A Study on Gröbner Basis with Inexact Input

Nagasaka, Kosaku

(Citation)
Lecture Notes in Computer Science, 5743:247-258

(Issue Date)
2009

(Resource Type)
journal article

(Version)
Accepted Manuscript

(URL)
https://hdl.handle.net/20.500.14094/90001897

A Study on Gröbner Basis with Inexact Input

Kosaku Nagasaka

Kobe University, Japan
nagasaka@main.h.kobe-u.ac.jp

Abstract. Gröbner basis is one of the most important tools in recent symbolic algebraic computations. However, computing a Gröbner basis for the given polynomial ideal is not easy and it is not numerically stable if polynomials have inexact coefficients. In this paper, we study what we should get for computing a Gröbner basis with inexact coefficients and introduce a naive method to compute a Gröbner basis by reduced row echelon form, for the ideal generated by the given polynomial set having a priori errors on their coefficients.

1 Introduction

Recently, computing a Gröbner basis for polynomials with inexact coefficients has been studied by several researchers ([1], [2], [3], [4], [5], [6], [7]). In Sasaki and Kako [1], this problem is classified into the first and the second kinds of problems. The first kind is computing a Gröbner basis for the ideal generated by the given polynomials with exact coefficients by numerical arithmetic (e.g. floating-point arithmetic). The second kind is for the given polynomials with inexact coefficients having a priori errors. In this case, we have to operate with a priori errors whether we compute a basis by exact arithmetic or not. For example, Shirayanagi's method ([3], [4]) by stabilization techniques requires to extend the input precision up to a point that the algorithm can work stably hence it is for the first kind since we cannot extend the input precision of inexact data in practice even if we can extend precisions during computations. For practical computations, coefficients may have a priori errors due to limited accuracy, representational error, measuring error and so on, hence the second kind is much more important than the first one. In this paper, we try to interpret the second kind of problem with the comprehensive Gröbner system and numerical linear algebra.

We assume that we compute a Gröbner basis or its variants for the ideal $I \subseteq \mathbf{C}[\boldsymbol{x}]$ generated by a polynomial set $F = \{f_1, \dots, f_k\} \subset \mathbf{C}[\boldsymbol{x}]$ where $\mathbf{C}[\boldsymbol{x}]$ is the polynomial ring in variables $\boldsymbol{x} = x_1, \dots, x_\ell$ over the complex number field $\mathbf{C}$. However, in our setting, coefficients may have a priori errors hence we have only a polynomial set $\tilde{F} = \{\tilde{f}_1, \dots, \tilde{f}_{\tilde{k}}\} \subset \mathbf{C}[\boldsymbol{x}]$ as the given inexact input, which may be different from F. We note that the number of polynomials may be also different (i.e. $k \neq \tilde{k}$). The most interesting part of this problem is what we should compute for the inexact input $\tilde{F}$ when we are not able to discover the hidden and desirable polynomial set F. We review some known interpretation of this problem in Section **2** and **3** and give another resolution in the latter sections.

2 Comprehensive Gröbner System with Inexact Input

If we can bound the difference between F and $\tilde{F}$ in some way, the most faithful solution for computing a Gröbner basis with inexact input is the comprehensive Gröbner basis (or comprehensive Gröbner system) introduced by Weispfenning ([8], [7], [9]). By representing error parts as unknown parameters, the problem becomes computing a parametric Gröbner basis. In this section, we briefly review this approach in our problem setting.

Let $\mathcal{A} = \mathbf{C}[\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\gamma]$ be the polynomial ring in parameters $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\gamma$ over the complex number field and consider the polynomial ring $\mathcal{A}[\boldsymbol{x}]$ in variables $x_1, \ldots, x_\ell$. For a fixed term order $\succ$ on $\mathbf{C}[\boldsymbol{x}]$, it is well-known that in general a Gröbner basis in $\mathcal{A}[\boldsymbol{x}]$ with respect to variables $\boldsymbol{x}$ will no longer remain a Gröbner basis in $\mathbf{C}[\boldsymbol{x}]$ when the parameters $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\gamma$ are specialized to some values in $\mathbf{C}$. The comprehensive Gröbner basis and system [8] are defined to overcome this situation.

Definition 1 (Comprehensive Gröbner Basis). *Let $F \subseteq \mathcal{A}[\boldsymbol{x}]$ be a finite parametric polynomial set and I be the ideal generated by F. We call a finite ideal basis G of I a comprehensive Gröbner basis of I if G is a Gröbner basis of the ideal generated by F in $\mathbf{C}[\boldsymbol{x}]$ for every specialization of parameters $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\gamma$ in $\mathbf{C}$.* $\triangleleft$

Definition 2 (Comprehensive Gröbner System). *Let $F \subseteq \mathcal{A}[\boldsymbol{x}]$ be a finite parametric polynomial set, S be a subset of $\mathbf{C}^\gamma$, $\mathcal{A}_1, \ldots, \mathcal{A}_r$ be algebraically constructible subsets of $\mathbf{C}^\gamma$ such that $S \subseteq \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_r$ and $G_1, \ldots, G_r$ be subsets of $\mathcal{A}[\boldsymbol{x}]$. We call a finite set $G = \{(\mathcal{A}_1, G_1), \ldots, (\mathcal{A}_r, G_r)\}$ of pairs a comprehensive Gröbner system for F on S if G_i is a Gröbner basis of the ideal generated by F in $\mathbf{C}[\boldsymbol{x}]$ for every specialization of parameters $(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\gamma)$ in $\mathcal{A}_i$. Each $(\mathcal{A}_i, G_i)$ is called a segment of G.* $\triangleleft$

Suppose that all the inexact parts on coefficients in $\tilde{F}$ can be represented by parameters $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\gamma$. Then, computing a Gröbner basis with inexact input can be done by computing a comprehensive Gröbner system for $F \in \mathcal{A}[\boldsymbol{x}]$ on S where S includes all the possible specialization of parameters $(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\gamma)$ in $\mathbf{C}^\gamma$. However, in general, a comprehensive Gröbner system has a huge number of segments and its computation time is quite slow (see [10] for example). Though Weispfenning [7] tried to decrease the time-complexity by using only a single parameter to represent the inexact parts, whose bounding error mechanism is very similar to interval arithmetic and Traverso and Zanoni [6] pointed out that an interval easily becomes too large when we compute a Gröbner basis by interval arithmetic. In the author's opinion, this is one of reasons that many researchers still have been studying Gröbner basis with inexact input.

3 Approximate Gröbner Basis with Inexact Input

As in the previous section, unfortunately, treating inexact parts of coefficients as parameters does not give us any reasonable (w.r.t. computation time and number of segments) answer to the second kind of problem. In this section, we review another approach by Sasaki and Kako [1]. They tried to define approximate Gröbner basis by the following approximate-zero tests for polynomials appearing in the Buchberger algorithm. We note that they also introduced several numerical techniques to prevent cancellation errors and we briefly review only their concept without their complete settings and definitions.

Definition 3 (Approximate-Zero Test). *Let $p(\boldsymbol{x})$ be a polynomial appearing in the Buchberger algorithm, and $(s_1(\boldsymbol{x}),\dots,s_{\tilde{k}}(\boldsymbol{x}))$ be the syzygy for $p(\boldsymbol{x})$ satisfying $p(\boldsymbol{x})=\sum_{i=1}^{\tilde{k}} s_i(\boldsymbol{x})\tilde{f}_i(\boldsymbol{x})$. If $\|p\|<\varepsilon\times\max\{\|s_1\tilde{f}_1\|,\dots,\|s_{\tilde{k}}\tilde{f}_{\tilde{k}}\|\}$ where $\|p\|$ denote the infinity norm of $p(\boldsymbol{x})$, then we say $p(\boldsymbol{x})$ is approximately zero at tolerance ε, and we denote this as $p(\boldsymbol{x})\equiv 0\ (tol\,\varepsilon)$.* $\triangleleft$

Definition 4 (Practical Approximate-Zero Test). *Let $p(\boldsymbol{x})$ be a polynomial appearing in the Buchberger algorithm, and $(p_1,\dots,p_m)$ be all the non-zero coefficients tuple of $p(\boldsymbol{x})$. If $\max\{|p_1|,\dots,|p_m|\}<\varepsilon$, then we say $p(\boldsymbol{x})$ is practically approximate-zero at tolerance ε, and we denote this as $p(\boldsymbol{x})\equiv 0\ (tol\ \varepsilon)$.* $\triangleleft$

With one of the above definitions (computation of syzygies is time-consuming, so they decided to use the second one in practice), they define the following approximate Gröbner basis.

Definition 5 (Approximate Gröbner Basis). *Let ε be a small positive number, and $G=\{g_1,\dots,g_r\}$ be a polynomial set. We call G an approximate Gröbner basis of tolerance ε, if we have $\overline{\mathrm{S}(g_i,g_j)}^G\equiv 0\ (tol\ \varepsilon)\quad(\forall i\neq j)$ where $\mathrm{S}(g_i,g_j)$ and $\overline{p}^G$ denote the S-polynomial of g_i and g_j and the normal form of p by G, respectively.* $\triangleleft$

The above definition can be considered as a numerical version of comprehensive Gröbner system with a single parameter by Weispfenning [7], using much reasonably relaxed bounds instead of exact interval arithmetic. In the Buchberger algorithm, head terms of polynomials appearing in the procedure are critically important hence most of known results have to take care of approximate zero tests by exact interval arithmetic, parametric representation or the above way for example. In the rest of the paper, we consider the second kind of problem as a problem in numerical linear algebra instead of trying to extend the Buchberger algorithm directly.

4 Gröbner Basis for Inexact Input as Linear Space

We note again that the first and second kinds of problem are fundamentally different. For the first kind, there exists the answer which is a Gröbner basis of the ideal I generated by F and can be computable by exact arithmetic. On the other hand, for the second one, there exist so many possible answers since F is not known in practice and the given polynomials of $\tilde{F}$ have a priori errors and we can absolutely not be able to know that they should be. Moreover, for the given $\tilde{F}$ and the unknown F, it may happen that $p(\boldsymbol{x}) \in \text{ideal}(G)$ and $p(\boldsymbol{x}) \notin \text{ideal}(F)$ even if we can compute a Gröbner basis G for ideal$(\tilde{F})$ by some method, where ideal(S) denotes the ideal generated by the elements of a set S. Because such a Gröbner basis is only a candidate for possible so many Gröbner bases for unknown F . It also be possible that they include $\{1\}$. Any resolution for the second kind of problem must guarantee that $p(\boldsymbol{x}) \in \text{ideal}(G)$ and $p(\boldsymbol{x}) \in \text{ideal}(\tilde{F})$ are equivalent with or without some conditions since what is the most reliable is not G but the given $\tilde{F}$ (this is the only reliable information) which does not have any posteriori error. In the below, we give a resolution from this point of view.

4.1 Gröbner Basis as Linear Space

Some researchers studied computing a Gröbner basis by reduced row echelon form ([11], [12]) though there are no concrete algorithms described. However, this is not efficient since we have to operate with large matrices. Using matrix operations partially like F4 and F5 ([13], [14], [2]) may be the best choice if we want to decrease the computation time. We note that the matrix constructed in the F4 algorithm is essentially the same as in this paper and is more compact and well considered. On the other hand, for the second kind of problem, it may be useful since we can use so many results from numerical linear algebra for the situation where we must inevitably operate with a priori errors. Hence we summarize an algorithm for computing Gröbner basis with exact input by reduced row echelon form in this subsection. We note that we use the following definition though there are several equivalents (see [15] or other text books).

Definition 6 (Gröbner Basis). *$G = \{g_1, \ldots, g_r\} \subseteq I \setminus \{0\}$ is a Gröbner basis for I w.r.t. a fixed term order $\succ$ if for any $f \in I \setminus \{0\}$, there exists $g_i \in G$ such that $\text{ht}(g_i) | \text{ht}(f)$ where $\text{ht}(p)$ denotes the head term of $p(\boldsymbol{x}) \in \mathbf{C}[\boldsymbol{x}]$ w.r.t. $\succ$.* $\triangleleft$

We consider the linear map $\phi_{\mathcal{T}} : \mathbf{C}[\boldsymbol{x}]_{\mathcal{T}} \to \mathbf{C}^m$ such that $\phi_{\mathcal{T}}(t_i) = \overrightarrow{e_i}$ where $\mathbf{C}[\boldsymbol{x}]_{\mathcal{T}}$ is the submodule of $\mathbf{C}[\boldsymbol{x}]$ generated by an ordered set (the most left element is the highest) of terms $\mathcal{T} = \{t_1, \ldots, t_m\}_{\succ}$ and $\overrightarrow{e_i} (i = 1, \ldots, m)$ denotes the canonical basis of $\mathbf{C}^m$. The coefficient vector $\overrightarrow{p}$ of $p(\boldsymbol{x}) \in \mathbf{C}[\boldsymbol{x}]$ is defined to be satisfying $\overrightarrow{p} = \phi_{\mathcal{T}}(p)$ and $p(\boldsymbol{x}) = \phi_{\mathcal{T}}^{-1}(\overrightarrow{p})$. With a fixed $\mathcal{T}$, we consider the following subset $F_{\mathcal{T}}$ of I.

$$F_{\mathcal{T}} = \left\{ \sum_{i=1}^{k} s_i(\boldsymbol{x}) f_i(\boldsymbol{x}) \mid s_i(\boldsymbol{x}) f_i(\boldsymbol{x}) \in \mathbf{C}[\boldsymbol{x}]_{\mathcal{T}},\ s_i(\boldsymbol{x}) \in \mathbf{C}[\boldsymbol{x}] \right\}.$$

The Buchberger algorithm guarantees that $G \subseteq F_{\mathcal{T}}$ if $\mathcal{T}$ has a large enough number of elements. To compute a Gröbner basis for I, we construct the matrix $\mathcal{M}_{\mathcal{T}}(F)$ whose each row vector $\overrightarrow{p}$ satisfies $\phi_{\mathcal{T}}^{-1}(\overrightarrow{p}) \in \mathcal{P}_{\mathcal{T}}(f)$ for $f(\boldsymbol{x}) \in F$ where

$$\mathcal{P}_{\mathcal{T}}(p) = \{t_i \times p(\boldsymbol{x}) \in \mathbf{C}[\boldsymbol{x}]_{\mathcal{T}} \mid t_i = \phi_{\mathcal{T}}^{-1}(\overrightarrow{e_i}),\ i = 1, \ldots, m\}.$$

By this definition, $F_\mathcal{T}$ and the linear space $\mathcal{V}_\mathcal{T}$ generated by the row vectors of $\mathcal{M}_\mathcal{T}(F)$ are isomorphic.

We note that a matrix is said to be in reduced row echelon form if it satisfies the following four conditions.

1. All nonzero rows appear above zero rows.
2. Each leading element of a row is in a column to the right of the leading element of the row above it.
3. The leading element in any nonzero row is 1.
4. Every leading element is the only nonzero element in its column.

Lemma 1. *Let $\overline{\mathcal{M}_\mathcal{T}(F)}$ be the reduced row echelon form of $\mathcal{M}_\mathcal{T}(F)$. If $g_i(\boldsymbol{x}) \in F_\mathcal{T}$ for a fixed $i \in \{1,\ldots,r\}$, $\overline{\mathcal{M}_\mathcal{T}(F)}$ has a row vector $\overrightarrow{p}$ satisfying $\mathrm{ht}(g_i) = \mathrm{ht}(\phi_\mathcal{T}^{-1}(\overrightarrow{p}))$.* ◁

Proof. Since the linear map $\phi_\mathcal{T}$ is defined by the ordered set $\mathcal{T}$, each leading element of a row vector $\overrightarrow{p}$ of $\overline{\mathcal{M}_\mathcal{T}(F)}$ is corresponding to $\mathrm{ht}(\phi_\mathcal{T}^{-1}(\overrightarrow{p}))$. The lemma follows from the facts that $F_\mathcal{T}$ and $\mathcal{V}_\mathcal{T}$ are isomorphic and all the leading entries of nonzero rows are disjoints since $\overline{\mathcal{M}_\mathcal{T}(F)}$ is in the reduced row echelon form. □

Lemma 2. *Let $\overline{\mathcal{M}_\mathcal{T}(F)}$ be the reduced row echelon form of $\mathcal{M}_\mathcal{T}(F)$. If $\mathcal{T}$ has a large enough number of elements, the following $G_\mathcal{T}$ is a Gröbner basis for I.*

$$G_\mathcal{T} = \left\{ \phi_\mathcal{T}^{-1}(\overrightarrow{p}) \mid \overrightarrow{p} \textit{ is a row vector of } \overline{\mathcal{M}_\mathcal{T}(F)} \right\}.$$

◁

Proof. The Buchberger algorithm guarantees that $G \subseteq F_\mathcal{T}$ if $\mathcal{T}$ has a large enough number of elements. Therefore, $G_\mathcal{T}$ satisfies the condition of Definition 6 since we have $g_i(\boldsymbol{x}) \in G_\mathcal{T}$, $i = \{1,\ldots,r\}$ by Lemma 1. □

The above lemmas lead us to the following algorithm directly.

Algorithm 1. (Gröbner Basis by Row Echelon Form)
Input: a term order $\succ$ and a set F of polynomials,
$$F = \{f_1(\boldsymbol{x}),\ldots,f_k(\boldsymbol{x})\} \subset \mathbf{C}[\boldsymbol{x}].$$
Output: a Gröbner basis G for the ideal generated by F,
$$G = \{g_1(\boldsymbol{x}),\ldots,g_r(\boldsymbol{x})\} \subset \mathbf{C}[\boldsymbol{x}].$$
1. $d \leftarrow \max_{i=1,\ldots,k} \mathrm{tdeg}(f_i)$ (the total degree of $f_i(\boldsymbol{x})$).
2. $\mathcal{T} \leftarrow$ the ordered set of the terms of total degrees $\leq d$.
3. $\overline{\mathcal{M}_\mathcal{T}(F)} \leftarrow$ the reduced row echelon form of $\mathcal{M}_\mathcal{T}(F)$.
4. $G_\mathcal{T} \leftarrow \left\{ \phi_\mathcal{T}^{-1}(\overrightarrow{p}) \mid \overrightarrow{p} \text{ is a row vector of } \overline{\mathcal{M}_\mathcal{T}(F)} \right\}$.
5. $G \leftarrow G_\mathcal{T} \setminus \{ g \in G_\mathcal{T} | \exists h \in G_\mathcal{T} \setminus \{g\} \text{ s.t. } \mathrm{ht}(h) | \mathrm{ht}(g) \}$.
6. Outputs G if the following conditions satisfied:
   6-1. $\forall f \in F$, $\overline{f_i}^G = 0$,
   6-2. $\forall g_i, g_j \in G$, $\overline{\mathrm{S}(g_i, g_j)}^G = 0$,

otherwise $d \leftarrow d+1$ and goto Step 2. ◁

Algorithm 1 is not optimized. For example, we should optimize the algorithm as follows. In Step 1, it is better that we start with a larger d (e.g. $\max_{i=1,\ldots,k} \operatorname{tdeg}(f_i) + 1$ or a large enough d such that all the S-polynomials of F can be calculated in $\mathbf{C}[\boldsymbol{x}]_{\mathcal{T}}$). Moreover, we can use the rectangular degree (bounding each variable separately and also called the multi degree) instead of the total degree. In Step 6, it is better that we increment d by Δ_d such that $\mathrm{S}(g_i, g_j)$ can be calculated in $F_{\mathcal{T}}$ for any pair of elements of G and $\mathcal{T}$ with $d \leftarrow d + \Delta_d$.

Lemma 3. *Algorithm 1 computes the reduced Gröbner basis for the ideal generated by the given polynomial set F.* ◁

Proof. The condition 6-1 guarantees that the ideals generated by F and G are the same. Hence, if $\mathcal{T}$ has a large enough number of elements, Algorithm 1 outputs a Gröbner basis for the ideal generated by F since the condition 6-2 means that G is a Gröbner basis for the ideal generated by G. Step 5 deletes verbose polynomials by Definition 6 hence G is a minimal Gröbner basis. The lemma follows from the fact that $\overline{\mathcal{M}_{\mathcal{T}}(F)}$ is in the reduced row echelon form so that all the polynomials corresponding to row vectors are already reduced by other rows (polynomials). In this algorithm, we use total degree bounds for $\mathcal{T}$ hence $\mathcal{T}$ must have a large enough number of elements in finite steps. □

Example 1. We compute the reduced Gröbner basis w.r.t. the graded lexicographic order for the ideal generated by the following polynomials. We note that we show only very simple example since it is difficult to show the whole matrices for nontrivial cases.

$$F = \{2x + 3y,\ xy - 2\}.$$

In this case, we construct the following matrix $\mathcal{M}_{\mathcal{T}}(F)$ with $d = 3$ and compute its reduced row echelon form $\overline{\mathcal{M}_{\mathcal{T}}(F)}$.

$$\mathcal{M}_{\mathcal{T}}(F) = \begin{pmatrix} 2&3&0&0&0&0&0&0&0&0 \\ 0&2&3&0&0&0&0&0&0&0 \\ 0&0&0&0&2&3&0&0&0&0 \\ 0&0&2&3&0&0&0&0&0&0 \\ 0&0&0&0&0&2&3&0&0&0 \\ 0&0&0&0&0&0&0&2&3&0 \\ 0&1&0&0&0&0&0&-2&0&0 \\ 0&0&1&0&0&0&0&0&-2&0 \\ 0&0&0&0&0&1&0&0&0&-2 \end{pmatrix},\ \overline{\mathcal{M}_{\mathcal{T}}(F)} = \begin{pmatrix} 1&0&0&0&0&0&0&0&-\frac{9}{2}&0 \\ 0&1&0&0&0&0&0&0&3&0 \\ 0&0&1&0&0&0&0&0&-2&0 \\ 0&0&0&1&0&0&0&0&\frac{4}{3}&0 \\ 0&0&0&0&1&0&0&0&0&3 \\ 0&0&0&0&0&1&0&0&0&-2 \\ 0&0&0&0&0&0&1&0&0&\frac{4}{3} \\ 0&0&0&0&0&0&0&1&\frac{3}{2}&0 \end{pmatrix}.$$

Hence, we have the following candidate $G_{\mathcal{T}}$ for a Gröbner basis.

$$\left\{x^3 - \frac{9y}{2},\ yx^2 + 3y,\ xy^2 - 2y,\ y^3 + \frac{4y}{3}, x^2 + 3,\ xy - 2,\ y^2 + \frac{4}{3},\ x + \frac{3y}{2}\right\}.$$

We delete all the verbose elements and test the conditions in Step 6. Since they pass the conditions, we obtain the following reduced Gröbner basis.

$$G = \left\{x + \frac{3y}{2},\ y^2 + \frac{4}{3}\right\}.$$

$\triangleleft$

4.2 Definition of Numerical Gröbner Basis as Linear Space

Let $\mathcal{M}_{\mathcal{T}}(\tilde{F},p)$ be the matrix whose row vectors are of $\mathcal{M}_{\mathcal{T}}(\tilde{F})$ and $\phi_{\mathcal{T}}(p)$ of a polynomial $p(\boldsymbol{x})$. We denote the numerical rank of matrix M by $\mathrm{rank}_\varepsilon(M)$ which satisfies

$$\mathrm{rank}_\varepsilon(M) = \min_{\|M-M'\|_2\leq\varepsilon} \mathrm{rank}(M')$$

where $\mathrm{rank}(M)$ denotes the conventional matrix rank of M. We note that for any $\kappa < \mathrm{rank}(M)$, we have

$$\min_{\mathrm{rank}(M')=\kappa} \|M - M'\|_2 = \sigma_{\kappa+1}$$

where σ_i denotes the i-th largest singular value of M.

The difference of the ideal membership of $p(\boldsymbol{x})$, between $\mathrm{ideal}(G) \supseteq \tilde{F}$ and $\mathrm{ideal}(F)$ may increase with increasing the total degree or the number of terms of $p(\boldsymbol{x})$. Hence, we consider the equivalence of $\mathrm{ideal}(G)$ and $\mathrm{ideal}(\tilde{F})$ by limiting the total degree or the number of terms that must be the lowest value satisfying $G \subset \tilde{F}_{\mathcal{T}}$ since we wish to keep the relations between G and $\tilde{F}$. We note again that $\tilde{F}$ is only reliable since F is not known.

Definition 7 (Numerical Membership). *For a polynomial $p(\boldsymbol{x})$, a polynomial set $\tilde{F}$ and an ordered set of terms $\mathcal{T}$, we say that $p(\boldsymbol{x})$ is numerically a member of* $\mathrm{ideal}(\tilde{F})$ *w.r.t.* $\mathcal{T}$ *and the tolerance* ε *if* $\mathrm{rank}(\mathcal{M}_{\mathcal{T}}(\tilde{F})) = \mathrm{rank}_\varepsilon(\mathcal{M}_{\mathcal{T}}(\tilde{F},p))$. *We denote this by* $p(\boldsymbol{x}) \in_{\mathcal{T},\varepsilon} \mathrm{ideal}(\tilde{F})$. $\triangleleft$

By this definition, we say $\mathrm{ideal}(\tilde{F})$ and $\mathrm{ideal}(G)$ are numerically equivalent if and only if $\forall f(\boldsymbol{x}) \in \tilde{F},\ f(\boldsymbol{x}) \in_{\mathcal{T},\varepsilon} \mathrm{ideal}(G)$ and $\forall g(\boldsymbol{x}) \in G,\ g(\boldsymbol{x}) \in_{\mathcal{T},\varepsilon} \mathrm{ideal}(\tilde{F})$. On may think that with this definition some strange situations can happen. For example, it is possible that every polynomials numerically belong to an ideal or that $s_1f_1 + s_2f_2$ does not numerically belong to an ideal even if f_1 and f_2 numerically belong to it. This is correct and inevitable for the second kind of problem. $\tilde{F}$ are just one of possible sets for F so we cannot ignore the extreme case: $1 \in \mathrm{ideal}(F)$. Moreover, even if we use exact arithmetic as in Section 2, after any computation (e.g. $s_1f_1 + s_2f_2$), the difference from F usually becomes larger hence some strange situations may happen.

The above definition cannot be used for testing $\overline{\mathrm{S}(g_i,g_j)}^G = 0$ $(g_i, g_j \in G)$ since it usually happens that $\mathrm{S}(g_i,g_j) \in_{\mathcal{T},\varepsilon} \mathrm{ideal}(G)$, depending on $\mathcal{T}$. We suppose $g_j(\boldsymbol{x}) \succ g_i(\boldsymbol{x})$ $(j < i)$ and construct the matrix $\mathcal{R}_{\mathcal{T}}(G)$ whose each row vector $\overrightarrow{p}$ satisfies $\phi_{\mathcal{T}}^{-1}(\overrightarrow{p}) \in \mathcal{P}_{\mathcal{T}}(g_i)$ for $g_i(\boldsymbol{x}) \in G$ where

$$\begin{aligned}\mathcal{P}_{\mathcal{T}}(g_i) = \{t_i \times g_i \in \mathbf{C}[\boldsymbol{x}]_{\mathcal{T}} \mid t_i = \phi_{\mathcal{T}}^{-1}(\overrightarrow{e_i}),\ i = 1,\ldots,m,\\ !\exists g \in \mathcal{P}_{\mathcal{T}}(g_j)\ (j < i),\ \mathrm{ht}(g) = \mathrm{ht}(t_i \times g_i)\}.\end{aligned}$$

Similar to $\mathcal{M}_{\mathcal{T}}(\tilde{F},p)$, $\mathcal{R}_{\mathcal{T}}(G,p)$ is defined as the matrix whose row vectors are the vectors of $\mathcal{R}_{\mathcal{T}}(G)$ and $\phi_{\mathcal{T}}(p)$ of a polynomial $p(\boldsymbol{x})$.

Definition 8 (Numerical S-Polynomial Check). *For polynomials $g_i(\boldsymbol{x})$ and $g_j(\boldsymbol{x})$ of a set G and an ordered set of terms $\mathcal{T}$, we say that the S-polynomial* $\mathrm{S}(g_i, g_j)$ *is numerically reduced to 0 by G w.r.t. $\mathcal{T}$ and the tolerance $\varepsilon \in \mathbf{R}_{\geq 0}$ if* $\mathrm{rank}(\mathcal{R}_{\mathcal{T}}(G)) = \mathrm{rank}_{\varepsilon}(\mathcal{R}_{\mathcal{T}}(G, \mathrm{S}(g_i, g_j)))$. *We denote it by* $\overline{\mathrm{S}(g_i, g_j)}^{G} =_{\mathcal{T},\varepsilon} 0$. ◁

Definition 9 (Numerical Gröbner Basis). *We say that $G = \{g_1, \ldots, g_r\}$ is a numerical Gröbner basis for* $\mathrm{ideal}(\tilde{F})$ *w.r.t. a fixed term order $\succ$ and a tolerance $\varepsilon \in \mathbf{R}_{\geq 0}$ if the following conditions are satisfied.*
1. $\forall i, j \in \{1, \ldots, r\}$, $\mathrm{lcm}(\mathrm{ht}(g_i), \mathrm{ht}(g_j)) \in \mathcal{T}$,
2. $\forall i, j \in \{1, \ldots, r\}$, $\overline{\mathrm{S}(g_i, g_j)}^{G} =_{\mathcal{T},\varepsilon} 0$
where $\mathcal{T}$ is an ordered set of terms such that $\mathrm{ideal}(\tilde{F})$ *and* $\mathrm{ideal}(G)$ *are numerically equivalent. In addition, minimal and reduced Gröbner basis are also defined in the ordinary way.* ◁

We note that the above definition is compatible with the conventional Gröbner basis since they are the same if $\varepsilon = 0$. Moreover, any conventional Gröbner basis is always a numerical Gröbner basis w.r.t. any tolerance. One may think that this definition for the second kind of problem is not well-posed which is the notion introduced by Hadamard and should have three properties: a solution exists, is unique, and continuously depends on the data. Analyzing the definition from this point of view is postponed for future work.

4.3 How to Compute Numerical Gröbner Basis

Computing a numerical Gröbner basis defined in the previous subsection is not easy. In this subsection, we give a naive method using the reduced row echelon form. Though Algorithm 1 uses only the reduced row echelon form, for the numerical case, we separate it into the forward Gaussian elimination and back-substitution. Let $\mathcal{U}_{\mathcal{T}}(\tilde{F})$ be the upper triangular matrix by the forward Gaussian elimination with partial pivoting, using an unitary transformation (i.e. givens rotation), of $\mathcal{M}_{\mathcal{T}}(\tilde{F})$, and $\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})$ be the same matrix but neglecting elements and rows that are smaller than the given tolerance ε in absolute value and 2-norm, respectively.

Algorithm 2. (Numerical Gröbner Basis)
Input: a tolerance $\varepsilon \ll 1$, a term order $\succ$ and a set $\tilde{F}$,
$$\tilde{F} = \{f_1(\boldsymbol{x}), \ldots, f_{\tilde{k}}(\boldsymbol{x})\} \subset \mathbf{C}[\boldsymbol{x}].$$
Output: a numerical Gröbner basis G for $\mathrm{ideal}(\tilde{F})$,
$$G = \{g_1(\boldsymbol{x}), \ldots, g_r(\boldsymbol{x})\} \subset \mathbf{C}[\boldsymbol{x}] \text{ or “failed”}.$$
1. $d \leftarrow \max_{i=1,\ldots,n} \mathrm{tdeg}(f_i)$ and $e \leftarrow 1$.
2. $\mathcal{T} \leftarrow$ the ordered set of the terms of total degrees $\leq d$.
3. $\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}) \leftarrow$ the upper triangular matrix by the forward Gaussian elimination with partial pivoting, using an unitary transformation of $\mathcal{M}_{\mathcal{T}}(\tilde{F})$.
4. $\overline{\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})} \leftarrow$ the reduced row echelon form of $\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})$ by back-substitution without scaling pivots to one.

5. $G_{\mathcal{T}} \leftarrow \left\{ \phi_{\mathcal{T}}^{-1}(\overrightarrow{p}) \mid \overrightarrow{p} \text{ is a row of } \mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}),\ \|\overrightarrow{p}\|_2 > \varepsilon \right\}$.
   $\overline{G_{\mathcal{T}}} \leftarrow \left\{ \phi_{\mathcal{T}}^{-1}(\overrightarrow{p}) \mid \overrightarrow{p} \text{ is a row of } \overline{\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})},\ \|\overrightarrow{p}\|_2 > \varepsilon \right\}$.
6. $\bar{G} \leftarrow G_{\mathcal{T}} \setminus \{ g \in G_{\mathcal{T}} | \exists h \in G_{\mathcal{T}} \setminus \{g\} \text{ s.t. } \mathrm{ht}(h) | \mathrm{ht}(g) \}$.
   $G \leftarrow \overline{G_{\mathcal{T}}} \setminus \{ g \in \overline{G_{\mathcal{T}}} | \exists h \in \overline{G_{\mathcal{T}}} \setminus \{g\} \text{ s.t. } \mathrm{ht}(h) | \mathrm{ht}(g) \}$.
7. Outputs G or $\bar{G}$ whichever satisfies the conditions:
   7-1. $\forall g_i, g_j \in G,\ \mathrm{lcm}(\mathrm{ht}(g_i), \mathrm{ht}(g_j)) \in \mathcal{T}$,
   7-2. $\forall f \in \tilde{F},\ f(\boldsymbol{x}) \in_{\mathcal{T},\varepsilon} \mathrm{ideal}(G)$,
   7-3. $\forall g_i, g_j \in G,\ \overline{\mathrm{S}(g_i, g_j)}^G =_{\mathcal{T},\varepsilon} 0$.
8. Outputs "failed" if $3^e \varepsilon \geq 1$.
9. $d \leftarrow d+1$, $e \leftarrow e+1$ and goto Step 2. ◁

Lemma 4. *Throughout Algorithm 2, we have*

$$\forall g \in G_{\mathcal{T}}\ (\supseteq \bar{G}),\ g(\boldsymbol{x}) \in_{\mathcal{T},\delta} \mathrm{ideal}(\tilde{F})$$

where $\delta = \|\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}) - \mathcal{U}_{\mathcal{T}}(\tilde{F})\|$. ◁

Proof. Let $\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}, g)$ be the matrix whose row vectors are of $\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})$ and $\phi_{\mathcal{T}}(g)$, and $\mathcal{U}_{\mathcal{T}}(\tilde{F}, g)$ be the matrix whose row vectors are of $\mathcal{U}_{\mathcal{T}}(\tilde{F})$ and $\phi_{\mathcal{T}}(g)$. By the assumption of the lemma and $\mathrm{rank}(\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}, g)) = \mathrm{rank}(\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}))$, we have $\| \mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}, g) - \mathcal{U}_{\mathcal{T}}(\tilde{F}, g) \|_2 \leq \delta$. Since $\mathcal{U}_{\mathcal{T}}(\tilde{F})$ is calculated by only unitary transformations, we have $\mathcal{U}_{\mathcal{T}}(\tilde{F}) = U \mathcal{M}_{\mathcal{T}}(\tilde{F})$ where U denotes the product of such transformations. Let U' be the following unitary matrix satisfying $\mathcal{U}_{\mathcal{T}}(\tilde{F}, g) = U' \mathcal{M}_{\mathcal{T}}(\tilde{F}, g)$.

$$U' = \begin{pmatrix} & & 0 \\ & U & \vdots \\ & & 0 \\ 0 & \cdots\ 0 & 1 \end{pmatrix}.$$

The lemma follows from the facts that all the singular values of $\mathcal{M}_{\mathcal{T}}(\tilde{F}, g)$ and $U' \mathcal{M}_{\mathcal{T}}(\tilde{F}, g)$ are the same since U' is unitary. □

Lemma 5. *Throughout Algorithm 2, we have*

$$\forall g \in \overline{G_{\mathcal{T}}}\ (\supseteq G),\ g(\boldsymbol{x}) \in_{\mathcal{T},\delta} \mathrm{ideal}(\tilde{F})$$

where $\delta = \|\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}) - \mathcal{U}_{\mathcal{T}}(\tilde{F})\|$. ◁

Proof. Since for any row vector $\overrightarrow{p}$ of $\overline{\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})}$, $\overrightarrow{p}$ is a linear combination of row vectors of $\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})$, we have $\mathrm{rank}(\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}, g)) = \mathrm{rank}(\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}))$. The lemma is proved by the same way in the above proof. □

Lemma 6. *Throughout Algorithm 2, we have*

$$\forall f \in \tilde{F}_{\mathcal{T}},\ f(\boldsymbol{x}) \in_{\mathcal{T},\delta} \mathrm{ideal}(G_{\mathcal{T}})$$

where $\delta = \|\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}) - \mathcal{U}_{\mathcal{T}}(\tilde{F})\|$. ◁

Proof. The lemma follows from the fact $\|\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F},f) - \mathcal{U}_{\mathcal{T}}(\tilde{F},f)\|_2 \leq \delta$ as in the above proves. □

Unfortunately, the above lemmas do not guarantee that Algorithm 2 always terminates with a numerical Gröbner basis. However, they suggest $3^e\varepsilon \geq 1$ in Step 8 as follows. One of the reasons that Algorithm 2 can fail to terminate with a numerical Gröbner basis is $\exists g \in \bar{G},\ t \in \mathcal{T},\ tg \notin_{\mathcal{T},\varepsilon} G_{\mathcal{T}}$. For a proper superset $\mathcal{T}'$ of $\mathcal{T}$, by the above lemmas, we have

$$\begin{aligned}
&\| \mathcal{U}_{\mathcal{T}',\varepsilon}(\tilde{F},tg) - \mathcal{U}_{\mathcal{T}',\varepsilon}(\tilde{F}) \|_2 \\
&= \| \mathcal{U}_{\mathcal{T}',\varepsilon}(\tilde{F},tg) - \mathcal{U}_{\mathcal{T}'}(\tilde{F},tg) + \mathcal{U}_{\mathcal{T}'}(\tilde{F},tg) - \mathcal{U}_{\mathcal{T}'}(\tilde{F}) + \mathcal{U}_{\mathcal{T}'}(\tilde{F}) - \mathcal{U}_{\mathcal{T}',\varepsilon}(\tilde{F}) \|_2 \\
&\leq \| \mathcal{U}_{\mathcal{T}',\varepsilon}(\tilde{F},tg) - \mathcal{U}_{\mathcal{T}'}(\tilde{F},tg) \|_2 + \| \mathcal{U}_{\mathcal{T}'}(\tilde{F},tg) - \mathcal{U}_{\mathcal{T}'}(\tilde{F}) \|_2 \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad + \| \mathcal{U}_{\mathcal{T}'}(\tilde{F}) - \mathcal{U}_{\mathcal{T}',\varepsilon}(\tilde{F}) \|_2 \\
&\leq 3\delta'
\end{aligned}$$

where $\delta' = \|\mathcal{U}_{\mathcal{T}',\varepsilon}(\tilde{F}) - \mathcal{U}_{\mathcal{T}'}(\tilde{F})\|$. This means that the distance between $\bar{G}$ and $G_{\mathcal{T}}$ increases by a factor of 3 in the worst case, even if we decrease δ and δ' such that $\delta,\ \delta' \approx \varepsilon$.

In our preliminary implementation, due to accumulating numerical errors, we use the following $G_{\mathcal{T}}$ and $\overline{G_{\mathcal{T}}}$ instead of the above.

$$G_{\mathcal{T}} \leftarrow \left\{ \phi_{\mathcal{T}}^{-1}(\overrightarrow{p}) \mid \overrightarrow{p} \text{ is a row of } \mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F}),\ \|\overrightarrow{p}\|_2 > \varepsilon^{1/2} \right\},$$

$$\overline{G_{\mathcal{T}}} \leftarrow \left\{ \phi_{\mathcal{T}}^{-1}(\overrightarrow{p}) \mid \overrightarrow{p} \text{ is a row of } \overline{\mathcal{U}_{\mathcal{T},\varepsilon}(\tilde{F})},\ \|\overrightarrow{p}\|_2 > \varepsilon^{1/2} \right\}.$$

In Step 7, we test G and $\bar{G}$. However, it is better that we test the all subset of $G_{\mathcal{T}}$ and $\overline{G_{\mathcal{T}}}$ if we do not consider the computing time though we do not implement this. According to our experiments, we could detect a suitable tolerance ε as follows.

$$\varepsilon = 10^{(\log_{10}\sigma_k + \log_{10}\sigma_{k+1})/2} \tag{4.1}$$

where σ_i denotes the i-th largest nonzero singular value of $\mathcal{M}_{\mathcal{T}}(\tilde{F})$ and k is the largest integer maximizes σ_k/σ_{k+1}. Moreover, in our preliminary implementation, we use matrices $\mathcal{N}_{\mathcal{T}}(\tilde{F})$ and $\mathcal{N}_{\mathcal{T}}(\tilde{F},p)$ instead of $\mathcal{M}_{\mathcal{T}}(\tilde{F})$ and $\mathcal{M}_{\mathcal{T}}(\tilde{F},p)$, respectively, whose row vectors are normalized in 2-norm. This normalization is not necessary for our definition, however this makes numerical computations more stable.

Example 2. We compute a numerical Gröbner basis w.r.t. the graded lexicographic order and the tolerance $\varepsilon = 10^{-5}$ for the ideal generated by the following polynomials that are the same polynomials in Example 1 but slightly perturbed.

$$\tilde{F} = \{2.000005x + 3.000001y,\ 0.999999xy - 2.000003\}.$$

In this case, we construct the matrix $\mathcal{N}_{\mathcal{T}}(\tilde{F})$ with $d = 3$ and compute the reduced row echelon form of $\mathcal{N}_{\mathcal{T}}(\tilde{F})$. In Step 5, we have the following candidate for a numerical Gröbner basis.

$$\begin{aligned}\overline{G_{\mathcal{T}}} = \{ & 0.554701x^3 - 2.49615y,\ 0.712525yx^2 + 2.13758y,\\ & 0.883413xy^2 - 1.76683y,\ 0.647575y^3 + 0.863437y,\\ & 0.554701x^2 + 1.6641,\ 0.712525xy - 1.42505,\\ & 0.522232y^2 + 0.696312,\ 0.716116x + 1.07417y \ \ \}.\end{aligned}$$

We delete all the verbose elements and test the conditions in Step 7. Since they pass the conditions, we obtain the following numerical Gröbner basis that are very similar to the result in Example 1.

$$G = \left\{1.0y^2 + 1.33334,\ 1.0x + 1.5y\right\}.$$

For the lexicographic order and $\varepsilon = 10^{-5}$, we start with the rectangular degree bound $d = \{2, 2\}$ and we have the following $\overline{G_{\mathcal{T}}}$.

$$\begin{aligned}\overline{G_{\mathcal{T}}} = \{ & 0.447213x^2y^2 - 1.78886,\ 0.712525yx^2 + 2.13758y,\\ & 0.554701x^2 + 1.6641,\ 0.68755xy^2 - 1.3751y,\\ & 0.712525xy - 1.42505,\ 0.716116x + 1.07417y,\ 0.522232y^2 + 0.696312 \}.\end{aligned}$$

We delete all the verbose elements and test the conditions in Step 7. Since they pass the conditions, we obtain the following numerical Gröbner basis.

$$G = \left\{1.0x + 1.5y,\ 1.0y^2 + 1.33334\right\}. \tag{4.2}$$

Our method can work for the following polynomials having a small head coefficient w.r.t. the lexicographic order.

$$\tilde{F} = \{ 0.0000001x^2 + 2.000005x + 3.000001y,\ 0.999999xy - 2.000003 \}.$$

With the tolerance $\varepsilon = 6.95972 \times 10^{-9}$ calculated by (4.1) and the rectangular degree bound $d = \{5, 4\}$, we have the following numerical Gröbner basis. We note that the head term of the first element is smaller than ε during inner calculations hence it is not reduced. Moreover, Algorithm 2 outputs the same as in just above (4.2) if we specify $\varepsilon = 10^{-6}$.

$$\{ 0.000861698y^2 + 1.5y + 1.0x + 0.00114879,\ 1.0y^2 - 0.0000001y + 1.33334 \}.$$

$\triangleleft$

5 Remarks

Our approach uses a huge matrix so that it is not effective if we try to compute a Gröbner basis for polynomials with exact coefficients. However, as noted in the beginning of Section 4, it is natural that we use several tools in numerical linear algebra since we have to handle a priori errors and most of symbolic-numeric algorithms for polynomials also use them from necessity. From this point of view, instead of row echelon form by the Gaussian elimination in Algorithm 2, one can use the QR decomposition or the singular value decomposition (SVD) to improve the algorithm though we've not yet analyzed their effectiveness. We note that for all the example in this paper, we use our preliminary implementation on *Mathematica* 6.

Acknowledgements The author wishes to thank the many referees (especially the one who knows the previous version of this paper) for their useful suggestions that explicitly indicate the future work on the problem mentioned in this paper.

References

1. Sasaki, T., Kako, F.: Computing floating-point gröbner bases stably. In: SNC'07. ACM, New York (2007) 180–189
2. Kondratyev, A., Stetter, H.J., Winkler, S.: Numerical computation of gröbner bases. In: Proceedings of CASC2004 (Computer Algebra in Scientific Computing). (2004) 295–306
3. Shirayanagi, K.: An algorithm to compute floating point gröbner bases. In: Proceedings of the Maple summer workshop and symposium on Mathematical computation with Maple V : ideas and applications, Cambridge, MA, USA, Birkhauser Boston Inc. (1993) 95–106
4. Shirayanagi, K.: Floating point gröbner bases. In: Selected papers presented at the international IMACS symposium on Symbolic computation, new trends and developments, Amsterdam, The Netherlands, The Netherlands, Elsevier Science Publishers B. V. (1996) 509–528
5. Stetter, H.J.: Approximate gröbner bases – an impossible concept? In: Proceedings of SNC2005 (Symbolic-Numeric Computation). (2005) 235–236
6. Traverso, C., Zanoni, A.: Numerical stability and stabilization of groebner basis computation. In: ISSAC '02: Proceedings of the 2002 international symposium on Symbolic and algebraic computation, New York, NY, USA, ACM (2002) 262–269
7. Weispfenning, V.: Gröbner bases for inexact input data. In: Proceedings of CASC2003 (Computer Algebra in Scientific Computing). (2002) 403–411
8. Weispfenning, V.: Comprehensive Gröbner bases. J. Symbolic Comput. **14** (1992) 1–29
9. Weispfenning, V.: Canonical comprehensive Gröbner bases. J. Symbolic Comput. **36** (2003) 669–683 International Symposium on Symbolic and Algebraic Computation (ISSAC'2002) (Lille).
10. Nabeshima, K.: A speed-up of the algorithm for computing comprehensive gröbner systems. In: ISSAC '07: Proceedings of the 2007 international symposium on Symbolic and algebraic computation, New York, NY, USA, ACM (2007) 299–306
11. Lazard, D.: Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In: Computer algebra (London, 1983). Volume 162 of Lecture Notes in Comput. Sci. Springer, Berlin (1983) 146–156
12. Byröd, M., Josephson, K., Åström, K.: Fast optimal three view triangulation. In Yagi, Y., Kweon, I.S., Kang, S.B., Zha, H., eds.: Asian Conference on Computer Vision. (2007)
13. Faugére, J.C.: A new efficient algorithm for computing Gröbner bases (F_4). J. Pure Appl. Algebra **139** (1999) 61–88
14. Faugère, J.C.: A new efficient algorithm for computing Gröbner bases without reduction to zero (F_5). In: Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation, New York, ACM (2002) 75–83 (electronic)
15. Becker, T., Weispfenning, V.: Gröbner bases. Volume 141 of Graduate Texts in Mathematics. Springer-Verlag, New York (1993) A computational approach to commutative algebra, In cooperation with Heinz Kredel.

The preliminary implementation code can be found at the following URL.
`http://wwwmain.h.kobe-u.ac.jp/~nagasaka/research/snap/casc09.nb`