCD of univariate polynomials to the parametric case.

As we known, the GCD d of univariate polynomials $f_1, \ldots, f_s$ can be obtained by computing the minimal Gröbner basis of the ideal $\langle f_1, \ldots, f_s \rangle$. To get the representation coefficients (or multipliers) $a_1, \ldots, a_s$ for the GCD expressed as a linear combination: $d = a_1 f_1 + \cdots + a_s f_s$, we construct a module generated by s column vectors $(f_1, \epsilon_1)^T, \ldots, (f_s, \epsilon_s)^T$, where $\{\epsilon_1, \ldots, \epsilon_s\}$ is a standard basis for s-dimensional vector space. Under the proper position over term (POT) monomial order, one computes a minimal Gröbner basis of this module in which there exists only one element $(d', a'_1, \ldots, a'_s)$ such that d' is nonzero. These are exactly what we want, i.e. $d = d'$ and $a_i = a'_i$ for $i = 1, \ldots, s$. Most importantly, using comprehensive Gröbner systems for modules which presented by Nabeshima [26] as the generalization of comprehensive Gröbner systems for polynomial rings studied by Weispfenning [36], this method can be naturally extended to the parametric case. Meanwhile, we also get a free basis for the syzygy module of given polynomials $f_1, \ldots, f_s$ as a by-product.

In the rest of this paper, we will apply the proposed extended GCD algorithm to the computation of the Smith normal form together with transforming matrices, which is different from the method presented by Storjohann in [33] for computing the Smith normal form and transforming matrices of an integer matrix using the modulo N extended GCD algorithm. The reduction of univariate polynomial matrices to the Smith normal form is very useful in many areas of system theory, for instance, the analysis and minimal realization of transfer function matrices of time-invariant linear dynamical systems [7, 30], and the existence of a solution to an integer programming problem [4]. A constructive proof of the uniqueness of the Smith form is given by Gantmakher [14]. This construction gives a basic algorithm for Smith form reduction and many other algorithms [6, 29] based on this have been proposed with the view to improving efficiency.

An essential step in the calculation of the Smith normal form is the calculation of the GCD and multipliers for each of its rows and columns. In order to get the GCD of each column (row), the algorithms in [6, 29] have to subtract multiples of the least degree polynomial in the corresponding column (row) of matrices, at any instant, from the others, until only one non-zero polynomial remains. The proposed extended GCD algorithm in this paper, however, can give the GCD and multipliers directly. What's more, our algorithm can be extended to parametric case naturally, which is, to our knowledge, the first algorithm for computing the Smith normal form of polynomial matrices with parameters. Also, it's worth mentioning that Corless et al. [11] presented an algorithm for computing the Jordan canonical form of a matrix in Frobenius (rational) canonical form where entries are polynomials with parameters.

This paper is organized as follows. In Section 2, we introduce some notations and definitions. The main results is presented in Section 3, where we start from the non-parametric case, giving the method for computing the extended GCD of univariate polynomials and extending this result to the parametric case. Consequently the extended GCD algorithm for parametric univariate polynomials is presented. In Section 4, we apply the proposed algorithm to the computation of Smith normal form. We end with some concluding remarks in Section 5.

2 PRELIMINARIES

In this section we will introduce some notations and definitions to prepare for the discussion of this article.

Let k be a field, L be an algebraic closed field containing k, $R = k[x]$ be the polynomial ring in the variable x (or $R = k[U][x]$ be the parametric polynomial ring with the parameters $U = \{u_1, \ldots, u_m\}$ and variable x). Generally, we use the letters f, g, h for single polynomials (or elements of the ring $k[x]$) and boldface letters $\mathbf{e}, \mathbf{f}, \mathbf{g}, \mathbf{h}$ for column vectors (that is, elements of the module $k[x]^s$).

In practice, we frequently consider such a very important class of modules as follows.

Definition 2.1. Let $(f_1, \ldots, f_s)$ be an ordered s−tuple with $f_i \in R$. The set of all $(a_1, \ldots, a_s)^T \in R^s$ such that $a_1 f_1 + \cdots + a_s f_s = 0$ is an R-submodule of R^s, called the **syzygy module** of $(f_1, \ldots, f_s)$, and denoted by $\mathrm{Syz}(f_1, \ldots, f_s)$.

Unlike vector spaces, modules need not have any generating set which is linearly independent. If a R-module have a module basis, that is, a generating set that is R-linearly independent, it is given a special name, **free module**.

For example, the R-module R^s is free. Let $\epsilon_1 = (1, 0, \cdots, 0)^T$, $\epsilon_2 = (0, 1, \cdots, 0)^T, \cdots, \epsilon_s = (0, 0, \cdots, 1)^T$, then $\{\epsilon_1, \cdots, \epsilon_s\}$ is a free basis of R^s.

Next, we introduce Gröbner bases and comprehensive Gröbner systems for modules.

Let $\succ$ be a monomial order on $k[x]$, and $\succ_s$ be a module order by extending $\succ$ in a position over term (POT) fashion to $k[x]^s$, that is, for $\alpha,\ \beta \in \mathbb{N}$, $x^\alpha \epsilon_i \succ_s x^\beta \epsilon_j$ if $i > j$, or $i = j$ and $x^\alpha \succ x^\beta$. For $f \in k[x], \mathbf{g} \in k[x]^s$, the leading term, leading coefficient, and leading monomial of f and $\mathbf{g}$ with respect to $\succ$ and $\succ_s$ respectively are conveniently denoted by $\mathrm{LT}(f)$, $\mathrm{LC}(f)$, $\mathrm{LM}(f)$, $\mathrm{LT}(\mathbf{g})$, $\mathrm{LC}(\mathbf{g})$, and $\mathrm{LM}(\mathbf{g})$.

The definition of Gröbner bases for submodules is as follows.

Definition 2.2. Let $R = k[x]$ and M be a submodule of R^s, and let $\succ_s$ be a monomial order on $k[x]^s$.

(1) We will denote by $\langle \mathrm{LT}(M) \rangle$ the monomial submodule generated by the leading terms of all $\mathbf{g} \in M$ w.r.t. $\succ_s$.
(2) A finite collection $G = \{\mathbf{g}_1, \ldots, \mathbf{g}_t\} \subset M$ is called a **Gröbner basis** for M if $\langle \mathrm{LT}(M) \rangle = \langle \mathrm{LT}(\mathbf{g}_1), \cdots, \mathrm{LT}(\mathbf{g}_t) \rangle$.

The following are about the definitions of minimal and reduced Gröbner bases for modules.

Definition 2.3. Let $G = \{\mathbf{g}_1, \ldots, \mathbf{g}_t\}$ be a Gröbner basis for $M \subset k[x]^s$ with respect to a monomial order $\succ_s$.

(1) G is said to be **minimal**, if $\mathrm{LM}(\mathbf{g}) \notin \langle \mathrm{LM}(G \backslash \{\mathbf{g}\}) \rangle$ for all $\mathbf{g} \in G$.

(2) G is said to be **reduced**, if $\mathrm{LC}(\mathbf{g}) = 1$ and no monomial of $\mathbf{g}$ lies in $\langle \mathrm{LM}(G\backslash\{\mathbf{g}\})\rangle$.

Now we introduce some definitions for parametric univariate polynomials. For $\mathbf{g} \in k[U][x]^s$, $\mathrm{LC}_x(\mathbf{g})$ denotes the leading coefficient of $\mathbf{g}$ with respect to the variable x under the order $\succ_s$.

A **specialization** of $k[U]$ is a homomorphism $\sigma : k[U] \longrightarrow L$. In this paper, we only consider the specializations induced by the elements in L^m. That is, for $\alpha = (\alpha_1, \dots, \alpha_m) \in L^m$, the induced specialization σ_α is defined as

$$\sigma_\alpha : f \longrightarrow f(\alpha),$$

where $f \in k[U]$. Every specialization σ: $k[U] \longrightarrow L$ extends canonically to a specialization σ: $k[U][x]^s \longrightarrow L[x]^s$ by applying σ coefficient-wise.

For an ideal $E \subset k[U]$, the variety defined by E in L^m is denoted by $\mathbb{V}(E) = \{\alpha \in L^m \mid f(\alpha) = 0 \text{ for all } f \in E\}$. $A = \mathbb{V}(E) \setminus \mathbb{V}(N)$ is an algebraically constructible set, where E, N are ideals in $k[U]$.

For parametric systems, the definitions of comprehensive Gröbner systems and minimal comprehensive Gröbner systems for modules are given below.

Definition 2.4. Let F be a subset of $k[U][x]^s$, S be a subset of L^m, $G_1, \dots, G_l$ be subsets of $k[U][x]^s$, and $A_1, \dots, A_l$ be algebraically constructible subsets of L^m such that $S = \bigcup_{i=1}^l A_i$. A finite set $\mathcal{G} = \{(A_1, G_1), \dots, (A_l, G_l)\}$ is called a **comprehensive Gröbner system** (CGS) on S for F if $\sigma_\alpha(G_i)$ is a Gröbner basis of the submodule $\langle\sigma_\alpha(F)\rangle \subset L[x]^s$ for $\alpha \in A_i$ and $i = 1, \dots, l$. Each (A_i, G_i) is called a branch of $\mathcal{G}$. In particular, if $S = L^m$, then $\mathcal{G}$ is called a comprehensive Gröbner system for F.

Definition 2.5. A comprehensive Gröbner system $\mathcal{G} = \{(A_1, G_1), \cdots, (A_l, G_l)\}$ on S for $M \subset k[U][x]^s$ is said to be **minimal** (**reduced**) under some monomial order $\succ_s$, if for each $i = 1, \dots, l$,

(1) $A_i \neq \varnothing$, and furthermore, for each $i, j = 1, \cdots, l$, $A_i \cap A_j = \emptyset$ whenever $i \neq j$, and
(2) $\sigma_\alpha(G_i)$ is a minimal (reduced) Gröbner basis of $\langle\sigma_\alpha(F)\rangle \subset L[x]^m$ for $\alpha \in A_i$, and
(3) for each $\mathbf{g} \in G_i \neq \{\mathbf{0}\}$, $\sigma_\alpha(\mathrm{LC}_x(\mathbf{g})) \neq 0$ for $\alpha \in A_i$.

Remark 1. *For the computation of CGSs for modules, there exists an algorithm given by Nabeshima[26] which is based on the results proposed by Suzuki and Sato [35]. Moreover, there exist various algorithms to compute the minimal CGS for polynomial rings; see [18, 22, 24, 25, 34, 35] and so on. These algorithms can be extended to the case of modules. In this paper, we extend the KSW algorithm for computing CGSs over polynomial rings presented by Kapur et al. [20, 21] to the case of modules and then compute CGSs for modules since the KSW algorithm generates fewer branches and is the most efficient algorithm so far.*

Finally, we introduce the GCD systems for parametric univariate polynomials.

Definition 2.6. Let $F = \{f_1, \cdots, f_s\}$ be a subset of $k[U][x]$, S be a subset of L^m and $d_1, \dots, d_l$ be parametric univariate polynomials in $k[U][x]$, and $A_1, \dots, A_l$ be algebraically constructible subsets of L^m such that $S = \bigcup_{i=1}^l A_i$ and $A_i \cap A_j = \emptyset$ for $i \neq j$. A finite set $\mathcal{D} = \{(A_1, d_1), \dots, (A_l, d_l)\}$ is called a **GCD system** on S for F if $\sigma_\alpha(d_i)$ is a GCD of $\sigma_\alpha(F) \subset L[x]$ for $\alpha \in A_i$ and $i = 1, \dots, l$. Moreover, for each $d_i \neq 0$, $\sigma_\alpha(\mathrm{LC}_x(d_i)) \neq 0$ for $\alpha \in A_i$. Each (A_i, d_i) is regarded as a branch of $\mathcal{D}$. In particular, $\mathcal{D}$ is simply called a GCD system for F if $S = L^m$.

3 THE PROPOSED ALGORITHM

As stated in the introduction, there is currently no algorithm for computing extended GCD of parametric univariate polynomials.

In this section, we are devoted to giving an extended GCD algorithm for parametric univariate polynomials. Since the algorithm based on Gröbner bases is more suitable to be generalized to the parametric case because of the CGS, by means of structural features of the module and by constructing a special module we compute the GCD and obtain an extended GCD algorithm based on the computation of Gröbner bases for modules, which can be naturally generalized to the parametric case.

Now, let us introduce what is to be stated in this section. We first present the key idea for computing the extended GCD of any finite number of non-parametric univariate polynomials, and then generalize it to the parametric case. As a consequence, we propose an algorithm based on CGSs for modules to compute the extended GCD system for a set of parametric univariate polynomials.

3.1 Extended GCD for univariate polynomials

Let $R = k[x]$ and $f_1, \cdots, f_s \in R$. Assume $d = \mathrm{GCD}(f_1, \cdots, f_s)$. Since R is a principal ideal domain (PID), then there are $a_1, \dots, a_s \in R$ such that $a_1 f_1 + \cdots + a_s f_s = d$, and we call $a_1, \dots, a_s$ **representation coefficients** for the GCD (not unique).

As we all know, one can obtain a GCD d by computing a Gröbner basis of the ideal generated by $f_1, \cdots, f_s$. Nevertheless, in many case we have to solve the problem: how can we get $a_1, \dots, a_s$ and d simultaneously? Next, we share our approach.

Before presenting the main theorem, there are several lemmas to be rendered. For the first lemma below, we can refer to [13].

Lemma 3.1. *Let $R = k[x]$ and suppose that $f_1, \dots, f_s \in R$ are polynomials that are not all zero. Then $Syz(f_1, \dots, f_s)$ is a free module with $s - 1$ generators.*

Therefore, the syzygy module M over $R = k[x]$ as a free module has two sets of bases: the free basis and the Gröbner basis under some monomial order, denoted by F and G respectively. Generally speaking, $|G| \geq |F|$, where "$|\cdot|$" represents the number of elements in the set. The proof is as follows.

Lemma 3.2. *Let $M \subset R^s$ be a free R-module, F and G be a free basis and a minimal Gröbner basis under some monomial order $\succ_s$ for M. Then $|G| \geq |F|$.*

Here we construct a module M and let's take a look at some of the properties of this module, which is from Exercise 15 of Chapter 5, Section 3 in [12].

Proposition 3.3. *Let $R' = k[x_1, \dots, x_n]$ be a polynomial ring with a monomial order $\succ$, and for any integer $s \geq 1$, we denote the standard basis of R'^{s+1} by $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{s+1}$. Let $\succ_{s+1}$ denote the POT extension of $\succ$ to R'^{s+1} with $\mathbf{e}_1 \succ \mathbf{e}_i$ for $2 \leq i \leq s+1$. Given $f_1, \dots, f_s \in R'$, without loss of generality, assume that $f_1, \dots, f_s$ are not all zero. Then consider the submodule $M \subset R'^{s+1}$ generated by*

$$\mathbf{m}_i = f_i\mathbf{e}_1 + \mathbf{e}_{i+1} = (f_i, 0, \cdots, 0, 1, 0, \cdots, 0)^T, \quad i = 1, \cdots, s.$$

Let G be a minimal Gröbner basis of M with respect to $\succ_{s+1}$, then the following conclusions hold:

1. *If $(g, h_1, \ldots, h_s)^T \in M$, then $g = h_1 f_1 + \cdots + h_s f_s$.*
2. *$M \cap (\{0\} \times R'^s) = \{0\} \times Syz(f_1, \ldots, f_s)$.*
3. *The set $G' = \{g \in R' \mid g \neq 0 \ \wedge \exists\, h_1, \ldots, h_s \in R'$ s.t. $(g, h_1, \ldots, h_s)^T \in G\}$ is a minimal Gröbner basis with respect to $\succ$ for the ideal $\langle f_1, \ldots, f_s\rangle$.*
4. *The set G'' defined by $\{0\} \times G'' = G \cap (\{0\} \times R'^s)$ is a minimal Gröbner basis with respect to $\succ_s$ being the restriction of $\succ_{s+1}$ to R'^s for the syzygy module $Syz(f_1, \ldots, f_s)$.*

Proof. According to the construction of M, (1) and (2) are obvious. Besides, (3) and (4) are also obtained by the definition of G', G'', and Gröbner bases for modules w.r.t. $\succ_{s+1}$. □

In particular, for the case of univariate, there are better results.

Theorem 3.4. *With the above notations. If $R' = R = k[x]$ is a univariate polynomial ring, then $|G'| = 1$ and $|G''| = s - 1$. Therefore, $|G| = s$. Note that s is the number of these given polynomials.*

Proof. First, it follows from (3) of Proposition 3.3 and the univariate polynomial ring R' that $|G'| = 1$.

Now we prove that $|G''| = s-1$. By Lemma 3.1 and Lemma 3.2, we have $|G''| \geq s-1$. In the following all we need to do is to prove that $|G''| > s - 1$ is impossible. Let $|G''| = t$ and $G'' = \{\mathbf{g}_1'', \ldots, \mathbf{g}_t''\}$ where $\mathbf{g}_1'' \succ_s \cdots \succ_s \mathbf{g}_t''$. Suppose that $t > s - 1$, i.e. $t \geq s$. By Proposition 3.3 we know that G'' is the minimal Gröbner basis for $\mathrm{Syz}(f_1, \ldots, f_s)$, hence $\mathbf{g}_t''$ must be in the form: $\mathbf{g}_t'' = (0, \cdots, 0, g)^T$ where $g \in k[x]$ and $g \neq 0$ because R' is a univariate polynomial ring. This contradicts $\mathbf{g}_t'' \in \mathrm{Syz}(f_1, \ldots, f_s)$, so $|G''| = s - 1$. □

Combining Lemma 3.1 and Theorem 3.4, it is easy to know that G'' is a free basis for the syzygy module $\mathrm{Syz}(f_1, \ldots, f_s)$ where $f_1, \ldots, f_s \in k[x]$.

Theorem 3.5. *As above, assume $G = \{\mathbf{g}_1, \ldots, \mathbf{g}_s\}$ is a minimal Gröbner basis for $M \subset k[x]^{s+1}$ under the order $\succ_{s+1}$ with $\mathbf{e}_1 \succ \mathbf{e}_i$ for $2 \leq i \leq s+1$, and $\mathbf{g}_1 = (d, u_{11}, \ldots, u_{1s})^T$, $\mathbf{g}_j = (0, u_{j1}, \ldots, u_{js})^T$, $2 \leq j \leq s$. Then d is a GCD of $f_1, \ldots, f_s$ and $u_{11}, \ldots, u_{1s}$ are the corresponding representation coefficients for d as a linear combination of $f_1, \ldots, f_s$. Further, the matrix $\mathbf{U} = (u_{ij})_{s\times s} \in k[x]^{s\times s}$ is unimodular, that is, $\det(\mathbf{U}) \in k \setminus \{0\}$, and $\mathbf{Uf} = \mathbf{d}$, where*

$$\mathbf{U} = \begin{pmatrix} u_{11} & \cdots & u_{1s} \\ u_{21} & \cdots & u_{2s} \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ u_{s1} & \cdots & u_{ss} \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ \vdots \\ f_s \end{pmatrix}, \quad \mathbf{d} = \begin{pmatrix} d \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix}.$$

Proof. According to Proposition 3.3 and Theorem 3.4, $G' = \{d\}$ is a Gröbner basis of the ideal $\langle f_1, \ldots, f_s\rangle$, then d is a GCD of $f_1, \ldots, f_s$ and $u_{11}, \ldots, u_{1s}$ are the corresponding representation coefficients. Moreover, by the construction of the matrix $\mathbf{U}$, it's obvious that $\mathbf{Uf} = \mathbf{d}$. Now let's prove that $\mathbf{U}$ is a unimodular matrix. Since $G = \{\mathbf{g}_1, \ldots, \mathbf{g}_s\}$ is the minimal Gröbner basis for M, hence these generators $\mathbf{m}_1, \ldots, \mathbf{m}_s$ of M can be represented by $\mathbf{g}_1, \ldots, \mathbf{g}_s$. In other words, there exists matrix $\mathbf{V} \in k[x]^{s\times s}$ such that $(\mathbf{m}_1, \ldots, \mathbf{m}_s)^T = \mathbf{V}(\mathbf{g}_1, \ldots, \mathbf{g}_s)^T$. To make things clearer, let's write out $(\mathbf{m}_1, \ldots, \mathbf{m}_s)^T$ and $(\mathbf{g}_1, \ldots, \mathbf{g}_s)^T$ concretely.

$$\begin{pmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \vdots \\ \mathbf{m}_s^T \end{pmatrix} = \begin{pmatrix} f_1 & 1 & 0 & \cdots & 0 \\ f_2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ f_s & 0 & 0 & \cdots & 1 \end{pmatrix}, \quad \begin{pmatrix} \mathbf{g}_1^T \\ \mathbf{g}_2^T \\ \vdots \\ \mathbf{g}_s^T \end{pmatrix} = \begin{pmatrix} d & u_{11} & \cdots & u_{1s} \\ 0 & u_{21} & \cdots & u_{2s} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & u_{s1} & \cdots & u_{ss} \end{pmatrix}.$$

By $(\mathbf{m}_1, \ldots, \mathbf{m}_s)^T = \mathbf{V}(\mathbf{g}_1, \ldots, \mathbf{g}_s)^T$, we have $\mathbf{E}_s = \mathbf{VU}$, where $\mathbf{E}_s$ is the $s \times s$ unit matrix. So $\mathbf{U}$ is unimodular. □

Based on the results of Theorem 3.4 and 3.5, we can design an algorithm to compute the GCD of $f_1, \ldots, f_s$ and unimodular matrix $\mathbf{U}$, where the first row $u_{11}, \ldots, u_{1s}$ of $\mathbf{U}$ are the representation coefficients. That is, we only need to construct the module M by inputting polynomials $f_1, \ldots, f_s$ and then compute a minimal Gröbner basis for M with respect to $\succ_{s+1}$.

3.2 Extended GCD systems for parametric univariate polynomials

Now we are ready to generalize the above method to the parametric case by means of the CGS for modules, and get the following result.

Theorem 3.6. *Given $f_1, \ldots, f_s \in k[U][x]$ and a subset $S \subset L^m$. Let $\mathcal{G} = \{(A_i, G_i)\}_{i=1}^{l}$ be a minimal comprehensive Gröbner system of the module $M = \langle f_1\mathbf{e}_1 + \mathbf{e}_2, \ldots, f_s\mathbf{e}_1 + \mathbf{e}_{s+1}\rangle \subset k[U][x]^{s+1}$ on S with respect to an order $\succ_{s+1}$ extended from $\succ$ in a position over term fashion with $\mathbf{e}_1 \succ \mathbf{e}_i$ for $2 \leq i \leq s+1$. For each branch (A_i, G_i) where $G_i \neq \{\mathbf{0}\}$ we have the following results.*

1. *Let $G_i' = \{g \in k[U][x] \mid g \neq 0 \ \wedge \exists\, h_1, \ldots, h_s \in k[U][x]$ s.t. $(g, h_1, \ldots, h_s)^T \in G_i\}$, then $\sigma_\alpha(G_i')$ is a minimal Gröbner basis of the ideal $\langle\sigma_\alpha(f_1), \ldots, \sigma_\alpha(f_s)\rangle$ with respect to $\succ$ for any $\alpha \in A_i$, and $|G_i'| = 1$.*
2. *Let G_i'' be a set defined by $\{0\} \times G_i'' = G_i \cap (\{0\} \times k[U][x]^s)$, then $\sigma_\alpha(G_i'')$ is a minimal Gröbner basis of the syzygy module $\mathrm{Syz}(\sigma_\alpha(\mathrm{f}_1), \ldots, \sigma_\alpha(\mathrm{f}_s))$ with respect to $\succ_s$ for any $\alpha \in A_i$, and $|G_i''| = s - 1$. Thus, $\sigma_\alpha(G_i'')$ is a free basis of the syzygy module $\mathrm{Syz}(\sigma_\alpha(\mathrm{f}_1), \ldots, \sigma_\alpha(\mathrm{f}_s))$.*
3. *Assume $G_i = \{\mathbf{g}_1, \cdots, \mathbf{g}_s\}$ and $\mathbf{g}_1 = (d_i, u_{11}, \cdots, u_{1s})^T$, $\mathbf{g}_j = (0, u_{j1}, \cdots, u_{js})^T$ for $2 \leq j \leq s$. Then $\sigma_\alpha(d_i)$ is a GCD of $\sigma_\alpha(f_1), \ldots, \sigma_\alpha(f_s)$ and $\sigma_\alpha(u_{11}), \ldots, \sigma_\alpha(u_{1s})$ are the representation coefficients for $\sigma_\alpha(d_i)$ as a linear combination of $\sigma_\alpha(f_1), \ldots, \sigma_\alpha(f_s)$. Moreover, assume the matrix $\mathbf{U}_i = (u_{kj})_{s\times s}$, then $\sigma_\alpha(\mathbf{U}_i)\sigma_\alpha(\mathbf{f}) = \sigma_\alpha(\mathbf{d}_i)$ and $\sigma_\alpha(\mathbf{U}_i)$ is unimodular for any $\alpha \in A_i$, where*

$$\mathbf{U}_i = \begin{pmatrix} u_{11} & \cdots & u_{1s} \\ u_{21} & \cdots & u_{2s} \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ u_{s1} & \cdots & u_{ss} \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ \vdots \\ f_s \end{pmatrix}, \quad \mathbf{d}_i = \begin{pmatrix} d_i \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix}.$$

Particularly, for the branch (A_i, G_i) where $G_i = \{\mathbf{0}\}$, $\sigma_\alpha(d_i) = 0$ and $\sigma_\alpha(\mathbf{U}_i) = \mathbf{E}_s$ for $\alpha \in A_i$. In this case, the corresponding syzygy module $\mathrm{Syz}(\sigma_\alpha(\mathrm{f}_1), \ldots, \sigma_\alpha(\mathrm{f}_s))$ is $k[x]^s$.

Proof. Since $\mathcal{G}$ is a minimal comprehensive Gröbner system, in each branch (A_i, G_i) where $G_i \neq \{\mathbf{0}\}$, the set $\sigma_\alpha(G_i)$ is a minimal Gröbner basis of $\sigma_\alpha(M)$ for any $\alpha \in A_i$. Besides, there is no element in G_i specializing to $\mathbf{0}$ because the leading coefficients of all elements in G_i are non-zero under specialization. Thus, it is easy to derive the results from Proposition 3.3, Theorem 3.4 and 3.5. □

3.3 Parametric extended GCD algorithm

Based on Theorem 3.6, we are ready to give an algorithm to compute the extended GCD system for parametric univariate polynomials.

Theorem 3.7. *Algorithm 1 works correctly and terminates.*

Algorithm 1: Parametric extended GCD algorithm

```
Input  : $f_1, \dots, f_s \in k[U][x]$, a constructible set $A \subset L^m$,
         and a POT order $\succ_{s+1}$ with $\mathbf{e}_1 \succ \mathbf{e}_i, i \geq 2$.
Output : an extended GCD system $\{(A_i, \mathbf{U}_i, d_i)_{i=1}^{l}\}$, where
         GCD$(\sigma_\alpha(f_1), \dots, \sigma_\alpha(f_s) = \sigma_\alpha(d_i)$ and $\sigma_\alpha(\mathbf{U}_i)$ is
         unimodular for any $\alpha \in A_i$.
begin
    compute a minimal CGS $\{(A_i, G_i)_{i=1}^{l}\}$ for the module
      $M = \langle f_1\mathbf{e}_1 + \mathbf{e}_2, \dots, f_s\mathbf{e}_1 + \mathbf{e}_{s+1}\rangle$ w.r.t. $\succ_{s+1}$;
    for i from 1 to l do
        $G_i := \{u_0\mathbf{e}_1 + \sum_{j=1}^{s} u_{1j}\mathbf{e}_{j+1}, \sum_{j=1}^{s} u_{2j}\mathbf{e}_{j+1}, \dots, \sum_{j=1}^{s} u_{sj}\mathbf{e}_{j+1}\}$;
        $\mathbf{U}_i := \left(u_{kj}\right)_{s\times s},\ 1 \leq k, j \leq s$;
        $d_i := u_0$;
    return $\{(A_i, \mathbf{U}_i, d_i)\}_{i=1}^{l}$;
```

Proof. The correctness of Algorithm 1 directly follows from Theorem 3.6, and the termination of Algorithm 1 fully depends on that of the algorithm for computing CGSs of the module M which is obviously derived from the termination of KSW algorithm as mentioned in Remark 1. □

Remark 2. *For each $(A_i, \mathbf{U}_i, d_i)$, the components of the first row vector in $\mathbf{U}_i$ are the representation coefficients of d_i.*

We use the following simple example to illustrate the steps in the above proposed algorithm.

Example 3.8. Let $f_1, f_2, f_3 \in \mathbb{C}[U][x]$ be as follows:

$$f_1 = (x-a)^2, \quad f_2 = (x-b)^2, \quad f_3 = x(x-b),$$

where $U = \{a, b\}$ and $\succ$ is a lexicographic order.

Step 1: we compute a minimal CGS $\mathcal{G}$ for the module $M = \langle f_1\mathbf{e}_1 + \mathbf{e}_2, f_2\mathbf{e}_1 + \mathbf{e}_3, f_3\mathbf{e}_1 + \mathbf{e}_4\rangle \subset \mathbb{C}[a,b][x]^4$ with respect to $\succ_4$ where $\mathbf{e}_1 \succ \mathbf{e}_2 \succ \mathbf{e}_3 \succ \mathbf{e}_4$, and the result is shown in Table 1 where

Table 1: a minimal CGS $\mathcal{G}$ for the module M

No.	A_i	G_i
1	$\mathbb{C}^2\backslash\mathbb{V}(b(b-a))$	G_1
2	$\mathbb{V}(b)\backslash\mathbb{V}(a^2)$	G_2
3	$\mathbb{V}(a-b)\backslash\mathbb{V}(b)$	G_3
4	$\mathbb{V}(a,b)$	G_4

$$\begin{aligned} G_1 =& \{\, b(a-b)^2\mathbf{e}_1 + b\mathbf{e}_2 + (-2a+b)\mathbf{e}_3 + (2a-2b)\mathbf{e}_4, \\ & (bx-b^2)\mathbf{e}_2 + a^2\mathbf{e}_3 + (-bx-a^2+2ab)\mathbf{e}_4,\ x\mathbf{e}_3 + (b-x)\mathbf{e}_4\}; \\ G_2 =& \{\, a^3\mathbf{e}_1 + (a+2x)\mathbf{e}_2 + (3a-2x)\mathbf{e}_4,\ x^2\mathbf{e}_2 - (a^2-2ax+x^2)\mathbf{e}_4, \\ & \mathbf{e}_3 - \mathbf{e}_4\}; \\ G_3 =& \{\, (-b^2+bx)\mathbf{e}_1 - \mathbf{e}_3 + \mathbf{e}_4,\ \mathbf{e}_2 - \mathbf{e}_3,\ x\mathbf{e}_3 + (b-x)\mathbf{e}_4\}; \\ G_4 =& \{\, x^2\mathbf{e}_1 + \mathbf{e}_4,\ \mathbf{e}_2 - \mathbf{e}_4,\ \mathbf{e}_3 - \mathbf{e}_4\}. \end{aligned}$$

Step 2: according to G_i in the minimal CGS for module M, we construct $\mathbf{U}_i$ and d_i, where

$$d_1 = b(a-b)^2, \quad d_2 = a^3, \quad d_3 = -b^2 + bx, \quad d_4 = x^2.$$

$$\mathbf{U}_1 = \begin{pmatrix} b & -2a+b & 2a-2b \\ bx-b^2 & a^2 & -bx-a^2+2ab \\ 0 & x & b-x \end{pmatrix}, \mathbf{U}_2 = \begin{pmatrix} a+2x & 0 & 3a-2x \\ x^2 & 0 & -(a-x)^2 \\ 0 & 1 & -1 \end{pmatrix},$$

$$\mathbf{U}_3 = \begin{pmatrix} 0 & -1 & 1 \\ 1 & -1 & 0 \\ 0 & x & b-x \end{pmatrix}, \qquad \mathbf{U}_4 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & -1 \\ 0 & 1 & -1 \end{pmatrix}.$$

In summary, parametric GCDs are expressed as the linear representations of f_1, f_2, f_3 as follows.

$$\begin{cases} if\ a \neq b\ and\ b \neq 0,\ bf_1 + (-2a+b)f_2 + (2a-2b)f_3 = b(a-b)^2; \\ if\ a \neq b\ and\ b = 0,\ (a+2x)f_1 + 0\cdot f_2 + (3a-2x)f_3 = a^3; \\ if\ a = b\ and\ b \neq 0,\ 0\cdot f_1 - 1\cdot f_2 + 1\cdot f_3 = -b^2 + bx; \\ if\ a = b\ and\ b = 0,\ 0\cdot f_1 + 0\cdot f_2 + 1\cdot f_3 = x^2. \end{cases}$$

4 APPLICATION TO SMITH NORMAL FORM

4.1 Notations and definitions

In this subsection, we give some definitions and notations related to the Smith normal form. A matrix is called non-parametric (parametric) univariate polynomial matrix if its entries belong to $k[x]$ ($k[U][x]$).

Definition 4.1. Let $\mathbf{D}$ be an $s \times t$ matrix over $k[x]$ such that

(1) all (i,j)-entries in $\mathbf{D}$ are zero for $i \neq j$, that is, $\mathbf{D}$ is a diagonal matrix;
(2) each (i,i)–entry d_i in $\mathbf{D}$ is either monic or zero;
(3) $d_i \mid d_{i+1}$ for $1 \leq i < min\{s,t\}$.

Then $\mathbf{D} = diag(d_1, \dots, d_{min\{s,t\}})$ is said to be in Smith normal form, where "diag" stands for the diagonal matrix.

In addition, we give the following theorem appearing in [28] which ensures the existence of the Smith normal form for any univariate polynomial matrix $\mathbf{B}$ over $k[x]$.

Theorem 4.2. *Let $\mathbf{B}$ be an $s \times t$ matrix over $k[x]$, then there is a sequence of elementary operations over $k[x]$ which changes $\mathbf{B}$ into $S(\mathbf{B})$ that is in Smith normal form, called the Smith normal form of $\mathbf{B}$.*

That is, there exist unimodular matrices $\mathbf{U} \in k[x]^{s\times s}$, $\mathbf{V} \in k[x]^{t\times t}$ such that $\mathbf{UBV} = S(\mathbf{B})$.

4.2 The Smith normal form of parametric univariate polynomial matrix

For the non-parametric case, as stated in Theorem 4.2 any univariate polynomial matrix can be reduced to its Smith normal form under the elementary operations. As for the the parametric case, corresponding to each algebraically constructible subset $A_i \subset L^m$, the parametric univariate polynomials matrix under the specialization σ_α can be reduced to its Smith normal form by elementary operations, i.e. there exist parametric unimodular matrices $\mathbf{U} \in k[U][x]^{s\times s}$, $\mathbf{V} \in k[U][x]^{t\times t}$ such that $\sigma_\alpha(\mathbf{U})\sigma_\alpha(\mathbf{B})\sigma_\alpha(\mathbf{V}) = S(\sigma_\alpha(\mathbf{B}))$ for $\alpha \in A_i$. Now we discuss how to reduce a univariate polynomials matrix to its Smith normal form.

In the above section, we have proposed an extended GCD algorithm which not only can output the GCD, but also gives a unimodular matrix $\mathbf{U}$. In particular, $\mathbf{U}(\mathbf{f}_1, \dots, \mathbf{f}_s)^T = (d, 0, \dots, 0)$, where $\mathbf{f}_1, \dots, \mathbf{f}_s$ are given polynomials and d is the GCD of these polynomials. Then, we can apply the extended GCD algorithm to the calculation of the Smith normal form, and the actual practice is as follows.

Given $\mathbf{B} \in k[x]^{s\times t}$ (without loss of generality, assume $s \leq t$), we first call the extended GCD algorithm on the first column of $\mathbf{B}$ and obtain the unimodular matrix $\mathbf{U} \in k[x]^{s\times s}$. Then $\mathbf{U}$ acts on $\mathbf{B}$, and the first column of $\mathbf{UB}$ are zeros except for the first element. Next, do the same operation for the first row of the $\mathbf{UB}$, we still get a unimodular matrix $\mathbf{V} \in k[x]^{t\times t}$ such that the first row in $\mathbf{UBV}$ are zeros except for the first element, but note that the first column are not necessarily zeros. So we repeatedly perform the above operation in order to get a matrix in which the first column and row are zeros except for the (1,1)-component. This is the first step. If all other elements in the new obtained matrix can be divisible by the (1,1)-element, then we only need to conduct the same step as the first step on the lower right submatrix of this matrix. Otherwise, we need an extra step to ensure the divisibility relation. Finally we will get the Smith normal form of $\mathbf{B}$. Most importantly, these can be naturally extended to the parametric case.

Here we will give the algorithm for the parametric case. Before discussing the algorithm, we would like to introduce some useful propositions which are related to the termination of the algorithm.

As known to all, currently the algorithms are all computing the minimal CGS, and the minimal CGS for modules over parametric multivariate polynomial rings can't always be reduced to the reduced CGS. Here we show that for univariate polynomial rings it can be done.

Proposition 4.3. *A minimal CGS $\mathcal{G} = \{(A_1, G_1), \dots, (A_l, G_l)\}$ for module $M \subset k[U][x]^s$ with respect to the POT order $\succ_s$ can be reduced to a reduced CGS.*

Proof. By Definition 2.5, we only need to prove that for each branch (A_k, G_k) of $\mathcal{G}$ where $k = 1, \dots, l$, the parametric minimal Gröbner basis G_k for M can be reduced to the parametric reduced Gröbner basis on A_k. For any $\mathbf{g_i}, \mathbf{g_j} \in G_k$, suppose that $\mathrm{LM}(\mathbf{g_i}) = g_1\epsilon_i$ and $\mathrm{LM}(\mathbf{g_j}) = g\epsilon_j$. Without loss of generality, one can assume $\epsilon_i \succ \epsilon_j$ and the j-th component of $\mathbf{g_i}$ is f, then the $i-$th component of $\mathbf{g_j}$ must be zero. If f is reduced w.r.t. g (i.e. no monomial of f is divisible by $\mathrm{LM}(g)$), there is nothing to do. Otherwise do pseudo division to f by g, then one get $hf = qg + r$ where h is the power of the leading coefficient of g w.r.t. the main variable x and $\sigma_\alpha(h) \neq 0$ for any $\alpha \in A_k$. Thus, $h\mathbf{g_i} - q\mathbf{g_j} = \mathbf{g_i'}$ where $\mathbf{g_i'}$ is reduced w.r.t $\mathbf{g_j}$. Replacing $\mathbf{g_i}$ with $\mathbf{g_i'}$ and repeating the above process. Moreover, according to the definition of minimal CGS, $\sigma_\alpha(\mathrm{LC}_x(\mathbf{g})) \neq 0$ for any $\mathbf{g} \in G_k$ and $\alpha \in A_k$, then we can divide the coefficient such that $\sigma_\alpha(\mathrm{LC}_x(\mathbf{g})) = 1$, Thus, $\sigma_\alpha(G_k)$ is reduced. This proves the proposition. □

By the above proposition, we can get a new version of Algorithm 1 by computing a reduced CGS instead of a minimal CGS for M, denoted by Algorithm 1*.

Proposition 4.4. *Given $f_1, \cdots, f_s \in k[U][x]$, a constructible set $A \subset L^m$ and a POT order $\succ_{s+1}$ with $\mathbf{e}_1 \succ \mathbf{e}_{s+1} \succ \cdots \succ \mathbf{e}_2$. By Algorithm 1* we will get a reduced CGS $\{(A_i, G_i)\}_{i=1}^l$ and a GCD system $\{(A_i, \mathbf{U}_i, d_i)\}_{i=1}^l$, where $G_i = \{\mathbf{g}_1, \dots, \mathbf{g}_s\}$, $\mathbf{g}_1 = (d_i, u_{11}, \cdots, u_{1s})^T$, $\mathbf{g}_j = (0, u_{j1}, \cdots, u_{js})^T$ for $2 \leq j \leq s$. Then for any $\alpha \in A_i$, under the specialization σ_α, $\mathbf{u}_i = (u_{11}, \dots, u_{1s})^T$ is the minimal element in $M_i = \{(h_1, \dots, h_s)^T | h_1 f_1 + \cdots + h_s f_s = d_i\}$ under $\succ_s$ being the restriction of $\succ_{s+1}$ on $k[x]^s$.*

Proof. Assume that under σ_α, $\mathbf{u}_i$ is not minimal, then there exists $\mathbf{u}_i' \in M_i$ and $\sigma_\alpha(\mathbf{u}_i) \succ_s \sigma_\alpha(\mathbf{u}_i')$. By the definition of M_i, we have $\sigma_\alpha(\mathbf{u}_i - \mathbf{u}_i') \in \mathrm{Syz}(\sigma_\alpha(f_1), \dots, \sigma_\alpha(f_s))$. Thus $\mathrm{LM}(\sigma_\alpha(\mathbf{u}_i)) = \mathrm{LM}(\sigma_\alpha(\mathbf{u}_i - \mathbf{u}_i')) \in \mathrm{LM}(\mathrm{Syz}(\sigma_\alpha(f_1), \dots, \sigma_\alpha(f_s)))$. By Theorem 3.6, it implies that some term of $\sigma_\alpha(\mathbf{g}_1)$ is divisible by one of $\mathrm{LM}(\sigma_\alpha(\mathbf{g}_2))$, $\dots$, $\mathrm{LM}(\sigma_\alpha(\mathbf{g}_s))$, which contradicts that $\sigma_\alpha(G_i)$ is reduced. □

Now we give the algorithm for computing the Smith normal form of univariate polynomial matrices with parameters, and prove the termination of the algorithm.

Algorithm 2: Parametric Smith normal form algorithm

```
Input  : $\mathbf{B} \in k[U][x]^{s\times t}$, a constructible set $A \subset L^m$, and
         a POT order $\succ_{s+1}$ with $\mathbf{e}_1 \succ \mathbf{e}_{s+1} \succ \cdots \succ \mathbf{e}_2$.
Output : $\{[A_i, \mathbf{B}_i, \mathbf{U}_i, \mathbf{V}_i]\}_{i=1}^l$, where $\sigma_\alpha(\mathbf{U}_i)\sigma_\alpha(\mathbf{B})\sigma_\alpha(\mathbf{V}_i)$
         $= \sigma_\alpha(\mathbf{B}_i)$ and $\sigma_\alpha(\mathbf{B}_i)$ is in Smith normal form
         for any $\alpha \in A_i$.
begin
    $G := \{\}$; $G_1 := \{[A, \mathbf{B}, \mathbf{E}_s, \mathbf{E}_t, \mathbf{B}]\}$; $d := 0$;
    while $G_1$ is not empty do
        $[A_0, \mathbf{B}_0, \mathbf{U}_0, \mathbf{V}_0, \mathbf{S}_0]$:=$G_1[1]$; $G_1 := G_1 \setminus \{G_1[1]\}$;
        $H_1 :=$ Reduce2Zero$(A_0, \mathbf{S}_0)$;
        for $[A_i, \mathbf{B}_i, \mathbf{U}_i, \mathbf{V}_i]$ in $H_1$ do
            $H_2 :=$ Divisible$(A_i, \mathbf{B}_i)$;
            for $[A_j, \mathbf{B}_j, \mathbf{U}_j, \mathbf{V}_j]$ in $H_2$ do
                $\mathbf{U}_1 := \mathrm{diag}(\mathbf{E}_d, \mathbf{U}_j\mathbf{U}_i)$;
                $\mathbf{V}_1 := \mathrm{diag}(\mathbf{E}_d, \mathbf{V}_i\mathbf{V}_j)$;
                $\mathbf{B}_1 := \mathbf{U}_1\mathbf{B}_0\mathbf{V}_1$; $\mathbf{U} := \mathbf{U}_1\mathbf{U}_0$; $\mathbf{V} := \mathbf{V}_0\mathbf{V}_1$;
                if $d = s - 1$ then
                    $G := G \cup \{[A_j, \mathbf{B}_1, \mathbf{U}, \mathbf{V}]\}$;
                else
                    $d := d + 1$;
                    $G_1 := G_1 \cup \{[A_j, \mathbf{B}_1, \mathbf{U}, \mathbf{V}, \mathrm{SubMatrix}(\mathbf{B}_1, d)]\}$;
    return $G$;
```

In Algorithm 2, Reduce2Zero$(A_0, \mathbf{S}_0)$ stands for repeatedly calling Algorithm 1* on the first column and row of the matrix (matrices) for each algebraically constructible subset and the details is as follows. Divisible$(A_i, \mathbf{B}_i)$ is used to check whether all other elements in $\mathbf{B}_i$ can be divisible by (1,1)-element on A_i, if not, we need the extra step: adding the corresponding column in which the element which isn't divisible by (1,1)-element of $\mathbf{B}_i$ is to the first column of $\mathbf{B}_i$ and getting $\mathbf{B}_i'$, then performing Reduce2Zero$(A_i, \mathbf{B}_i')$. SubMatrix$(\mathbf{B}_1$, d) denotes the lower right submatrix of $\mathbf{B}_1$ which consists of the last $s - d$ rows and $t - d$ columns.

In Algorithm 3, CEGCD(A, $\mathbf{B}$) and REGCD$(A, \mathbf{B})$ stand for calling Algorithm 1* on the first column and row of matrix $\mathbf{B}$ on the constructible set A, respectively. IsZero$(A_{i_j}, \mathbf{B}_{i_j})$ is a subroutine to determine if the first column and row of $\mathbf{B}_{i_j}$ are zeros except for the (1,1)-element on algebraically constructible subset A_{i_j}.

Proposition 4.5. *Algorithm 2 terminates within finite steps.*

Proof. According to the design of the algorithm and above explain, we only need to prove that Algorithm 3 (Reduce2Zero$(A, \mathbf{B})$) terminates within finite steps. Since the original (1,1)-element of univariate polynomial matrix $\mathbf{B}$ has a definite degree and since

```
Algorithm 3: Reduce2Zero
    Input  : $\mathbf{B} \in k[U][x]^{s\times t}$, a constructible set $A \subset L^m$, and
             a POT order $\succ_{s+1}$ with $\mathbf{e}_1 \succ \mathbf{e}_{s+1} \succ \cdots \succ \mathbf{e}_2$.
    Output : $\{[A_i, \mathbf{B}_i, \mathbf{U}_i, \mathbf{V}_i]\}_{i=1}^{l}$, where $\sigma_\alpha(\mathbf{U}_i)\sigma_\alpha(\mathbf{B})\sigma_\alpha(\mathbf{V}_i)$
             $= \sigma_\alpha(\mathbf{B}_i)$ for any $\alpha \in A_i$ and the first column
             and row of $\mathbf{B}_i$ are zeros except for the (1,1)-
             element on $A_i$.
    begin
        $G := \{\}$; $G_1 := \{[A, \mathbf{B}, \mathbf{E}_s, \mathbf{E}_t]\}$;
        while $G_1$ is not empty do
            $[A_0, \mathbf{B}_0, \mathbf{U}_0, \mathbf{V}_0] := G_1[1]$; $G_1 := G_1 \setminus \{G_1[1]\}$;
            $H_1 :=$ CEGCD($A_0, \mathbf{B}_0$);
            for $[A_i, \mathbf{U}_i, d_i]$ in $H_1$ do
                $\mathbf{B}_i := \mathbf{U}_i\mathbf{B}_0$; $\mathbf{U}_i := \mathbf{U}_i\mathbf{U}_0$;
                $H_2 :=$ REGCD($A_i, \mathbf{B}_i$);
                for $[A_{i_j}, \mathbf{V}_{i_j}, d_{i_j}]$ in $H_2$ do
                    $\mathbf{B}_{i_j} := \mathbf{B}_i\mathbf{V}_{i_j}^T$; $\mathbf{V}_{i_j} := \mathbf{V}_0\mathbf{V}_{i_j}^T$;
                    if IsZero($A_{i_j}, \mathbf{B}_{i_j}$) then
                        $G := G \cup \{[A_{i_j}, \mathbf{B}_{i_j}, \mathbf{U}_i, \mathbf{V}_{i_j}]\}$;
                    else
                        $G_1 := G_1 \cup \{[A_{i_j}, \mathbf{B}_{i_j}, \mathbf{U}_i, \mathbf{V}_{i_j}]\}$;
        return $G$;
```

the process of reducing the degree for the (1,1)-element cannot be continued indefinitely, after a finite times of loops the degree of (1,1)-element w.r.t. main variable x is stable and assume at the moment we get $\mathbf{B}_i$ of which the first column of are zeros except for the (1,1)-element on A_i. Then $H_2 :=$ REGCD($\mathrm{A_i}, \mathrm{B_i}$), and we get a unimodular matrix $\mathbf{V}_{i_j}^T$ which can reduce the first row of $\mathbf{B}_i$ to be zeros on new algebraically constructible subset A_{i_j}. Since under the specialization, the degree of (b_{11}) is stable, b_{11} is the GCD of the first row elements of $\mathbf{B}_i$. We claim that $\mathbf{V}_{i_j}^T$ has the following form:

$$\mathbf{V}_{i_j}^T = \left[\begin{array}{cccc} v_{11} & v_{12} & \dots & v_{1t} \\ 0 & v_{11} & \dots & v_{2t} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & v_{t2} & \dots & v_{tt} \end{array}\right].$$

Otherwise, assume that for some $\alpha \in A_{i_j}$, there exists at least one $\sigma_\alpha(v_{l1}) \neq 0, 2 \leq l \leq t$. Obviously, $\sigma_\alpha(\mathbf{v}_1) = (\sigma_\alpha(v_{11}), \dots, \sigma_\alpha(v_{t1}))^T \succ_t (\sigma_\alpha(v_{11}), 0, \dots, 0)^T$ under the POT order $\succ_t$ being the restriction of $\succ_{t+1}$ with $\mathbf{e}_1 \succ \mathbf{e}_{t+1} \succ \cdots \succ \mathbf{e}_2$ on $k[x]^t$, which contradicts that $\sigma_\alpha(\mathbf{v}_1)$ should be minimal by Proposition 4.4.

Thus, $\mathbf{B}_{i_j} = \mathbf{B}_i\mathbf{V}_{i_j}^T$ satisfies that the first column and row are zeros except for the (1,1)-element on A_{i_j}. Consequently, Algorithm 3 terminates. □

We use a simple example to illustrate Algorithm 2.

Example 4.6. Given a matrix $B \in \mathbb{C}[a][x]^{3\times 3}$ and a constructible set $A = \mathbb{C}$ as follows:

$$\mathbf{B} = \left[\begin{array}{ccc} a-x & 2x & 0 \\ 0 & 0 & x \\ x^2+1 & x^3+a+x & -x^2 \end{array}\right].$$

Step 1: perform the routine Reduce2Zero(A, $\mathbf{B}$), that is, repeatedly call Algorithm 1* on the first column and row of the matrix, then we get the matrices in which the first column and row are zeros except for the (1,1)-component.

Table 2: Output of Reduce2Zero(A, $\mathbf{B}$)

No.	A_i	$\mathbf{B}_i$	$\mathbf{U}_i$	$\mathbf{V}_i$
1	$\mathbb{C}\backslash\mathbb{V}(a^2+1)$	$\mathbf{B}_1$	$\mathbf{U}_1$	$\mathbf{V}_1$
2	$\mathbb{V}(a^2+1)$	$\mathbf{B}_2$	$\mathbf{U}_2$	$\mathbf{V}_2$

where ($\mathbf{U}_i\mathbf{B}\mathbf{V}_i = \mathbf{B}_i$, $i = 1, 2$.)

$$\mathbf{B}_1 = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & x(a^2+1) & 0 \\ 0 & (a^2+1)(a-x)x^2 & b_{133} \end{array}\right], \quad \mathbf{B}_2 = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & x & 0 \\ 0 & -2x^2 & b_{233} \end{array}\right],$$

$$\mathbf{U}_1 = \left[\begin{array}{ccc} a+x & 0 & 1 \\ u_{121} & 1 & u_{123} \\ u_{131} & 0 & u_{133} \end{array}\right], \quad \mathbf{U}_2 = \left[\begin{array}{ccc} -1 & 0 & 0 \\ 0 & 1 & 0 \\ ax^3+2ax^2+ax+2a-1 & 0 & 2 \end{array}\right],$$

$$\mathbf{V}_1 = \left[\begin{array}{ccc} -4x^2+1 & x^2 & v_{113} \\ 2ax-a+x & 0 & a^2+1 \\ v_{131} & a^2+1 & 0 \end{array}\right], \quad \mathbf{V}_2 = \left[\begin{array}{ccc} a & 0 & 2x \\ a/2 & 0 & -a+x \\ 0 & 1 & 0 \end{array}\right],$$

$$\begin{aligned}
b_{133} &= -(a^2+1)(ax^3-x^4-2x^3+a^2-x^2-2x), \\
b_{233} &= -2(a-x)(x^3+2ax+2x^2+a+x), \\
u_{121} &= -x(a+x)(2ax^2+3ax+x^2+2a+2x-3), \\
u_{123} &= -2ax^3-3ax^2-x^3-2ax-2x^2+3x, \\
u_{131} &= (a^2+1)(-4ax^3-4x^4+2a^2x-2x^3-a^2+x^2+1), \\
u_{133} &= (a^2+1)(-4x^3+2ax-2x^2-a+x), \\
v_{113} &= -x^3-2ax-2x^2-a-x, \\
v_{131} &= 2ax^2+3ax+x^2+2a+2x-3.
\end{aligned}$$

Step 2: perform the subroutine Divisible(A_i, $\mathbf{B}_i$) to check if all elements in $\mathbf{B}_i$ are divisible by the (1,1)-element.

Obviously, $\mathbf{B}_1$ and $\mathbf{B}_2$ satisfy the divisibility relation between the (1,1)-element and other elements.

Step 3: repeat the Step 1 and Step 2 on the lower right submatrices of $\mathbf{B}_1$ and $\mathbf{B}_2$. We obtain the following, where $A'_1 \cup A'_2 = A_1$, $\mathbf{B}'_1$ and $\mathbf{B}'_2$ come from SubMatrix($\mathbf{B}_1$, 1).

Table 3: Output of SubMatrix($\mathbf{B}_1$, 1) **and** SubMatrix($\mathbf{B}_2$, 1)

No.	A'_i	$\mathbf{B}'_i$	$\mathbf{U}'_i$	$\mathbf{V}'_i$
1	$\mathbb{C}\backslash\mathbb{V}(a(a^2+1))$	$\mathbf{B}'_1$	$\mathbf{U}'_1$	$\mathbf{V}'_1$
2	$\mathbb{V}(a)\backslash\mathbb{V}(a^2+1)$	$\mathbf{B}'_2$	$\mathbf{U}'_2$	$\mathbf{V}'_2$
3	$\mathbb{V}(a^2+1)$	$\mathbf{B}'_3$	$\mathbf{U}'_3$	$\mathbf{V}'_3$

$$\mathbf{B}'_1 = \left[\begin{array}{cc} 1 & 0 \\ 0 & b'_{122} \end{array}\right], \quad \mathbf{B}'_2 = \left[\begin{array}{cc} x & 0 \\ 0 & b'_{222} \end{array}\right], \quad \mathbf{B}'_3 = \left[\begin{array}{cc} x & 0 \\ 0 & b'_{322} \end{array}\right],$$

$$\mathbf{U}'_1 = \left[\begin{array}{cc} u'_{111} & -1/(a^4+a^2) \\ u'_{121} & x/(a^4+a^2) \end{array}\right], \quad \mathbf{U}'_2 = \left[\begin{array}{cc} 1 & 0 \\ u'_{221} & 1/(a^4+a^2) \end{array}\right], \quad \mathbf{U}'_3 = \left[\begin{array}{cc} 1 & 0 \\ x & 1/2 \end{array}\right],$$

$$\mathbf{V}'_1 = \left[\begin{array}{cc} 1 & v'_{112} \\ 1 & v'_{122} \end{array}\right], \quad \mathbf{V}'_2 = \left[\begin{array}{cc} 1 & 0 \\ 1 & 1 \end{array}\right], \quad \mathbf{V}'_3 = \left[\begin{array}{cc} 1 & 0 \\ x & 1/2 \end{array}\right],$$

$$\begin{aligned}
b'_{122} &= -x(ax^3 - x^4 - 2x^3 + a^2 - x^2 - 2x),\\
b'_{222} &= -x(ax^2 - x^3 - 2x^2 - x - 2),\\
b'_{322} &=(x - a)(x^3 + 2ax + 2x^2 + a + x),\\
u'_{111} &=(-ax^2 + x^3 + ax + x^2 + x + 2)/(a^4 + a^2),\\
u'_{121} &=(ax^3 - x^4 - ax^2 - x^3 + a^2 - x^2 - 2x)/(a^4 + a^2),\\
u'_{221} &=(ax^2 - x^3 - ax - x^2 - x - 2)/(a^4 + a^2),\\
v'_{112} &= -ax^3 + x^4 + 2x^3 - a^2 + x^2 + 2x,\\
v'_{122} &= -ax^3 + x^4 + 2x^3 + x^2 + 2x.
\end{aligned}$$

Step 4: recover the Smith normal forms. Where

Table 4: recover Smith normal forms

No.	A''_i	$\mathbf{B}''_i$	$\mathbf{U}''_i$	$\mathbf{V}''_i$
1	$\mathbb{C}\backslash\mathbb{V}(a(a^2+1))$	$\mathbf{B}''_1$	$\mathbf{U}''_1$	$\mathbf{V}''_1$
2	$\mathbb{V}(a)\backslash\mathbb{V}(a^2+1)$	$\mathbf{B}''_2$	$\mathbf{U}''_2$	$\mathbf{V}''_2$
3	$\mathbb{V}(a^2+1)$	$\mathbf{B}''_3$	$\mathbf{U}''_3$	$\mathbf{V}''_3$

$$\mathbf{U}''_1 = \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{U}'_1 \end{bmatrix}\mathbf{U}_1, \quad \mathbf{U}''_2 = \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{U}'_2 \end{bmatrix}\mathbf{U}_1, \quad \mathbf{U}''_3 = \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{U}'_3 \end{bmatrix}\mathbf{U}_2,$$

$$\mathbf{V}''_1 = \mathbf{V}_1\begin{bmatrix} 1 & 0 \\ 0 & \mathbf{V}'_1 \end{bmatrix}, \quad \mathbf{V}''_2 = \mathbf{V}_1\begin{bmatrix} 1 & 0 \\ 0 & \mathbf{V}'_2 \end{bmatrix}, \quad \mathbf{V}''_3 = \mathbf{V}_2\begin{bmatrix} 1 & 0 \\ 0 & \mathbf{V}'_3 \end{bmatrix},$$

$$\mathbf{B}''_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & b''_{133} \end{bmatrix}, \quad \mathbf{B}''_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & b''_{233} \end{bmatrix}, \quad \mathbf{B}''_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & b''_{333} \end{bmatrix},$$

$$\begin{aligned}
b''_{133} &=x^5 + (-a + 2)x^4 + x^3 + 2x^2 - a^2x,\\
b''_{233} &=x^4 + 2x^3 + x^2 + 2x,\\
b''_{333} &= -2a^2x^2 - a^2x + x^3 + (-a + 2)x^4 + x^5.
\end{aligned}$$

5 CONCLUDING REMARKS

An algorithm for computing extended GCD systems of parametric univariate polynomials has been proposed. We can see that this algorithm simultaneously give the GCD and the representation coefficients by computing the CGS of a constructed module, which adds the unit vectors to record the representation coefficients (as mentioned in [5]). Meanwhile, this CGS for M also gives a set of free bases for the parametric syzygy module of input polynomials. It is worth noting that we get a stronger result: the unimodular matrix $\mathbf{U}$. Therefore, we can apply the proposed extended GCD algorithm to the computation of the Smith normal form and present the first algorithm for computing the Smith normal form of univariate polynomial matrices with parameters. In addition, the proposed algorithms have been implemented on the computer algebra system *Maple*, and the codes and examples are available on the web: http://www.mmrc.iss.ac.cn/~dwang/software.html.

ACKNOWLEDGMENTS

This research was supported in part by CAS Project QYZDJ-SSW-SYS022.

REFERENCES

[1] S.A. Abramov and K.Y. Kvashenko. 1993. On the Greatest Common Divisor of Polynomials which Depend on a Parameter. In *Proceedings of the 1993 ACM International Symposium on Symbolic and Algebraic Computation.* 152–156.

[2] A. Ayad. 2010. Complexity of algorithms for computing greatest common divisors of parametric univariate polynomials. *International Journal of Algebra* 4 (2010), 173–188.

[3] T. Bächler, V. Gerdt, M. Lange-Hegermann, and D. Robertz. 2012. Algorithmic Thomas decomposition of algebraic and differential systems. *Journal of Symbolic Computation* 47, 10 (2012), 1233–1266.

[4] S. Bamett. 1971. Matrices in control theory. *Van Norstrand Reinhold* (1971).

[5] B. Beckermann, G. Labahn, and G. Villard. 1999. Shifted normal forms of polynomial matrices. In *Proceedings of ISSAC' 1999.* 189–196.

[6] G. Bradley. 1971. Algorithms for Hermite and Smith normal matrices and linear diophantine equations. *Math. Comp.* 25, 116 (1971), 897–907.

[7] R. P. Brent and H. T. Kung. 1984. Systolic VLSI Arrays for Polynomial GCD Computation. *IEEE Trans. Comput.* 100, 8 (1984), 731–736.

[8] W.S. Brown. 1971. On Euclid's Algorithm And The Computation Of Polynomial Greatest Common Divisors. *J. ACM* 18, 4 (1971), 478–504.

[9] C. Chen and M. Maza. 2012. Algorithms for computing triangular decomposition of polynomial systems. *Journal of Symbolic Computation* 47, 6 (2012), 610–642.

[10] S.C. Chou. 1988. *Mechanical geometry theorem proving.* Vol. 41. Springer Science and Business Media.

[11] R.M. Corless, M.M. Maza, and S.E. Thornton. 2017. Jordan Canonical Form with Parameters from Frobenius Form with Parameters. In *International Conference on Mathematical Aspects of Computer and Information Sciences.* 179–194.

[12] D. Cox, J. Little, and D. O'shea. 2006. *Using algebraic geometry.* Vol. 185. Springer Science & Business Media.

[13] D. Cox, T. Sederberg, and F.L. Chen. 1998. The moving line ideal basis of planar rational curves. *Computer Aided Geometric Design* 15, 8 (1998), 803–827.

[14] F. R. Gantmakher. 1959. *The theory of matrices.* American Mathematical Soc.

[15] K. Geddes, S. Czapor, and G. Labahn. 1992. *Algorithms for computer algebra.* Springer Science and Business Media.

[16] P. Gianni and B. Trager. 1985. Gcd's and factoring multivariate polynomials using Grobner bases. In *European Conference on Computer Algebra.* Springer, 409–410.

[17] H. Kai and M.-T. Noda. 2000. Hybrid rational approximation and its applications. *Reliable Computing* 6 (2000), 429–438.

[18] M. Kalkbrener. 1997. On the Stability of Gröbner Bases Under Specializations. *Journal of Symbolic Computation* 24, 1 (1997), 51–58.

[19] D. Kapur, D. Lu, M. Monagan, Y. Sun, and D.K. Wang. 2018. An Efficient Algorithm for Computing Parametric Multivariate Polynomial GCD. In *Proceedings of the 2018 International Symposium on Symbolic and Algebraic Computation.* 239–246.

[20] D. Kapur, Y. Sun, and D.K. Wang. 2010. A new algorithm for computing comprehensive Gröbner systems. In *Proceedings of ISSAC' 2010.* 29–36.

[21] D. Kapur, Y. Sun, and D.K. Wang. 2013. An efficient algorithm for computing a comprehensive Gröbner system of a parametric polynomial system. *Journal of Symbolic Computation* 49 (2013), 27–44.

[22] A. Montes. 2002. A new algorithm for discussing Gröbner bases with parameters. *Journal of Symbolic Computation* 33, 2 (2002), 183–208.

[23] J. Moses and D. Yun. 1973. The ez gcd algorithm. In *Proceedings of the ACM annual conference.* ACM, 159–166.

[24] K. Nabeshima. 2007. PGB: a package for computing parametric Gröbner and related objects. *ACM Communications in Computer Algebra* 41, 3 (2007), 104–105.

[25] K. Nabeshima. 2007. A speed-up of the algorithm for computing comprehensive Gröbner systems. In *Proceedings of ISSAC' 2007.* 299–306.

[26] K. Nabeshima. 2010. On the computation of parametric gröbner bases for modules and syzygies. *Japan Journal of Industrial and Applied Mathematics* 27, 2 (2010), 217–238.

[27] K. Nagasaka. 2017. Parametric Greatest Common Divisors using Comprehensive Gröbner Systems. In *Proceedings of ISSAC' 2017.* 341–348.

[28] C. Norman. 2012. Finitely Generated Abelian Groups and Similarity of Matrices over a Field. *Springer Undergraduate Mathematics* (2012).

[29] I.S. Pace and S. Barnett. 1974. Efficient algorithms for linear system calculations. I: Smith form and common divisor of polynomial matrices. *Internat.j.systems Sci* (1974), 403–411.

[30] H.H. Rosenbrock. 1970. State-space and multivariable theory. (1970).

[31] T. Sasaki and M. Suzuki. 1992. Three new algorithms for multivariate polynomial GCD. *Journal of Symbolic Computation* 13, 4 (1992), 395–411.

[32] J. Sendra and J. Llovet. 1992. An extended polynomial GCD algorithm using Hankel matrices. *Journal of symbolic computation* 13, 1 (1992), 25–39.

[33] A. Storjohann. 1997. A solution to the extended GCD problem with applications. In *Proceedings of the 1997 international symposium on Symbolic and algebraic computation.* 109–116.

[34] A. Suzuki and Y. Sato. 2002. An alternative approach to comprehensive Gröbner bases. *Journal of Symbolic Computation* 36, 3 (2002), 649–667.

[35] A. Suzuki and Y. Sato. 2006. A simple algorithm to compute comprehensive Gröbner bases using Gröbner bases. In *Proceedings of the 2006 ACM International Symposium on Symbolic and Algebraic Computation.* 326–331.

[36] V. Weispfenning. 1992. Comprehensive Gröbner bases. *Journal of Symbolic Computation* 14, 1 (1992), 1–29.

[37] R. Zippel. 1979. Probabilistic algorithms for sparse polynomials. In *Proceedings of the EUROSAM'79.* Springer-Verlag, 216–226.

[38] R. Zippel. 1993. *Effective Polynomial Computation.* Vol. 241. Springer Science and Business Media.