J Syst Sci Complex (2019) 32: 234–255

A Survey on Algorithms for Computing Comprehensive Gröbner Systems and Comprehensive Gröbner Bases*

LU Dong · SUN Yao · WANG Dingkang

DOI: 10.1007/s11424-019-8357-z
Received: 12 October 2018 / Revised: 4 December 2018

Abstract Weispfenning in 1992 introduced the concepts of comprehensive Gröbner system/basis of a parametric polynomial system, and he also presented an algorithm to compute them. Since then, this research field has attracted much attention over the past several decades, and many efficient algorithms have been proposed. Moreover, these algorithms have been applied to many different fields, such as parametric polynomial equations solving, geometric theorem proving and discovering, quantifier elimination, and so on. This survey brings together the works published between 1992 and 2018, and we hope that this survey is valuable for this research area.

Keywords Comprehensive Gröbner basis, comprehensive Gröbner system, discovering geometric theorems mechanically, parametric polynomial system, quantifier elimination.

1 Introduction

In the past, the problems related to parametric polynomial system have been extensively studied. For instance, people often need to find solutions of a parametric polynomial system in many engineering fields[1–7]; geometry theorem proving and discovering[8–13] need to find some conditions such that a geometric statement becomes true or true on components; the Perspective-n-Point (PnP) problem[14, 15] wants to determine the position of the camera with

LU Dong
KLMM, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190*, China; School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing* 100049*, China.*
Email: donglu@amss.ac.cn.
SUN Yao
SKLOIS, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093*, China.*
Email: sunyao@iie.ac.cn.
WANG Dingkang
KLMM, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190*, China; School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing* 100049*, China.*
Email: dwang@mmrc.iss.ac.cn.
*This research was supported in part by the CAS Project QYZDJ-SSW-SYS022, the National Natural Science Foundation of China under Grant No. 61877058, and the Strategy Cooperation Project AQ-1701.
◇ *This paper was recommended for publication by Editor LI Hongbo.*

respect to a scene object from n corresponding points; the goal of quantifier elimination[16–21] is to eliminate quantifiers "$\exists$" and "$\forall$" in a given algebraic first order formula over $\mathbb{R}$ or $\mathbb{C}$. Because of these wide range of applications, the theory related to parametric polynomial systems has been widely developed.

In 1992, Weispfenning[22] introduced the concept of comprehensive Gröbner basis, as a special basis of a parametric polynomial system. That is, for any given parametric ideal $I \subset k[U][X]$, a comprehensive Gröbner basis G of I is a subset in $k[U][X]$ such that for every specialization of parameters $\sigma_\alpha : k[U] \to L$ extending to $k[U][X] \to L[X]$, the set $\sigma_\alpha(G)$ is a Gröbner basis of the specialized ideal $\langle \sigma_\alpha(I) \rangle$ in $L[X]$, where L is an algebraic closed filed containing k, $U = \{u_1, \cdots, u_m\}$ are parameters, $X = \{x_1, \cdots, x_n\}$ are variables, and $\alpha = (\alpha_1, \cdots, \alpha_m) \in L^m$. Weispfenning also introduced the concept of comprehensive Gröbner system. In addition, he proposed algorithms for computing comprehensive Gröbner systems and comprehensive Gröbner bases. In 1994, Pesh[23] implemented these algorithms on the computer algebra system MAS.

In 1995, Kapur[24] proposed the parametric Gröbner basis independently. In this paper, he introduced the concept of constrained polynomials. A constrained polynomial is defined as a pair $\langle H, f \rangle$, where $f \in k[U][X]$ and H is a finite set of constraints over parameters. For any given $\alpha \in L^m$, constraints return an explicit true or false value. Kapur proposed two methods: Comprehensive Gröbner basis method and parametric characteristic sets method, to solve parametric polynomial systems based on constrained polynomials. In his methods, he need to assume that the leading coefficient of a constrained polynomial is nonzero. When the leading coefficient of a constrained polynomial cannot be judged to be nonzero, this constrained polynomial is called ambiguous; otherwise, non-ambiguous. In the process of calculation, we can change ambiguous polynomials into non-ambiguous polynomials by adding some constraints. Chen, et al.[8] gave an algorithm to compute parametric Gröbner bases according to Kapur's idea.

In 2002, Montes[25] proposed an algorithm to compute comprehensive Gröbner systems. The main purpose of this paper is to get all different Gröbner bases corresponding to all possible parameter values, and obtained an algorithm which is called the DISPGB algorithm. The DISPGB algorithm produces disjoint segments for the whole parameter space, defined by some polynomial equations and inequations, and the corresponding Gröbner bases. For every specialization in the same segment, the leading monomials remain unchanged.

Inspired by Montes's approach, Weispfenning rethinked a question that had plagued him for ten years: For a parameter ideal I in $k[U][X]$, is there a purely structural sense of the canonical comprehensive Gröbner basis? That is, does this canonical comprehensive Gröbner basis always exist and is it uniquely determined by the ideal I and the term order $\prec$? In [26, 27], Weispfenning tried to solve these problems constructively. He defined two types of regular (canonical) Gröbner systems: A faithful Gröbner system based on [22], which can induce a canonical comprehensive Gröbner basis; and a non-faithful Gröbner system based on [25]. In the classical case, Weispfenning constructed a faithful Gröbner system, and he tried to give a canonical comprehensive Gröbner basis determined by the associated ideal I and term order $\prec$.

On the other hand, under the influence of Weispfenning's new idea[26, 27], Manubens and Montes[28] made important improvements to the DISPGB algorithm. They redesigned the flow of DISPGB algorithm, and obtained a new algorithm which is called the DPGB algorithm. By discussing a compact tree, the DPGB algorithm can avoid many unnecessary branches and make the output partitions easier. The new algorithm is very efficient and nearly 20 times faster than the original algorithm.

Since then, there are many papers such as Suzuki and Sato[29, 30], Wibmer[31], Manubens and Montes[32], Montes and Wibmer[33], that made some improvements to improve the algorithms for computing a comprehensive Gröbner basis and system of a parametric polynomial ideal. In 2006, Suzuki and Sato[34] made a major breakthrough. Based on Kalkbrener's work[35], they proposed a new algorithm (called the SS algorithm in our paper) to compute a comprehensive Gröbner system. Moreover, they modified the SS algorithm to compute a comprehensive Gröbner basis of a parametric polynomial ideal. The two most important properties of their algorithms are: 1) The segments of parameter space may not be disjoint; 2) It is to compute a Gröbner basis of a parametric ideal in the polynomial ring $k[U][X]$ rather than in the polynomial ring $k(U)[X]$. The two properties can help them do not need to deal with inequations ("not equal to zero") and make the algorithms faster than previous algorithms. However, the first property causes the algorithms to generate many redundant segments. It will take a lot of time to check whether a segment is redundant. Therefore, the computation may be heavy.

In 2007, Nabeshima[36] improved the SS algorithm and presented a speed-up algorithm for computing comprehensive Gröbner systems, which generates fewer cells of parameter space than the SS algorithm by using inequations ("not equal zero"). Moreover, he used the Rabinovitch's trick to check redundant segments. These two steps make the speed-up algorithm superior to the SS algorithm in practice.

Kapur, et al.[37, 38] in 2010 made the most important improvements to the SS algorithm by using minimal Dickson basis to remove redundant segments. For any given set $G \subset k[U][X]$, the minimal Dickson basis G_m of G satisfies that the ideal generated by the leading monomials of G_m is equal to the ideal generated by the leading monomials of G, and neither $\mathrm{lm}_X(f_1) \mid \mathrm{lm}_X(f_2)$ nor $\mathrm{lm}_X(f_2) \mid \mathrm{lm}_X(f_1)$ for any two distinct $f_1, f_2 \in G_m$. Compared with the Suzuki and Sato's method, the minimal Dickson basis is used to avoid unnecessary branches, and hence, to reduce the computations of Gröbner basis. The other important improvement in references [37, 38] is to propose many tricks and heuristics for checking whether a parametric constraint is empty. For a parametric constraint (E, N), where E, N are ideals in $k[U]$, the algebraically constructible set $\boldsymbol{V}(E) \setminus \boldsymbol{V}(N)$ is empty if and only if for every $f \in N$, we have that $f \in \sqrt{E}$. Nabeshima[36] used the Rabinovitch's trick to check whether $f \in \sqrt{E}$. However, this trick needs to introduce an auxiliary variable and this step can be very expensive, since the complexity of Gröbner basis computations is heavily influenced by the number of variables. The tricks and heuristics, proposed by Kapur, et al. such as checking whether $f \in E$, are more efficient than Nabeshima's method. According to the two improvements, Kapur, et al. proposed a more efficient algorithm (called the KSW algorithm in our paper) that generates fewer segments than existing algorithms.

In 2011, Kapur, et al.[39, 40] modified the KSW algorithm to compute a faithful comprehensive Gröbner basis of a parametric polynomial system. The main idea of their method is to keep track of the nonzero part and zero part of a parametric polynomial for the specialization by using the two components of a tuple in $(k[U][X])^2$. The first component can form a comprehensive Gröbner system. At the same time, the nonzero part plus the zero part of all the tuples generate a comprehensive Gröbner basis. This algorithm has been found to be more efficient in practice.

Based on the results in [41, 42], Kapur[43] in 2017 gave a completion algorithm for computing a minimal faithful comprehensive Gröbner basis directly. This algorithm is similar to the Buchberger's algorithm. Computing the S-polynomial of two distinct parametric polynomials and parameterized rewriting are two important steps in the algorithm. There are some shortcomings in the algorithm, such as the computations of rewriting and redundancy check which could be expensive. This requires further work. In the same year, Hashemi, et al.[44] considered the problem of converting parametric Gröbner bases. Based on the generic Gröbner walk algorithm proposed by Fukuda, et al.[45], they presented an efficient algorithm to convert a comprehensive Gröbner system w.r.t. a given monomial ordering into a Gröbner system w.r.t. another monomial ordering. In 2018, Hashemi, et al.[46] introduced the concept of universal Gröbner basis for a parametric ideal. Combining the Gröbner basis conversion[44] and the comprehensive Gröbner bases algorithm in [39, 40], they could compute a universal Gröbner basis of a parametric ideal.

This paper is structured as follows: In Section 2, we introduce some notations and the definition of comprehensive Gröbner basis and system. Section 3 contains some major theorems and efficient algorithms for computing comprehensive Gröbner bases and systems. In Section 4, we introduce some applications of comprehensive Gröbner systems. Finally, Section 5 includes some conclusions.

2 Preliminaries

This section contains some basic notations and definitions for parametric polynomial, specialization, comprehensive Gröbner system, comprehensive Gröbner basis, and so on.

Let k be a field, L be an algebraic closed field containing k, $k[U]$ be the parameter ring in the parameters $U = \{u_1, \cdots, u_m\}$, and $k[U][X]$ be the polynomial ring in the variables $X = \{x_1, \cdots, x_n\}$ with coefficients in $k[U]$. It is assumed that U and X are disjoint sets.

We first introduce some notations for parametric multivariate polynomials. For a polynomial $f \in k[U][X]$, the leading term, leading coefficient, and leading monomial of f w.r.t. a monomial order $\prec_X$ are denoted by $\mathrm{lt}_X(f)$, $\mathrm{lc}_X(f)$, and $\mathrm{lm}_X(f)$ respectively. We have $\mathrm{lt}_X(f) = \mathrm{lc}_X(f) \cdot \mathrm{lm}_X(f)$. For example, let $f = 2u_1u_2x_1^3 + u_1x_1^2x_2 - u_2^2x_2$ be a parametric polynomial in $k[u_1, u_2][x_1, x_2]$, where $\prec_X$ is the lexicographic order with $x_2 \prec x_1$, then $\mathrm{lt}_X(f) = 2u_1u_2x_1^3$, $\mathrm{lc}_X(f) = 2u_1u_2$, and $\mathrm{lm}_X(f) = x_1^3$.

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

specialization σ_α is defined as

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

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

For a set $E \subset k[U]$, the variety defined by E in L^m is denoted by $\boldsymbol{V}(E) = \{\alpha \in L^m \mid f(\alpha) = 0$ for all $f \in E\}$. In this paper, an **algebraically constructible set** A is defined as follows: $A = \boldsymbol{V}(E) \setminus \boldsymbol{V}(N)$, where E, N are subsets of $k[U]$. It is easy to see that the algebraically constructible set A is not empty by ensuring that at least one $f \in N$ is not in the radical of $\langle E\rangle$.

For a parametric polynomial system, the definition of comprehensive Gröbner system is given below.

Definition 2.1 (CGS) Let F be a set of $k[U][X]$, $A_1, \cdots, A_l$ be algebraically constructible subsets of L^m, $G_1, \cdots, G_l$ be subsets of $k[U][X]$, and S be a subset of L^m such that $S \subset A_1 \cup \cdots \cup A_l$. A finite set $\mathcal{G} = \{(A_1, G_1), \cdots, (A_l, G_l)\}$ is called a **comprehensive Gröbner system** (CGS) on S for F if $\sigma_\alpha(G_i)$ is a Gröbner basis for the ideal $\langle \sigma_\alpha(F)\rangle \subset L[X]$ for $\alpha \in A_i$ and $i = 1, 2, \cdots, l$. 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.

Example 2.2 Let F be an ideal in $\mathbb{C}[a, b][x]$ generated by $f_1 = (1 - a)x + b$, where $\mathbb{C}$ is the complex field. x is the variable and a, b are the parameters. $\prec_{a,b}$ is the lexicographic order with $a > b$. Let $S = \mathbb{C}^2$, then a CGS $\mathcal{G}$ on $\mathbb{C}^2$ for F is: $\{(\mathbb{C}^2 \setminus \boldsymbol{V}(1-a), \{(1-a)x+b\}), (\boldsymbol{V}(1-a) \setminus \boldsymbol{V}(b), \{1\}), (\boldsymbol{V}(1-a, b), \{0\})\}$.

In most cases, we need to compute a minimal CGS of a parametric polynomial system. Hence, a minimal CGS is defined as follows.

Definition 2.3 A CGS $\mathcal{G} = \{(A_1, G_1), \cdots, (A_l, G_l)\}$ on S for F is said to be **minimal**, if for every $i = 1, 2, \cdots, l$,

1) $A_i \neq \emptyset$, furthermore, $\cup_{i=1}^{l} A_i = S$ and $A_i \cap A_j = \emptyset$ whenever $i \neq j$;

2) $\sigma_\alpha(G_i)$ is a minimal Gröbner basis for $\langle \sigma_\alpha(F)\rangle \subset L[X]$ for $\alpha \in A_i$;

3) for each $g \in G_i$, $\sigma_\alpha(\mathrm{lc}_X(g)) \neq 0$ for any $\alpha \in A_i$.

In Example 2.2, $\mathcal{G}$ is a minimal CGS on $\mathbb{C}^2$ for F.

Definition 2.4 (CGB) Let F be a subset of $k[U][X]$, and S be a subset of L^m. A finite subset G in $k[U][X]$ is called a **comprehensive Gröbner basis** (CGB) on S for F, if $\sigma_\alpha(G)$ is a Gröbner basis of the ideal $\langle \sigma_\alpha(F)\rangle \subset L[X]$ for each $\alpha \in S$. If $S = L^m$, then G is called a comprehensive Gröbner basis for F. A comprehensive Gröbner basis G of F is called **faithful** if in addition, every element of G is also in $\langle F\rangle$.

Example 2.5 (see [40]) Let $F = \{ax_1 - b, bx_2 - a, cx_1^2 - x_2, cx_2^2 - x_1\} \subset \mathbb{C}[a, b, c][x_1, x_2]$, where x_1, x_2 are variables and a, b, c are parameters. For block order $\prec_{X,U}$, $\{a, b, c\} \ll \{x_1, x_2\}$; within each block, $\prec_X$ and $\prec_U$ are graded reverse lexicographic orders with $x_2 \prec x_1$ and

$c < b < a$, respectively. The CGB on $\mathbb{C}^3$ for F is $\{a^6 - b^6, a^3c - b^3, b^3c - a^3, ac^2 - a, bc^2 - b, bx_1 - acx_2, bx_2 - a, cx_1^2 - x_2, cx_2^2 - x_1\}$.

Remark 2.6 For any given parametric polynomial system $F \subset k[U][X]$, and $\{(A_1, G_1), \cdots, (A_l, G_l)\}$ be a CGS on S for F. The set $\bigcup_{i=1}^{l} G_i$ may not be a CGB of F. For example, $F = \{ax + 1\} \subset \mathbb{C}[a][x]$ where x is the variable and a is the parameter. The CGS $\mathcal{G}$ of F on $\mathbb{C}$ is: $\{(\mathbb{C} \setminus \boldsymbol{V}(a), \{ax + 1\}), (\boldsymbol{V}(a), \{1\})\}$. Then $G_1 \cup G_2 = \{ax + 1, 1\}$ and clearly it is not the CGB of F. In general, it is more difficult to compute a CGB of F than to compute a CGS of F.

3 Algorithms for Computing CGS and CGB

In this section, we will mainly introduce the algorithms proposed by Suzuki and Sato[34], and Kapur, et al.[37–40] for computing CGS and CGB. This is because they are not only one of the most efficient algorithms so far, but also implemented in many computer algebra systems such as Risa/Asir, Maple, Singular, and so on. Moreover, Suzuki and Sato used the SS algorithm to solve quantifier elimination, Kapur, et al. proved and discovered reducible geometric theorems by using the KSW algorithm.

First, we study the stability of Gröbner bases under specializations. Let I be an ideal in $k[U][X]$, and σ be a ring homomorphism from $k[U]$ to L. When does a Gröbner basis G of I map to a Gröbner basis $\widetilde{G}$ of the ideal $\langle \sigma(I) \rangle$ in $L[X]$? That is, $\widetilde{G} = \sigma(G)$. In 1997, Kalkbrener[35] considered the above problem, and obtained the following theorem.

Theorem 3.1 (see [35]) *Let σ be a ring homomorphism from $k[U]$ to L, I be an ideal in $k[U][X]$ and $G = \{g_1, \cdots, g_s\}$ a Gröbner basis of I with respect to an admissible order $\prec_X$. We assume that the g_is are ordered in such a way that there exists an $r \in \{0, \cdots, s\}$ with $\sigma(\mathrm{lc}_X(g_i)) \neq 0$ for $i \in \{1, \cdots, r\}$ and $\sigma(\mathrm{lc}_X(g_i)) = 0$ for $i \in \{r + 1, \cdots, s\}$. Then the following two conditions are equivalent.*

(a) *$\{\sigma(g_1), \cdots, \sigma(g_r)\}$ is a Gröbner basis of $\langle \sigma(I) \rangle$ w.r.t. $\prec_X$;*

(b) *For every $i \in \{r+1, \cdots, s\}$ the polynomial $\sigma(g_i)$ is reducible to 0 modulo $\{\sigma(g_1), \cdots, \sigma(g_r)\}$.*

According to Theorem 3.1, Suzuki and Sato[34] proposed a lemma which plays an important role in the SS algorithm.

Lemma 3.2 (see [34]) *Let G be a Gröbner basis of the ideal $\langle F \rangle \subset k[U][X]$ w.r.t. an order $\prec_{X,U}$. For any $\alpha \in L^m$, let $G_1 = \{g \in G \mid \sigma_\alpha(\mathrm{lc}_X(g)) \neq 0\}$. Then $\sigma_\alpha(G_1) = \{\sigma_\alpha(g) \mid g \in G_1\}$ is a Gröbner basis of $\langle \sigma_\alpha(F) \rangle$ in $L[X]$ w.r.t. $\prec_X$ if and only if $\sigma_\alpha(g)$ reduces to 0 modulo $\sigma_\alpha(G_1)$ for every $g \in G$.*

The following lemma is the direct consequence of Lemma 3.2.

Lemma 3.3 (see [34]) *Let G be a Gröbner basis of the ideal $\langle F \rangle \subset k[U, X] (= k[U][X])$ w.r.t. a block order $U \ll X$. If $\sigma_\alpha(\mathrm{lc}_X(g)) \neq 0$ for each $g \in G \setminus (G \cap k[U])$, then $\sigma_\alpha(G)$ is a Gröbner basis of $\langle \sigma_\alpha(F) \rangle$ in $L[X]$ w.r.t. $\prec_X$ for any $\alpha \in \boldsymbol{V}(G \cap k[U])$.*

With the help of Lemma 3.3, the main idea of Suzuki and Sato's algorithm is as follows: We first compute a reduced Gröbner basis G of $\langle F\rangle$ in $k[U][X]$ w.r.t. an order $\prec_{X,U}$. Let $\{h_1,\cdots,h_l\}=\{\mathrm{lc}_X(g)\mid g\in(G\setminus k[U])\}$, then $(\boldsymbol{V}(G\cap k[U])\setminus(\boldsymbol{V}(h_1)\cup\cdots\cup\boldsymbol{V}(h_l)),G)$ forms a segment of a CGS for F. Second, we need to get Gröbner bases from the parametric space $\boldsymbol{V}(h_1)\cup\cdots\cup\boldsymbol{V}(h_l)$. For each $\boldsymbol{V}(h_i)$, we use Lemma 3.3 to compute a reduced Gröbner basis of $\langle F\cup\{h_i\}\rangle$ in $k[U][X]$ w.r.t. an order $\prec_{X,U}$. Similarly, we can get a new segment of the CGS for F. Repeat the above process and we can obtain a CGS for F. Since, $h_i\notin\langle F\rangle$, the algorithm terminates in finitely many steps.

According to the above idea, Suzuki and Sato designed an efficient algorithm, SS algorithm for short, to compute CGS.

Algorithm SS-CGS($\boldsymbol{F}$)
Input F, a finite subset of $k[U][X]$.
Output $\mathcal{G}$, a finite set of triples which forms a CGS for F.
begin
$\mathcal{H}:=\mathrm{CGSMain}(F)$;
$G_0:=\mathrm{ReducedGröbnerBasis}(F,\prec_{X,U})$;
if $G_0\cap k[U]=\emptyset$ **then** $\mathcal{G}:=\emptyset$;
  else $\mathcal{G}:=\{(\emptyset,G_0\cap k[U],\{1\})\}$;
end if;
for each $(h,G)\in\mathcal{H}$ **do**
  $\mathcal{G}:=\mathcal{G}\cup\{(G\cap k[U],\{h\},G\setminus k[U])\}$;
end for;
return $\mathcal{G}$;
end.

In the above algorithm, CGSMain(F) is a subroutine whose details are as follows, and the subroutine ReducedGröbnerBasis($F,\prec_{X,U}$) outputs a reduced Gröbner basis of F w.r.t. $\prec_{X,U}$.

Algorithm CGSMain($\boldsymbol{F}$)
Input F, a finite subset of $k[U][X]$.
Output a finite set $\mathcal{H}$ of pairs (h,G) of a polynomial and a Gröbner basis in $k[U][X]$.
1) $G:=\mathrm{ReducedGröbnerBasis}(F,\prec_{X,U})$.
2) **If** $1\in G$, then $\mathcal{H}:=\{(1,F)\}$;
  else $\{h_1,\cdots,h_l\}:=\{\mathrm{lc}_X(g)\mid g\in(G\setminus k[U])\}$, $h:=\mathrm{lcm}\{h_1,\cdots,h_l\}$,
    $\mathcal{H}:=\{(h,G)\}\cup\mathrm{CGSMain}(G\cup\{h_1\})\cup\cdots\cup\mathrm{CGSMain}(G\cup\{h_l\})$;
3) Return $\mathcal{H}$.

Compared with the previous algorithms, the SS algorithm only requires the computations of Gröbner bases in $k[U,X]$ (not in $k(U)[X]$), which improves the computational efficiency. Since the segments of the parameter space may not be disjoint, there are many ways to optimize the SS algorithm (see [34] for more details).

In order to get a Gröbner basis for an ideal I in $k[U][X]$ w.r.t. an order $\prec_X$, Suzuki and Sato compute a Gröbner basis G of I in $k[U,X]$ with a block order $U\ll X$, which is compatible

with $\prec_X$, and then G should be a Gröbner basis for I in $k[U][X]$ w.r.t. $\prec_X$. Kurata[47] pointed out a subset of G can be still a Gröbner basis for the ideal I w.r.t. $\prec_X$, this fact can be used to improve Suzuki and Sato's algorithm.

Definition 3.4 Let F be a set of $k[U][X]$, and S be a subset of L^m. A CGS $\mathcal{G} = \{(A_1, G_1), \cdots, (A_l, G_l)\}$ on S for F is called a **faithful** CGS, if $G_i \subseteq \langle F\rangle$ for each $i = 1, 2, \cdots, l$.

The key step of getting a faithful CGS is to introduce a new variable y, where $y \gg X \gg U$. In order to introduce the following two important lemmas, we first give some new notations and definitions. For any given polynomial $g \in k[y, X, U]$, $\mathrm{lc}_{y,X}(g)$ denotes the leading coefficient of g w.r.t. $\prec_{y,X}$ as a polynomial of $k[U][y, X]$; $\mathrm{lm}_{y,X,U}(g)$ denotes the leading monomial of g w.r.t. $\prec_{y,X,U}$ as a polynomial of $k[y, X, U]$; $\mathrm{lc}_y(g)$ denotes the leading coefficient of g w.r.t. $\prec_y$ as a polynomial of $k[X, U][y]$. Moreover, let $M(U, X)$ be the set of monomials of $U \cup X$. We define homomorphisms σ^0 and σ^1 from $k[y, X, U]$ to $k[X, U]$ as a specialization of y with 0 and 1 respectively. That is, for any polynomial $g \in k[y, X, U]$, we have that $\sigma^0(g) = g(0, X, U)$ and $\sigma^1(g) = g(1, X, U)$. For a set $H \subset k[y, X, U]$, $g \cdot H$ denotes the set $\{g \cdot h \mid h \in H\}$.

Lemma 3.5 (see [34]) *Let F and E be sets of $k[X, U]$. For any $g \in \langle (y \cdot F) \cup ((y-1) \cdot E)\rangle_{k[y,X,U]}$, $\sigma^0(g) \in \langle E\rangle_{k[X,U]}$ and $\sigma^1(g) \in \langle F\rangle_{k[X,U]}$.*

Lemma 3.6 (see [34]) *Let F be a finite subset of $k[X, U]$, E be a finite subset of $k[U]$ such that $\boldsymbol{V}(E) \subseteq \boldsymbol{V}(\langle F\rangle \cap k[U])$, and G be the reduced Gröbner basis of the ideal $\langle (y \cdot F) \cup ((y-1) \cdot E)\rangle$ in $k[y, X, U]$ with respect to $\prec_{y,X,U}$. If $\{h_1, \cdots, h_l\} = \{\mathrm{lc}_{y,X}(g) \mid g \in G'\} \subseteq k[U]$, then $\sigma_\alpha(\sigma^1(G))$ is a Gröbner basis of $\langle \sigma_\alpha(F)\rangle$ in $L[X]$ for each $\alpha \in \boldsymbol{V}(E)$ such that $h_1(\alpha) \neq 0, \cdots, h_l(\alpha) \neq 0$, where $G' = \{g \in G \mid \mathrm{lm}_{y,X,U}(g) \notin M(U, X), \mathrm{lc}_y(g) \notin k[U]\}$.*

In Lemma 3.6, $\mathrm{lm}_{y,X,U}(g) \notin M(U, X)$ implies that $\mathrm{lm}_{y,X,U}(g)$ includes the variable y and $\mathrm{lc}_y(g) \notin k[U]$ is equivalent to that $\mathrm{lc}_y(g)$ includes at least one variable of X. Therefore, if the set G' is not empty, then the set $\{\mathrm{lc}_{y,X}(g) \mid g \in G'\}$ is not empty. According to Lemmas 3.5 and 3.6, Suzuki and Sato obtained an algorithm called the algorithm CGBMain(F, E) in [34] which outputs a faithful CGS on $\boldsymbol{V}(E)$ for F.

Algorithm CGBMain$(\boldsymbol{F}, \boldsymbol{E})$
Input F, a finite subset of $k[U][X]$; E, a finite subset of $k[U]$ such that $\boldsymbol{V}(E) \subseteq \boldsymbol{V}(\langle F\rangle \cap k[U])$.
Output $\mathcal{G}$, a finite set of triples of polynomials which forms a faithful CGS on $\boldsymbol{V}(E)$ for F.
begin
if $1 \in \langle E\rangle$ **then** $\mathcal{G} := \emptyset$;
else
$G :=$ ReducedGröbnerBasis$(y \cdot F \cup (y-1) \cdot E, \prec_{y,X,U})$;
$\{h_1, \cdots, h_l\} := \{\mathrm{lc}_{y,X}(g) \mid g \in G, \mathrm{lm}_{y,X,U}(g) \notin M(U, X), \mathrm{lc}_y(g) \notin k[U]\}$;
$h := \mathrm{lcm}\{h_1, \cdots, h_l\}$,
$\mathcal{G} := \{(E, \{h\}, \sigma^1(G))\} \cup$ CGBMain$(F, E \cup \{h_1\}) \cup \cdots \cup$ CGBMain$(F, E \cup \{h_l\})$;
end if;
return $\mathcal{G}$;
end.

Combining with the above algorithm, Suzuki and Sato obtained an algorithm for computing

a CGB of F. In the following algorithm, the subroutine Elim(F) computes a Gröbner basis of the elimination ideal $\langle F\rangle \cap k[U]$. Assume that the set G_0 is a Gröbner basis of the elimination ideal $\langle F\rangle \cap k[U]$ and $\{(A_1, G_1), \cdots, (A_l, G_l)\}$ is a faithful CGS on $\boldsymbol{V}(G_0)$ for F, then $\bigcup_{i=0}^{l} G_i$ form a CGB for F.

Algorithm SS-CGB($\boldsymbol{F}$)
Input F, a finite subset of $k[U][X]$.
Output G, a CGB for F.
begin
$E := \text{Elim}(F)$; $G := E$; $\mathcal{G} := \text{CGBMain}(F, E)$;
for each $(E', T', G') \in \mathcal{G}$ **do**
  $G := G \cup G'$;
end for;
return G;
end.

The results Suzuki and Sato achieved were a major breakthrough in the computation of CGS and CGB. However, the two algorithms suffers from some weaknesses. For example, the two algorithms allow the segments of parameter space not to be pairwise disjoint, which leads to produce redundant segments. In this case, the computation cost for checking redundant segments and consistency of parametric constrains may be very high. In 2010, Kapur, et al.[37] presented a new algorithm for computing a CGS of a parametric polynomial system which can avoid unnecessary branches in the SS algorithm. The algorithm proposed by Kapur, et al. is based on the following definition and theorem.

Definition 3.7 (Minimal Dickson Basis, see [37]) Given a set G of polynomials which is a subset of $k[U][X]$ and an admissible block order with $U \ll X$, we say $F \subset k[U][X]$, denoted as MDBasis(G), is a **Minimal Dickson Basis** of G, if

1) F is a subset of G,

2) for every polynomial $g \in G$, there is some polynomial $f \in F$ such that $\text{lm}_X(g)$ is a multiple of $\text{lm}_X(f)$, i.e., $\langle \text{lm}_X(F)\rangle = \langle \text{lm}_X(G)\rangle$, and

3) for any two distinct $f_1, f_2 \in F$, neither $\text{lm}_X(f_1)$ is a multiple of $\text{lm}_X(f_2)$ nor $\text{lm}_X(f_2)$ is a multiple of $\text{lm}_X(f_1)$.

Example 3.8 (see [37]) For any given set $G \subset k[U][X]$, we show that MDBasis(G) may not be unique. Let $G = \{ax_1^2 - x_2, ax_2^2 - 1, ax_1 - 1, (a+1)x_1 - x_2, (a+1)x_2 - a\} \subset \mathbb{C}[a][x_1, x_2]$, and $\prec_X$ be the lexicographic order with $a \ll x_2 < x_1$. Then $F = \{ax_1 - 1, (a+1)x_2 - a\}$ and $F' = \{(a+1)x_1 - x_2, (a+1)x_2 - a\}$ are both MDBasis(G). It is easy to verify $\langle \text{lm}_X(F)\rangle = \langle \text{lm}_X(F')\rangle = \langle \text{lm}_X(G)\rangle = \langle x_1, x_2\rangle$.

Theorem 3.9 (see [37]) *Let G be a Gröbner basis of the ideal $\langle F\rangle \subset k[U][X]$ w.r.t. an admissible block order with $U \ll X$. Let $G_r = G \cap k[U]$ and $G_m = \text{MDBasis}(G \setminus G_r)$. If σ is a specialization from $k[U]$ to L such that*

1) $\sigma(g) = 0$ *for* $g \in G_r$, *and*

2) $\sigma(h) \neq 0$, *where* $h = \Pi_{g \in G_m} \mathrm{lc}_X(g) \in k[U]$,

then $\sigma(G_m)$ *is a (minimal) Gröbner basis of* $\langle \sigma(F) \rangle$ *in* $L[X]$ *w.r.t.* $\prec_X$.

According to Theorem 3.9, Kapur, et al. proposed a more efficient algorithm which is called KSW algorithm in [37] to compute a CGS for a parametric polynomial system F.

Algorithm KSW-CGS($\boldsymbol{E}, \boldsymbol{N}, \boldsymbol{F}$)

Input (E, N, F): E and F are subsets of $k[U]$; F is a finite subset of $k[U][X]$.

Output a finite set of 3-tuples (E_i, N_i, G_i) such that $\{(\boldsymbol{V}(E_i) \setminus \boldsymbol{V}(N_i), G_i)\}$ constitutes a **minimal** CGS of F on $\boldsymbol{V}(E) \setminus \boldsymbol{V}(N)$.

1) If inconsistent(E, N), then return $\emptyset$.
2) Otherwise, $G :=$ ReducedGröbnerBasis$(F \cup E)$.
3) If $1 \in G$, then return $\{(E, N, \{1\})\}$.
4) Let $G_r := G \cap k[U]$.
5) If inconsistent$(E, G_r \times N)$, then $\mathcal{CGS} := \emptyset$, else $\mathcal{CGS} := \{(E, G_r \times N, \{1\})\}$.
6) If inconsistent(G_r, N), then return $\mathcal{CGS}$.
7) Otherwise, let $G_m :=$ MDBasis$(G \setminus G_r)$.
8) If consistent$(G_r, N \times \{h\})$, then $\mathcal{CGS} := \mathcal{CGS} \cup \{(G_r, N \times \{h\}, G_m)\}$, where $h = \mathrm{lcm}\{h_1, \cdots, h_k\}$, $h_i = \mathrm{lc}_X(g_i)$ and $g_i \in G_m$.
9) return $\mathcal{CGS} \cup \bigcup_{h_i \in \{h_1, \cdots, h_k\}}$ KSW-CGS$(G_r \cup \{h_i\}, N \times \{h_1 h_2 \cdots h_{i-1}\}, G \setminus G_r)$.

In the above algorithm, $A \times B := \{fg \mid f \in A, g \in B\}$. Inconsistent$(E, N)$ and consistent$(G_r, N \times \{h\})$ are subroutines used to check whether the algebraically constructible set $\boldsymbol{V}(E) \setminus \boldsymbol{V}(N)$ is empty.

From Theorem 3.9, it is easy to see that G_m has less polynomial than G_r since G_m is a subset of $G \setminus G_r$. This fact makes KSW-CGS will produce less branches or segments, and hence, have better performance than SS-CGS. In [37], Kapur, et al. also used many tricks and optimizations to check the consistency of parametric constraints. Please see [37] for more details.

In [39], Kapur, et al. proposed a new algorithm to compute comprehensive Gröbner systems and comprehensive Gröbner bases simultaneously. The key idea is that all the terms for every polynomial produced in KSW-CGS will be stored, no matter whether they are zero or not under specialization. All the polynomials are divided into two parts: Nonzero part and zero part for the specialization, and a module structure in $(k[U][X])^2$ is used to store these two parts separately. They collected all the two parts during the calculation process and eventually got a CGS and a faithful CGB. We first make a simple modification of the Algorithm KSW-CGS(E, N, F) to satisfy the computation of CGB.

Algorithm CGSMainMod($\boldsymbol{E}, \boldsymbol{N}, \boldsymbol{F}$)

Input (E, N, F): E, N, finite subsets of $k[U]$; F, a finite subset of $k[U][X]$.

Output $\mathcal{CGS}$: A finite set of 3-tuples $(E_i, N_i, \boldsymbol{G}_i)$ such that $\{(V(E_i) \setminus V(N_i), G_i^{1\mathrm{st}})\}$, where $G_i^{1\mathrm{st}} = \{g \mid (g, \overline{g}) \in \boldsymbol{G}_i\}$, constitutes a comprehensive Gröbner system on $V(E) \setminus V(N)$ for F,

and for each $(g,\overline{g}) \in \boldsymbol{G}_i$, $g+\overline{g} \in \langle F\rangle$ and $\sigma(\overline{g})$ is 0 for every parameter specialization σ from $V(E_i) \setminus V(N_i)$.

1) If inconsistent(E,N), then return $\emptyset$.

2) Otherwise, $\boldsymbol{G}_0 :=$ ReducedGröbnerBasis$(\{(f,0) \mid f \in F\} \cup \{(e,-e) \mid e \in E\})$.

3) $\boldsymbol{G} := \boldsymbol{G}_0 \setminus \{(g,\overline{g}) \in \boldsymbol{G}_0 \mid g = 0\}$ and $G^{1\text{st}} := \{g \mid (g,\overline{g}) \in \boldsymbol{G}\}$.

4) If there exists $(g,\overline{g}) \in \boldsymbol{G}$ such that $g = 1$, then return $\{(E,N,\{(g,\overline{g})\})\}$.

5) Let $\boldsymbol{G}_r := \{(g,\overline{g}) \in \boldsymbol{G} \mid g \in k[U]\}$ and $G_r := \{g \mid (g,\overline{g}) \in \boldsymbol{G}_r\}$.

6) If inconsistent$(E, G_r \times N)$, then $\mathcal{CGS} := \emptyset$, else $\mathcal{CGS} := \{(E, G_r \times N, \boldsymbol{G}_r)\}$.

7) If inconsistent(G_r, N), then return $\mathcal{CGS}$.

8) Otherwise, let $G_m :=$ MDBasis$(G^{1\text{st}} \setminus G_r)$ and $\boldsymbol{G}_m := \{(g,\overline{g}) \in \boldsymbol{G} \setminus \boldsymbol{G}_r \mid g \in G_m\}$.

9) If consistent$(G_r, N \times \{h\})$, then $\mathcal{CGS} := \mathcal{CGS} \cup \{(G_r, N \times \{h\}, \boldsymbol{G}_m)\}$, where $h = \text{lcm}\{h_1, \cdots, h_k\}$ and $\{h_1, \cdots, h_k\} = \{\text{lc}_X(g) \mid g \in G_m\}$.

10) Return $\mathcal{CGS} \cup \bigcup_{h \in [h_1,\cdots,h_k]}$ CGSMainMod$(G_r \cup \{h_i\}, N \times \{h_1 h_2 \cdots h_{i-1}\}, \{g+\overline{g} \mid (g,\overline{g}) \in \boldsymbol{G} \setminus \boldsymbol{G}_r\})$.

Now we can use Algorithm CGSMainMod(E,N,F) to compute a CGB of F on the algebraically constructible set $\boldsymbol{V}(E) \setminus \boldsymbol{V}(N)$.

Algorithm KSW-CGB$(\boldsymbol{E},\boldsymbol{N},\boldsymbol{F})$
Input (E,N,F): E, N, finite subsets of $k[U]$; F, a finite subset of $k[U,X]$.
Output A CGB of the set F on $\boldsymbol{V}(E) \setminus \boldsymbol{V}(N)$.

1) $\mathcal{CGS} :=$ CGSMainMod(E,N,F), where $\mathcal{CGS}$ is a finite set of 3-tuples $(E_i, N_i, \boldsymbol{G}_i)$ such that $\{(\boldsymbol{V}(E_i) \setminus \boldsymbol{V}(N_i), G_i^{1\text{st}})\}$, where $G_i^{1\text{st}} = \{g \mid (g,\overline{g}) \in \boldsymbol{G}_i\}$, constitutes a CGS on $\boldsymbol{V}(E) \setminus \boldsymbol{V}(N)$ for F, and for each $(g,\overline{g}) \in \boldsymbol{G}_i$, $g+\overline{g} \in \langle F\rangle$ and $\sigma(\overline{g})$ is 0 for every parameter specialization σ from $\boldsymbol{V}(E_i) \setminus \boldsymbol{V}(N_i)$.

2) Return $\{g+\overline{g} \mid (g,\overline{g}) \in \boldsymbol{G}_i$ for all $i\}$.

The following theorem guarantees the correctness of the Algorithm KSW-CGB(E,N,F) for computing a CGB.

Theorem 3.10 *Let F be a set of polynomials in $k[U][X]$, E be a subset of $k[U]$, and $\boldsymbol{M}$ be a $k[U][X]$-module generated by $\{(f,0)|f \in F\} \cup \{(g,-g)|g \in E\}$. Suppose $\boldsymbol{G}$ is a Gröbner basis of the module $\boldsymbol{M}$ w.r.t. an order extended from $\prec_{X,U}$ in a position over term fashion with $(0,1) \prec (1,0)$, where $\prec_{X,U}$ is an admissible block order with $U \ll X$.*

Denote $G^{1\text{st}} = \{g \mid (g,\overline{g}) \in \boldsymbol{G}\}$, $G_r = G^{1\text{st}} \cap k[U]$ and $G_m =$ MDBasis$(G^{1\text{st}} \setminus G_r)$. $\boldsymbol{G}_m$ is a subset of $\boldsymbol{G}$ such that $\{(g,\overline{g}) \in \boldsymbol{G}_m \mid g \in G_m\}$. If σ is a specialization from $k[U]$ to L such that

1) $\sigma(g) = 0$ *for* $g \in G_r$, *and*

2) $\sigma(h) \neq 0$, *where* $h = \Pi_{g \in G_m} \mathrm{lc}_X(g) \in k[U]$,

then

1) *for each* $(g, \overline{g}) \in \boldsymbol{G}_m$, $g + \overline{g} \in \langle F \rangle$ *and* $\sigma(\overline{g}) = 0$, *and*

2) $\{\sigma(g, \overline{g}) \mid (g, \overline{g}) \in \boldsymbol{G}_m\}$ *is a minimal Gröbner basis of* $\langle \sigma(F) \rangle$ *in* $L[X]$ *w.r.t.* $\prec_X$.

As we see, the second component of every tuple is 0 under specialization, and the sum of the first component and the second component in the tuple is in the ideal generated by $\langle F \rangle$. This property can help us obtain a CGB for F.

The above KSW algorithm has been implemented in computer algebra systems such as Maple, Singular, and Magma. You can download the source codes from the following website:

`http://www.mmrc.iss.ac.cn/~dwang/software.html`.

This algorithm also has been included in Montes's "Gröbner cover" in Singular, please see:

`https://mat.upc.edu/en/people/antonio.montes/`.

4 Some Applications of CGS

With the improvement of the computational efficiency of CGS, many algorithms on CGS are applied to various fields. In the following, we will introduce the applications of CGS: Solving systems of parametric polynomial equations, automated discovering of geometric theorem, quantifier elimination, and computing parametric polynomial GCD.

4.1 Solving Systems of Parametric Polynomial Equations

We know that Gröbner bases can be used for solving systems of polynomial equations. It is natural that CGS can be used to solve systems of parametric polynomial equations.

For a given parametric polynomial equations $F = 0$, The following problem is of interest. That is: For what values of the parameters, $F = 0$ have solutions or have no solution. If it has infinitely many solutions, what is the dimension? If it has finitely many solutions, what is the number of solutions. This problem can be solved by CGS easily.

Assume that $\mathcal{G} = \{(A_1, G_1), \cdots, (A_l, G_l)\}$ is a CGS for F. Then for every $\alpha \in A_i$, it is easy to give the structure of the solution space for F according to the Gröbner basis G_i, where $i = 1, \cdots, l$. For example, considering the following parametric polynomial equations:

$$\begin{cases} f_1 = x_4 + b - d = 0, \\ f_2 = x_1 + x_2 + x_3 + x_4 - a - c - d = 0, \\ f_3 = x_1x_4 + x_2x_3 + x_3x_4 - ac - ad - cd = 0, \\ f_4 = x_1x_3x_4 - acd = 0, \end{cases}$$

where $f_1, f_2, f_3, f_4 \in \mathbb{C}[a, b, c, d][x_1, x_2, x_3, x_4]$, $\prec_X$ and $\prec_U$ are all degree lexicographic orders with $x_1 < x_2 < x_3 < x_4$ and $d < c < b < a$, respectively. The CGS of this parametric polynomial system on $\mathbb{C}^4$ is:

Table 1 The CGS of $\langle f_1, f_2, f_3, f_4\rangle$

No.	A_i	G_i
1	$\boldsymbol{V}(b-d)\setminus\boldsymbol{V}(acd)$	$\{1\}$
2	$\boldsymbol{V}(b-d,acd)$	$\{\underline{x_4}, \underline{x_3}+x_2+x_1-a-c-d,$ $\underline{x_2^2}+x_2(x_1-a-c-d)+a(c+d)+cd\}$
3	$\mathbb{C}^4\setminus\boldsymbol{V}(b-d)$	$\{\underline{x_4}+b-d, \underline{x_3}+x_2+x_1-a-b-c,$ $\underline{x_2^2}+x_2x_1-x_2(a-2b-c+d)+a(b+c)+b(b+c-d),$ $\underline{x_1x_2}(b-d)+x_1^2(b-d)+-x_1(ab+ad-b^2-bc+bd+cd)-acd,$ $\underline{x_1^2}(b-d)^2+x_2acd+x_1(abc+abd-acd$ $-ad^2+bcd-cd^2)-acd(b-acd)\}$

The first branch tells us that if $b=d$ and $acd\neq 0$, then the parametric polynomial system has no solution. The second branch implies that the parametric polynomial system has an infinite number of solutions, and the dimension of the solution space is 1. The parametric polynomial system has three solutions in the third branch.

4.2 Automatic Geometric Theorem Discovering

A geometric statement contains some hypotheses and a conclusion, the hypotheses can be expressed by parametric polynomial equations $\{h_1=0,\cdots,h_s=0\}$ and the conclusion is expressed by $h=0$, where $h_1,\cdots,h_s,h\in k[U][X]$. For any given geometric statement, we want to solve a problem: How can the fact that h follows from $h_1,\cdots,h_s$ be deduced algebraically? Two algebraic methods, Wu's method and Gröbner basis method, are often used to prove or discover geometric theorems mechanically. Wu Wen-Tsün[48, 49] successfully proposed the characteristic sets method and proved many geometric theorems automatically. In the following, we will show how to discover geometric theorems mechanically by using CGS.

It is well-known that the radical ideal membership problem can be solved by the following theorem.

Theorem 4.1 (see [50]) *Let I be a polynomial ideal in $k[X]$, h is in the radical of I if and only if $\{1\}$ is the Gröbner basis of ideal $\langle I, yh-1\rangle$, where y is a new variable.*

Let I be the ideal generated by the hypotheses polynomials $h_1,\cdots,h_s$ of a geometric statement, and h be the conclusion polynomial. If $\{1\}$ is the Gröbner basis of ideal $\langle I, yh-1\rangle$, then $h=0$ can be deduced from $I=0$.

Extending this theorem to the parametric case, Chen, et al.[8] first proposed a method for proving geometric theorems mechanically by using CGS. Moreover, this method can be directly used to discover geometric theorems. Wang and Lin[10] used this method for discovering geometric theorems mechanically. The method was further investigated by Montes and Recio[11].

Let I be the ideal generated by parametric polynomials $h_1,\cdots,h_s$ of a geometric statement, and h be the conclusion polynomial, where $h_1,\cdots,h_s,h\in k[U][X]$. Assume that

$$\{(A_1,G_1),\cdots,(A_r,G_r),(A_{r+1},G_{r+1}),\cdots,(A_s,G_s)\}$$

is a CGS of $\langle I, yh-1\rangle$ w.r.t. some term order on $y\gg X\gg U$, where $G_i=\{1\}$ for $i=1,2,\cdots,r$

and $G_i \neq \{1\}$ for $i = r+1, \cdots, s$, then the geometric statement is true under the constraint $A_1, \cdots, A_r$ and false under $A_{r+1}, \cdots, A_s$. Now, let's look at the following example.

Example 4.2 Let ΔABC be a triangle in the plane, M_1, M_2, M_3 be the midpoints of the edges AB, BC, AC respectively. The questions is: For what kind of triangles, three midpoint M_1, M_2, M_3 and point A are on a same circle.

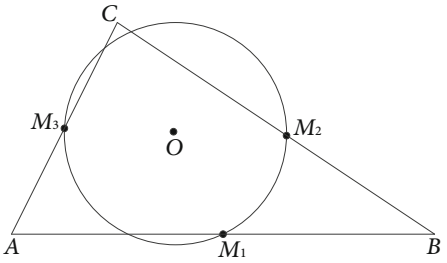

Figure 1

Let O be the center of the circle which passes the three midpoints M_1, M_2 and M_3. From the above picture, it is easy to see that point A is not on the circle O generally. Without loss of generality, we take the coordinates of the points $A(0,0)$, $B(u_1,0)$, $C(u_2,u_3)$ and $M_1(x_1,0)$, $M_2(x_2,x_3)$, $M_3(x_4,x_5)$, and $O(x_6,x_7)$. The coordinates of A,B,C are in terms of u_1,u_2,u_3, where u_1,u_2,u_3 are parameters, which can take any value. This means that points A,B,C can move on the plane freely. The coordinates of the M_1,M_2,M_3,O are in terms of the variables $x_1,\cdots,x_7$. This implies that the midpoints and the center of the circle are not free, they depend on the given triangle.

The hypotheses of the above problem can be expressed as following.

$$\begin{cases} h_1 = 2x_1 - u_1 = 0, & (M_1 \text{ is the midpoint of } AB) \\ h_2 = 2x_2 - (u_1+u_2) = 0, & (M_2 \text{ is the midpoint of } BC) \\ h_3 = 2x_3 - u_3 = 0, & (M_2 \text{ is the midpoint of } BC) \\ h_4 = 2x_4 - u_2 = 0, & (M_3 \text{ is the midpoint of } AC) \\ h_5 = 2x_5 - u_3 = 0, & (M_3 \text{ is the midpoint of } AC) \\ h_6 = (x_6-x_4)^2 + (x_7-x_5)^2 - ((x_6-x_1)^2 + x_7^2) = 0, & (|M_3O| = |M_1O|) \\ h_7 = (x_6-x_2)^2 + (x_7-x_3)^2 - ((x_6-x_1)^2 + x_7^2) = 0. & (|M_2O| = |M_1O|) \end{cases}$$

The conclusion $|AO| = |M_1O|$ is expressed as

$$h = x_6^2 + x_7^2 - ((x_6-x_1)^2 + x_7^2) = 0.$$

For the term order with $y > x_7 > x_6 > x_5 > x_4 > x_4 > x_3 > x_2 > x_1$, the CGS of $\langle h_1, \cdots, h_7, yh-1\rangle$ is $\{(A_i, G_i)_{i=1}^6\}$, where (A_i, G_i) is as follows.

For segment $A_1 = \boldsymbol{V}(u_2) \setminus \boldsymbol{V}(u_1u_3)$, u_2 is the x-coordinate of C, $u_2 = 0$ means that $\angle A$ is a right angle. $G_1 = \{1\}$ implies the conclusion is true, that is point A is also on the circle O.

For segments A_2, A_3, A_4, although the corresponding Gröbner bases G_2, G_3, G_4 are $\{1\}$, they are degenerated cases. For segments A_5 and A_6, since the Gröbner bases are not $\{1\}$, the conclusion is false under the conditions A_5 and A_6. Please notice that segment A_6 is the generic case because there is no equations constraints over the parameters u_1, u_2, u_3 in A_6.

Table 2 The CGS of $\langle h_1, \cdots, h_7, yh-1\rangle$ in Example 4.2

No.	A_i	G_i
1	$\boldsymbol{V}(u_2) \setminus \boldsymbol{V}(u_1u_3)$	$\{1\}$
2	$\boldsymbol{V}(u_2, u_3) \setminus \boldsymbol{V}(u_1)$	$\{1\}$
3	$\boldsymbol{V}(u_3) \setminus \boldsymbol{V}(u_1u_2(u_1-u_2))$	$\{1\}$
4	$\boldsymbol{V}(u_1)$	$\{1\}$
5	$\boldsymbol{V}(u_1-u_2, u_3) \setminus \boldsymbol{V}(u_1)$	$\{2x_1-u_2, x_2-u_2, x_3, 2x_4-u_2, x_5, -3u_2+4x_6, u_2^2y-2\}$
6	$\mathbb{C}^3 \setminus \boldsymbol{V}(u_1u_2u_3)$	$\{2x_1-u_2, 2x_2-u_1-u_2, 2x_3-u_3, 2x_4-u_2, 2x_5-u_3, 4x_6-u_1-2u_2, 4u_3x_7-u_3^2+u_2^2-u_1u_2, u_1u_2y-2\}$

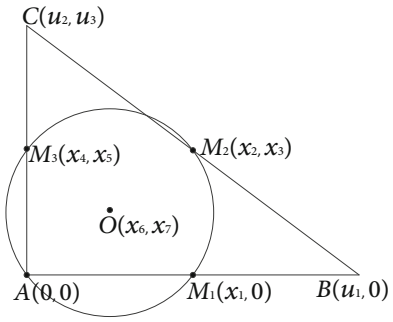

Figure 2

4.3 Quantifier Elimination by CGS

Quantifier elimination (QE) is one of the basic problems in mathematics research, and it has a wide range of applications in scientific research and practical engineering, such as polynomial optimization, surface intersection, robot motion planning, and so on. Thereby, QE has attracted much attention over the past several decades, and great progress has been made on the algorithms of QE. There are two major methods to solve QE: The cylindrical algebraic decomposition (CAD) algorithm[51, 52] and CGS. In the past three decades, several very important improvements have been made to CAD, and it has been one of the most efficient methods for real QE up to present. Unfortunately, there are some demerits of CAD. For example, CAD often executes useless computations on unnecessary cells, which leads to huge computation and time consuming. An alternative real QE algorithm was proposed by Weispfenning[16] in 1998. Using the real root counting theorem and CGS, he can eliminate all quantifies in the given quantified formula. As the computational efficiency of CGS algorithms improves, we can use

them to solve QE. In 2015, Fukasaku, et al.[18] further improved the Weispfenning's QE algorithm. They modified the Suzuki-Sato's CGS algorithm[34] into an optimal form for applying to QE, and obtained an efficient QE algorithm. The implementation in [18] shows that for many examples the proposed algorithm is superior to other existing algorithms.

Next, we introduce QE and apply CGS to solve QE. For an arbitrary given algebraic first order formula over the real number field $\mathbb{R}$ containing quantifiers "$\exists$" and "$\forall$", we can always produce an equivalent quantifier free formula. As a simple example, for a formula $\exists x \in \mathbb{R}(x^2 + Ax + B = 0)$, we can produce an equivalent quantifier free formula $A^2 - 4B >= 0$. This procedure is called QE. Using a prenex normal form of a given formula, we can reduce any QE problem to a QE problem of the following basic formula with polynomials $f_1, \cdots, f_s, g_1, \cdots, g_t$ of $k[U][X]$:

$$\exists U(f_1(U, X) = 0 \wedge \cdots \wedge f_s(U, X) = 0 \wedge g_1(U, X) \neq 0 \wedge \cdots \wedge g_t(U, X) \neq 0).$$

The above formula is equivalent to the following formula without inequalities:

$$\exists y, U(f_1(U, X) = 0 \wedge \cdots \wedge f_s(U, X) = 0 \wedge 1 - yg_1(U, X) \cdots g_t(U, X) = 0),$$

where y is a new variable with $y \gg X \gg U$. (There is no doubt that we can introduce t new variables $y_1, \cdots, y_t$ to remove inequalities: $1 - y_1 g_1(U, X) = 0, \cdots, 1 - y_t g_t(U, X) = 0$.) Combining real root counting theorem, we can eliminate quantifiers by the computation of a suitable CGS. See [18–21] for more details.

This following example shows that it is possible to use a plane to cut a pyramid to get a regular pentagon, and it is an illustration example of using CGS to solve QE problem. This problem was first solved by Wang in his master degree thesis[53, 54] in 1990, and it was called as "Beijing Theorem" by Deakin in [55]. In [53, 54], the problem was solved by Wu's method. In the following, we will solve it by CGS method.

Example 4.3 (see [53, 54]). Let $P - ABCD$ be a pyramid whose base surface $ABCD$ is a square, and O be the center of the square. We further assume P is just on the top of O, i.e., $PO \perp$ the base surface $ABCD$. Let E, F, G, H, I be the intersection of a plane φ and AB, BC, PC, PD, PA respectively. Can $EFGHI$ be a regular pentagon?

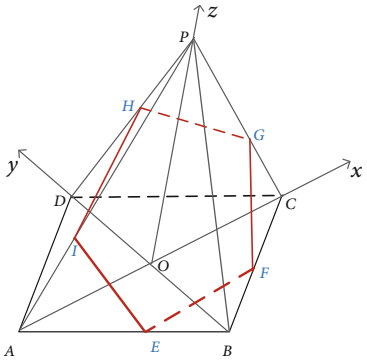

Figure 3

Setting a coordinate system, the coordinates of points O, A, B, C, D are set to $O(0,0,0)$, $A(-1,0,0)$, $B(0,-1,0)$, $C(1,0,0)$, $D(0,1,0)$. If $EFGHI$ is a regular pentagon, we have $GH \parallel EF \parallel AC$. It is reasonable to set $E(-x_1,x_2,0)$, $F(x_1,x_2,0)$, $G(x_3,0,x_4)$, $I(-x_3,0,x_4)$, $H(0,x_5,x_6)$, $P(0,0,a)$. Note that $x_1,\cdots,x_6$ are variables and a is a parameter. Suppose J and K are the midpoints of GI and EF respectively. Since $EFGHI$ is a regular pentagon, then we have

$$\mathcal{H}:\begin{cases} h_1 = x_2 - x_1 + 1 = 0 & (F \text{ is on line } BC)\\ h_2 = a(x_3-1) + x_4 = 0 & (G \text{ is on line } PC)\\ h_3 = a(x_5-1) + x_6 = 0 & (H \text{ is on line } PD)\\ h_4 = x_6x_2 + x_4(x_5 - x_2) = 0 & (J \text{ is on line } HK)\\ h_5 = x_3^2 + x_5^2 + (x_6-x_4)^2 - 4x_1^2 = 0 & (|HI| = |EF|)\\ h_6 = (x_3-x_1)^2 + x_2^2 + x_4^2 - 4x_1^2 = 0 & (|IE| = |EF|)\\ h_7 = (x_3+x_1)^2 + x_2^2 + x_4^2 - x_1^2 - (x_5-x_2)^2 - x_6^2 = 0 & (|EG| = |EG|)\\ h_8 = x_3^2 - x_1x_3 - x_1^2 = 0 & (|GI|^2 - |GI||EF| - |EF|^2 = 0) \end{cases}.$$

$EFGHI$ is a regular pentagon, that is,

$$\exists x_1,x_2,x_3,x_4,x_5,x_6,x_7(\mathcal{H}=0).$$

We can solve this problem by using CGS, and the CGS $\{(A_i,G_i)_{i=1}^5\}$ of $\langle\mathcal{H}\rangle$ is as follows.

Table 3 the CGS of $\langle\mathcal{H}\rangle$ in Example 4.3

No.	A_i	G_i
1	$\mathbb{C}\setminus \boldsymbol{V}(a(a-1)(a+1)(a^2+1))$	$\{1\}$
2	$\boldsymbol{V}(a)$	$\{x_1^4+6x_1^3+x_1^2-4x_1+1, x_2-x_1+1, x_4,$ $2x_3+5x_1^4+32x_1^3+17x_1^2-16x_1+1, 5x_5-$ $7x_3x_1^2-23x_3x_1-2x_1^3-2x_1^2-14x_1+8, x_6\}$
3	$\boldsymbol{V}(a-1)$	$\{x_1^2-3x_1+1, x_2-x_1+1, x_3+x_1-1,$ $x_4-x_1, x_5-x_1, x_6+x_1-1\}$
4	$\boldsymbol{V}(a+1)$	$\{x_1^2-3x_1+1, x_2-x_1+1, x_3+x_1-1,$ $x_4+x_1, x_5-x_1, x_6-x_1+1\}$
5	$\boldsymbol{V}(a^2+1)$	$\{x_2-x_1+1, x_3-x_1, x_4+ax_1-a,$ $x_1^2, 2x_5-x_1, 2x_6+ax_1-2a\}$

According to the above CGS, since $G_i \neq \{1\}$ for i=2, 3, 4, 5, then $\mathcal{H}=0$ has solutions for under the condition A_i for $i=2,3,4,5$. We get the equivalent quantifier free formula for $\exists x_1,x_2,x_3,x_4,x_5,x_6,x_7(\mathcal{H}=0)$:

$$a=0 \vee a-1=0 \vee a+1=0 \vee a^2+1=0.$$

For this problem, we can get a quantifier free formula over the **real number field** $\mathbb{R}$. $\exists x_1,x_2,x_3,x_4,x_5,x_6,x_7 \in \mathbb{R}(\mathcal{H}=0)$ is equivalent to the following formula:

$$a=0 \vee a-1=0 \vee a+1=0.$$

It is easy to see that $a = 0$ is a degenerated case. i.e., P is on the base plane $ABCD$ and P becomes to the point O. If $a = 1$ or $a = -1$, then there is a plane to cut the pyramid $P - ABCD$ to get a regular pentagon $EFGHI$. For $a = 1$, there are two solutions for $G_3 = 0$. They are $\{x_1 = \frac{3-\sqrt{5}}{2}, x_2 = \frac{1-\sqrt{5}}{2}, x_3 = -\frac{1-\sqrt{5}}{2}, x_4 = \frac{3-\sqrt{5}}{2}, x_5 = \frac{3-\sqrt{5}}{2}, x_6 = -\frac{1-\sqrt{5}}{2}\}$ and $\{x_1 = \frac{3+\sqrt{5}}{2}, x_2 = \frac{1+\sqrt{5}}{2}, x_3 = -\frac{1+\sqrt{5}}{2}, x_4 = \frac{3+\sqrt{5}}{2}, x_5 = \frac{3+\sqrt{5}}{2}, x_6 = -\frac{1+\sqrt{5}}{2}\}$. The first solution shows that $EFGHI$ is a regular pentagon, and the second solution means that $EFGHI$ is a regular five-pointed star.

4.4 Computing Parametric Polynomial GCD

As we know, polynomial GCD (greatest common divisor) computation plays an important role in computer algebra. Many efficient algorithms have been proposed to compute the polynomial GCD. For an ideal in univariate polynomial ring with coefficient in a field, the reduced Gröbner basis contains only one polynomial, which is just the GCD of all the polynomials in the ideal. This conclusion still remains true for univariate polynomial ideals with parameters, and hence, CGS can give the GCD directly. For multivariate case, little progress has been made. In 2017, Nagasaka[56] presented two algorithms to compute the GCD for the first time. In 2018, Kapur, et al.[57] proposed a new efficient algorithm which is based on a CGS of a quotient ideal to compute parametric polynomial GCD. Experimental data suggests that their algorithm is superior in practice in comparison with the pervious algorithms. The following theorem and example which come from [57] will tell us how to use CGS to compute parametric polynomial GCD.

Theorem 4.4 (see [57]) *Given $f_1, f_2 \in k[U][X]$ and an algebraically constructible set $A = \boldsymbol{V}(E) \setminus \boldsymbol{V}(N) \subset \overline{k}^m$, let $\mathcal{G} = \{(A_i, G_i)\}_{i=1}^l$ be a minimal comprehensive Gröbner system of the module $W = \langle f_1 \cdot \boldsymbol{e}_1, f_2 \cdot \boldsymbol{e}_1 - \boldsymbol{e}_2\rangle$ on A w.r.t. an order extended from $\prec_X$ in a position over term fashion with $\boldsymbol{e}_2 < \boldsymbol{e}_1$, where $\boldsymbol{e}_1 = (1, 0)$ and $\boldsymbol{e}_2 = (0, 1)$. For each branch (A_i, G_i) let $H_i = \{h \in k[U][X] \mid h \cdot \boldsymbol{e}_2 \in G_i\}$. Then we have the following results.*

1) *If H_i is empty, then $\gcd(\sigma_\alpha(f_1), \sigma_\alpha(f_2)) = \sigma_\alpha(f_2)$ for any $\alpha \in A_i$.*

2) *If H_i is not empty, then $H_i = \{g_i\}$ and $\gcd(\sigma_\alpha(f_1), \sigma_\alpha(f_2)) = \frac{\sigma_\alpha(f_1)}{\sigma_\alpha(g_i)}$ for any $\alpha \in A_i$.*

Example 4.5 (see [57]) Let $f_1, f_2, f_3 \in \mathbb{C}[U][X]$ be as follows:

$$\begin{cases} f_1 = ax_1^2 + bx_1x_2 + a^2x_1x_3 + abx_1 + abx_2x_3 + b^2x_2, \\ f_2 = ax_1^2 + bx_1x_2 + (ab - a)x_1x_3 - a^2x_1 + (b^2 - b)x_2x_3 - abx_2, \\ f_3 = ax_1^2 + bx_1x_2 + a^2x_1x_3 + (a^2 - ab)x_1 + abx_2x_3 + (ab - b^2)x_2, \end{cases}$$

where $U = \{a, b\}$, $X = \{x_1, x_2, x_3\}$, $\prec_X$ and $\prec_U$ are all lexicographic orders with $x_3 < x_2 < x_1$ and $b < a$, respectively.

We first compute a minimal CGS $\mathcal{G}_{12}$ of $\langle f_1 \cdot \boldsymbol{e}_1, f_2 \cdot \boldsymbol{e}_1 - \boldsymbol{e}_2\rangle$, and obtain six branches in $\mathcal{G}_{12}$. The first branch of $\mathcal{G}_{12}$ is $(A_1, G_1) = (\mathbb{C}^2 \setminus \boldsymbol{V}(a(a - b + 1)), \{(x_1 + ax_3 + b) \cdot \boldsymbol{e}_2, ((a^2 - ab + a)x_1x_3 + (a^2 + ab)x_1 + (ab - b^2 + b)x_2x_3 + (ab + b^2)x_2) \cdot \boldsymbol{e}_1 + \boldsymbol{e}_2, f_2 \cdot \boldsymbol{e}_1 - \boldsymbol{e}_2\})$. Then,

$H_1 = \{x_1 + ax_3 + b \in \mathbb{C}[U][X] \mid (x_1 + ax_3 + b) \cdot \boldsymbol{e}_2 \in G_1\}$ and the GCD of f_1 and f_2 on A_1 is $h_1 = f_1/(x_1 + ax_3 + b) = ax_1 + bx_2$.

Next, we compute the GCD of h_1 and f_3 on A_1. A minimal CGS of $\langle h_1 \cdot \boldsymbol{e}_1, f_3 \cdot \boldsymbol{e}_1 - \boldsymbol{e}_2\rangle$ on A_1 has one branch: $(A_2, G_2) = (\mathbb{C}^2 \setminus \boldsymbol{V}(a(a-b+1)), \{\boldsymbol{e}_2, h_1 \cdot \boldsymbol{e}_1\})$. Then $H_2 = \{1\}$ and the GCD of h_1 and f_3 on A_1 is $h = h_1/1 = ax_1 + bx_2$. This implies that the GCD of f_1, f_2, f_3 on A_1 is $h = ax_1 + bx_2$.

Similarly, we can compute the GCDs of f_1, f_2, f_3 on other five branches and get the following results.

Table 4 The GCDs of f_1, f_2, f_3 in Example 4.5

No.	A_i	h (GCD)
1	$\mathbb{C}^2 \setminus \boldsymbol{V}(a(a-b+1))$	$ax_1 + bx_2$
2	$\boldsymbol{V}(a-b+1) \setminus \boldsymbol{V}((2b-1)(b-1))$	$(b-1)x_1 + bx_2$
3	$\boldsymbol{V}(2a+1, 2b-1)$	$x_1 - x_2$
4	$\boldsymbol{V}(a) \setminus \boldsymbol{V}(b(b-1))$	x_2
5	$\boldsymbol{V}(a, b-1)$	x_2
6	$\boldsymbol{V}(a, b)$	0

5 Conclusions

In this paper, we studied the algorithms for computing CGS and CGB, as well as some of their applications. This area was initiated by Weispfenning in 1992[22], then many researchers have contributed to the development of this field. Suzuki and Sato made a major breakthrough in 2006. Different from the previous algorithms, they designed an algorithm to compute a Gröbner basis of a parametric ideal in $k[U][X]$. This method makes the algorithms for computing CGS and CGB can be easily implemented in the computer algebra systems which support an efficient implementation of a Gröbner basis algorithm. In 2010, Kapur, et al.[37,38] improved the Suzuki and Sato's work by using minimal Dickson basis to remove redundant branches, many tricks and heuristics are also used to improve the efficiency. In 2011, Kapur, et al.[39] proposed an algorithm to compute CGS and CGB simultaneously by computing a CGS for a module. These algorithms proposed by Kapur, et al.[40−43] are the most efficient algorithms for computing CGS and CGB so far. CGS and CGB have been widely applied in the fields of automatic geometric theorem proving and discovering, quantifier elimination, and so on.

We believe that some of the work in this area, such as canonical forms of CGS and CGB which can be further studied. We hope this survey will contribute to the further development of this research area.

References

[1] Donald B R, Kapur D, and Mundy J L, Symbolic and numerical computation for artificial intelligence, *Computational Mathematics and Applications*, Academic Press, Orlando, Florida, 1992, 52–55.

[2] Gao X S and Chou S C, Solving parametric algebraic systems, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, ACM Press, New York, 1992, 335–341.

[3] William Y S, An algorithm for solving parametric linear systems, *Journal of Symbolic Computation*, 1992, **13**(4): 353–394.

[4] Chen C, Golubitsky O, Lemaire F, et al., Comprehensive triangular decomposition, *International Workshop on Computer Algebra in Scientific Computing*, Springer, Berlin, 2007, 73–101.

[5] Lazard D and Rouillier F, Solving parametric polynomial systems, *Journal of Symbolic Computation*, 2007, **42**(6): 636–667.

[6] Huang Z, Parametric equation solving and quantifier elimination in finite fields with the characteristic set method, *Journal of Systems Science and Complexity*, 2012, **25**(4): 778–791.

[7] Chen Z H, Tang X X, and Xia B C, Generic regular decompositions for parametric polynomial systems, *Journal of Systems Science and Complexity*, 2015, **28**(5): 1194–1211.

[8] Chen X F, Li P, Lin L, et al., Proving geometric theorems by partitioned-parametric Gröbner bases, *International Workshop on Automated Deduction in Geometry*, 2004, 34–43.

[9] Lin L, Automated geometric theorem proving and parametric polynomial equations solving, Master Degree Thesis, Institute of Systems Science, CAS, Beijing, 2006.

[10] Wang D K and Lin L, Automatic discovering of geometric theorems by computing Gröbner bases with parameters. *The* 11*th Internatinal Conference on Applications of Computer Algebra*, 2005.

[11] Montes A and Recio T, Automatic discovery of geometry theorems using minimal canonical comprehensive Gröbner systems, *International Workshop on Automated Deduction in Geometry*, 2006, 113–138.

[12] Zhou J, Wang D K, and Sun Y, Automated reducible geometric theorem proving and discovery by Gröbner basis method, *Journal of Autotamed Reasoning*, 2017, **59**(3): 331–344.

[13] Botana F, Montes A, and Recio T, An algorithm for automatic discovery of algebraic loci, *International Workshop on Automated Deduction in Geometry*, 2012, 53–59.

[14] Gao X S, Hou X, Tang J, et al., Complete solution classification for the perspective-three-point problem, *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, **25**(8): 930–943.

[15] Zhou J and Wang D K, Solving the perspective-three-point problem using comprehensive Gröbner systems, *Journal of Systems Science and Complexity*, 2016, **29**(5): 1446–1471.

[16] Weispfenning V, A new approach to quantifier elimination for real algebra, *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Springer, 1998, 376–392.

[17] Kapur D, A quantifier-elimination based heuristic for automatically generating inductive assertions for programs, *Journal of Systems Science and Complexity*, 2006, **19**(3): 307–330.

[18] Fukasaku R, Iwane H, and Sato Y, Real quantifier elimination by computation of comprehensive Gröbner systems, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, ACM Press, Bath, 2015, 173–180.

[19] Fukasaku R, Inoue S, and Sato Y, On QE algorithms over an algebraically closed field based on

comprehensive Gröbner systems, *Mathematics in Computer Science*, 2015, **9**(3): 267–281.

[20] Fukasaku R, Iwane H, and Sato Y, Improving a CGS-QE algorithm, *Revised Selected Papers of the International Conference on Mathematical Aspects of Computer and Information Sciences*, Springer-Verlag, New York, 2015, 231–235.

[21] Fukasaku R, Iwane H, and Sato Y, On the implementation of CGS real QE, *International Congress on Mathematical Software*, Springer International Publishing, 2016, 165–172.

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

[23] Pesh M, Computing comprehensive Gröbner bases using MAS, *User Manual*, 1994.

[24] Kapur D, An approach for solving systems of parametric polynomial equations, *Principles and Practice of Constraint Programming*, MIT Press, Cambridge, Massachusetts, 1995, 217–224.

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

[26] Weispfenning V, Canonical comprehensive Gröbner bases, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, ACM Press, New York, 2002, 270–276.

[27] Weispfenning V, Canonical comprehensive Gröbner bases, *Journal of Symbolic Computation*, 2003, **36**(3): 669–683.

[28] Manubens M and Montes A, Improving DISPGB algorithm using the discriminant ideal, *J. Symbolic. Comput.*, 2006, **41**(11): 1245–1263.

[29] Suzuki A and Sato Y, An alternative approach to comprehensive Gröbner bases, *J. Symbolic. Comput.*, 2003, **36**(3–4): 649–667.

[30] Suzuki A and Sato Y, Comprehensive Gröbner bases via ACGB, *The* 10*th Internatinal Conference on Applications of Computer Algebra*, 2004, 65–73.

[31] Wibmer M, Gröbner bases for families of affine or projective schemes, *J. Symbolic. Comput.*, 2007, **42**(8): 803–834.

[32] Manubens M and Montes A, Minimal canonical comprehensive Gröbner system, *J. Symbolic. Comput.*, 2009, **44**(5): 463–478.

[33] Montes A and Wibmer M, Gröbner bases for polynomial systems with parameters, *J. Symbolic. Comput.*, 2010, **45**(12): 1391–1425.

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

[35] Kalkbrener M, On the stability of Gröbner bases under specializations, *Journal of Symbolic Computation*, 1997, **24**(1): 51–58.

[36] Nabeshima K, A speed-up of the algorithm for computing comprehensive Gröbner systems, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, 2007, 299–306.

[37] Kapur D, Sun Y, and Wang D K, A new algorithm for computing comprehensive Gröbner systems, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, 2010, 29–36.

[38] Kapur D, Sun Y, and Wang D K, An efficient algorithm for computing a comprehensive Gröbner system of a parametric polynomial systems, *Journal of Symbolic Computation*, 2010, **49**: 27–44.

[39] Kapur D, Sun Y, and Wang D K, Computing comprehensive Gröbner systems and comprehensive Gröbner bases simultaneously, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, 2011, 193–200.

[40] Kapur D, Sun Y, and Wang D K, An efficient method for computing comprehensive Gröbner

bases, *Journal of Symbolic Computation*, 2013, **52**: 124–142.

[41] Kapur D and Yang Y, An algorithm for computing a minimal comprehensive Gröbner basis of a parametric polynomial system, *Proceedings of Conference Encuentros de Algebra Comptacionaly Aplicaciones* (*EACA*), Invited Talk, Barcelona, Spain, 2014, 21–25.

[42] Kapur D and Yang Y, An algorithm to check whether a basis of a parametric polynomial system is a comprehensive Gröbner basis and the associated completion algorithm, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, 2015, 243–250.

[43] Kapur D, Comprehensive Gröbner basis theory for a parametric polynomial ideal and the associated completion algorithm, *Journal of Systems Science and Complexity*, 2017, **30**(1): 196–233.

[44] Hashemi A, Darmian M D, and Barkhordar M, Gröbner systems conversion, *Mathematics in Computer Science*, 2017, **11**(1): 61–77.

[45] Fukuda K, Jensen A, Lauritzen N, et al., The generic Gröbner walk, *J. Symb. Comput.*, 2007, **42**(3): 298–312.

[46] Hashemi A, Darmian M D, and Barkhordar M, Universal Gröbner basis for parametric polynomial ideals, *The International Congress on Mathematical Software*, Springer, Cham, 2018, 191–199.

[47] Kurata Y, Improving Suzuki-Sato's CGS algorithm by using stability of Gröbner bases and basic manipulations for efficient implementation, *Communications of the Japan Society for Symbolic and Algebraic Computation*, 2011, **1**: 39–66.

[48] Wu W T, On the decision problem and the mechanization of theorem proving in elementary geometry, *Sci. Sin.*, 1978, **21**: 159–172.

[49] Wu W T, Basic principles of mechanical theorem proving in elementary geometries, *J. Autom. Reason*, 1986, **2**(3): 221–252.

[50] Cox D, Little J, and O'shea D, *Ideals, Varieties, and Algorithms*, Springer, New York, 1992.

[51] Caviness B F and Johnson J R, *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Springer Science and Business Media, New York, 2012.

[52] Collins G E, Quantifier elimination for real closed fields by cylindrical algebraic decomposition, *Automata Theory and Formal Languages* (*Second GI Conf., Kaiserslautern*), 1975, 134–183.

[53] Wang D K, Mechanical proving of a group of space geometric theorem, Master Degree Thesis, Institute of Systems Science, CAS, Beijing, 1990.

[54] Wang D K, A mechanical solution to a group of space geometry problem, *Proceedings of the International Workshop on Mathematics Mechanization*, 1992, 236–243.

[55] Deakin M A B, A simple proof of the Beijing theorem, *The Mathematical Gazette*, 1992, **76**(476): 251–254.

[56] Nagasaka K, Parametric greatest common divisors using comprehensive Gröbner systems, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, 2017, 341–348.

[57] Kapur D, Lu D, Monagan M, et al., An efficient algorithm for computing parametric multivariate polynomial GCD, *Proceedings of International Symposium on Symbolic and Algebraic Computation*, 2018, 239–246.