Towards Computing a Gröbner Basis of a Polynomial Ideal over a Field by Using Matrix Triangularization

Manuela Wiesinger-Widi*
Doctoral Program Computational Mathematics
Johannes Kepler University Linz
4040 Linz, Austria
manuela.wiesinger@dk-compmath.jku.at

Abstract

We give first results of our investigation of the connection between Gröbner bases computation and Gaussian elimination. We show that for every input set F of polynomials a matrix of shifts of those polynomials exists such that by triangularizing this matrix we obtain a Gröbner basis of F.

1 Introduction

In his PhD thesis [1], Buchberger introduced the notion of Gröbner bases and gave the first algorithm for computing them. Since then, extensive research has been done in order to reduce the complexity of the computation. But nevertheless, even for small examples the computation sometimes does not terminate in reasonable time.

Apart from the approach pursued by the Buchberger algorithm to compute a Gröbner basis — namely we start from the initial set F, execute certain reduction steps (consisting of multiplication of polynomials by terms, called shifts, and subtraction of polynomials) and after finitely many iterations of this procedure (by Buchberger's theorem) obtain a Gröbner basis of the ideal generated by F — Buchberger has for a long time proposed a second one. This approach is to start from F, execute certain shifts of the initial polynomials in F, arrange them as rows in a matrix, triangularize this matrix and from the resulting matrix extract a Gröbner basis.

In project DK1 of the Doctoral Program, which was proposed by Buchberger, we pursue the second approach and seek to improve the theory in order to speed up the Gröbner bases computation. This approach has been studied a couple of times in the past, but never thoroughly. The immediate question is: Does there exist a finite set of shifts such that a triangularization of the matrix built by these shifts yields a Gröbner basis and, if so, how can we construct these shifts? After our results for the univariate case (see [3]), we give first results for the multivariate case.

*This project is funded by the Austrian Science Fund (FWF) under grant W1214/DK1.

2 Notation

We denote the set of natural numbers (including 0) by $\mathbb{N}$ and we denote the cardinality of a set A by $|A|$. For $k \in \mathbb{N}$ we define

$$[k] := \{i \in \mathbb{N} : 1 \leq i \leq k\}.$$

For a k-tuple s we denote its length k by $|s|$ (the length of the empty tuple is 0) and its i-th entry by $s[i]$, $i \in [k]$. We denote the concatenation of two tuples s_1 and s_2 by $s_1 \smile s_2$.

Let $\mathbb{K}$ be a field and $n \in \mathbb{N} \setminus \{0\}$. We denote the ring of $n-$variate polynomials over $\mathbb{K}$ by $\mathbb{K}[x_1, \ldots, x_n] = \mathbb{K}[X]$ and we call $[x_1, \ldots, x_n] = [X]$ the set of terms or power products over $x_1, \ldots, x_n$. The ideal generated by a set $F \subseteq \mathbb{K}[X]$ over $\mathbb{K}[X]$ will be written as $\mathrm{Id}(F)$.

We fix an admissible order $\prec$ on $[X]$. For any polynomial $f \in \mathbb{K}[X]$, $f = \sum_{t \in [X]} c_t\, t$, the support of f is defined as $\mathrm{supp}(f) = \{t \in [X] : c_t \neq 0\}$. Note that $\mathrm{supp}(f)$ is a finite set. We denote the leading term and the leading coefficient of $f \neq 0$ by $\mathrm{lt}(f) = \max(\mathrm{supp}(f))$ and $\mathrm{lc}(f) = c_{\mathrm{lt}(f)}$, respectively. For a set F of polynomials we define $\mathrm{lt}(F) := \{\mathrm{lt}(f) : f \in F \setminus \{0\}\}$.

For $k, l \in \mathbb{N} \setminus \{0\}$ we denote the set of $k \times l$-matrices over $\mathbb{K}$ by $\mathbb{K}^{k \times l}$. For $M \in \mathbb{K}^{k \times l}$ we denote its entry in the i-th row ($i \in [k]$) and j-th column ($j \in [l]$) by $M_{i,j}$.

3 Preliminaries

The formulations of some of the following definitions are taken or adapted from the definitions in [2].

Definition 1 (Reduction)**.** Let $f, g \in \mathbb{K}[X] \setminus \{0\}$ have the same leading term. The *reduction* of f by g is defined as $\mathrm{red}(f, g) = \mathrm{lc}(g)\, f - \mathrm{lc}(f)\, g$.

Remark 2. Actually, we could have taken any polynomial $c(\mathrm{lc}(g)\, f - \mathrm{lc}(f)\, g)$, $c \in \mathbb{K} \setminus \{0\}$, as $\mathrm{red}(f, g)$ and still all the statements we are about to make remain true. We choose the above definition in order to avoid introducing fractions.

Definition 3 (S-polynomial, Polynomial reduction)**.** Let $f_1, f_2 \in \mathbb{K}[X] \setminus \{0\}$.

1. The *S-polynomial* of f_1 and f_2 is defined as

   $$\mathrm{Spol}(f_1, f_2) = \mathrm{red}(f_1 u_1, f_2 u_2)$$

   where $u_i = \mathrm{lcm}(\mathrm{lt}(f_1), \mathrm{lt}(f_2))/\,\mathrm{lt}(f_i)$, $i = 1, 2$.

2. If $\mathrm{lt}(f_2)$ divides a term u which appears with a non-zero coefficient c in f_1 then we say f_1 *reduces* by f_2 to h where $h = \mathrm{lc}(f_2) f_1 - c \cdot (u/\,\mathrm{lt}(f_2)) \cdot f_2$. We write $f_1 \to_{f_2} h$. If u is the leading term of f_1, the reduction is called *head reduction*.
   We say f_1 *reduces* to h w.r.t. a set of polynomials F (written $f_1 \to_F h$) if there exists $f_2 \in F$ such that $f_1 \to_{f_2} h$. We denote by $\to_f^*$ and $\to_F^*$ the reflexive transitive closures of the relations $\to_f$ and $\to_F$, respectively.
   If $f_1 \to_F^* h$ and h is irreducible w.r.t. F then we call h a *normal form* of f w.r.t. F.

Since the relation $\to_F$ is Noetherian, a normal form always exists. However, it does not need to be unique. Uniqueness is guaranteed, if F is a Gröbner basis.

Definition 4 (Gröbner basis)**.** A finite set $G \subseteq K[X] \setminus \{0\}$ is a *Gröbner basis* iff it satisfies the following condition:

$$\text{For all } u \in \mathrm{lt}(\mathrm{Id}(G)) \text{ there exists a } v \in \mathrm{lt}(G) \text{ such that } v \mid u.$$

For a set $F \subseteq \mathbb{K}[X]$ we say that G is a *Gröbner basis of* F (or $\mathrm{Id}(F)$), if G is a Gröbner basis and $\mathrm{Id}(G) = \mathrm{Id}(F)$.

Lemma 5. *Let $G \subseteq \mathbb{K}[X]$ be a Gröbner basis and let $f \in \mathbb{K}[X]$. Then $f \in \mathrm{Id}(G)$ if and only if $f \to_G^* 0$.*

Proof. Assume $f \to_G^* 0$. Then there exist $g_1, \ldots, g_m \in G$, $p_1, \ldots, p_m \in \mathbb{K}[X]$ such that $f = \sum_{i=1}^m p_i g_i$. Hence, $f \in \mathrm{Id}(G)$.

Assume $f \in \mathrm{Id}(G)$. Let h be a normal form of f w.r.t. G. Assume that $h \neq 0$. Since $f \to_G^* h$, it follows that $h \in \mathrm{Id}(G)$. Since G is a Gröbner basis and $h \neq 0$, there exists a $g \in G$ such that $\mathrm{lt}(g) \mid \mathrm{lt}(h)$. Hence, h is reducible by g and cannot be a normal form of f w.r.t. G. Therefore, $h = 0$. □

Lemma 6. *Let $G \subseteq K[X]$ be a Gröbner basis and let $f, g \in G$, $f \neq g$, such that $\mathrm{lt}(g) \mid \mathrm{lt}(f)$. Then $G \setminus \{f\}$ is a Gröbner basis of G.*

Proof. It needs to be shown that $G \setminus \{f\}$ is a Gröbner basis and that $\mathrm{Id}(G \setminus \{f\}) = \mathrm{Id}(G)$.

We first prove that $\mathrm{Id}(G \setminus \{f\}) = \mathrm{Id}(G)$. Clearly, $\mathrm{Id}(G \setminus \{f\}) \subseteq \mathrm{Id}(G)$. In order to show that $\mathrm{Id}(G) \subseteq \mathrm{Id}(G \setminus \{f\})$, it suffices to show that $f \in \mathrm{Id}(G \setminus \{f\})$. Let $t \in [X]$ such that $t\ \mathrm{lt}(g) = \mathrm{lt}(f)$ and let $p = f - tg$. If $p = 0$, then $f = tg \in \mathrm{Id}(G \setminus \{f\})$, since $g \in G \setminus \{f\}$. If $p \neq 0$, then $\mathrm{lt}(p) \prec \mathrm{lt}(f)$. Since G is a Gröbner basis, $f \in \mathrm{Id}(G)$ and f cannot be used for a reduction step anymore, we get with Lemma 5 that $p \to_{G \setminus \{f\}}^* 0$. It follows that $p \in \mathrm{Id}(G \setminus \{f\})$ and hence $f = p + tg \in \mathrm{Id}(G \setminus \{f\})$.

We now prove that $G \setminus \{f\}$ is a Gröbner basis. We need to show that

$$\underset{u \in \mathrm{lt}(\mathrm{Id}(G \setminus \{f\}))}{\forall} \ \underset{v \in \mathrm{lt}(G \setminus \{f\})}{\exists} v \mid u.$$

Since $\mathrm{Id}(G \setminus \{f\}) = \mathrm{Id}(G)$ as shown in the first part, we obtain $\mathrm{lt}(\mathrm{Id}(G \setminus \{f\})) = \mathrm{lt}(\mathrm{Id}(G))$. Hence it remains to show that

$$\underset{u \in \mathrm{lt}(\mathrm{Id}(G))}{\forall} \ \underset{v \in \mathrm{lt}(G \setminus \{f\})}{\exists} v \mid u.$$

Let $u \in \mathrm{lt}(\mathrm{Id}(G))$. Since G is a Gröbner basis, there exists a $w \in \mathrm{lt}(G)$ such that $w \mid u$. Assume $w \notin \mathrm{lt}(G \setminus \{f\})$, then $w = \mathrm{lt}(f)$. Since $\mathrm{lt}(g) \mid \mathrm{lt}(f) = w$, it follows with $v = lt(g) \in \mathrm{lt}(G \setminus \{f\})$ that $v \mid u$. □

Lemma 7. *Let $G \subseteq K[X]$ be a Gröbner basis and let $G' \subseteq \mathrm{Id}(G)$ be finite such that $\mathrm{lt}(G) = \mathrm{lt}(G')$. Then G' is a Gröbner basis of G.*

Proof. Since G is a Gröbner basis of g and $G' \subseteq \mathrm{Id}(G)$, also $G \cup G'$ is a Gröbner basis of G. Now by Lemma 6, $(G \cup G') \setminus (G \setminus G') = G'$ is a Gröbner basis of G. □

The following theorem and algorithm are due to Buchberger (see [1]).

Theorem 8. *A finite set $G \subseteq K[X] \setminus \{0\}$ is a Gröbner basis if and only if the following holds:*

$$\text{For all } f, g \in G,\ \mathrm{Spol}(f,g) \to_G^* 0.$$

Algorithm Gröbner basis (Buchberger algorithm)
Input: F finite set of non-zero polynomials
Output: G Gröbner basis for F
$G \leftarrow F$;
$B \leftarrow \{\{f,g\} \mid f,g \in G, f \neq g\}$;
while $B \neq \emptyset$
take a pair $\{f,g\}$ from B;
$B \leftarrow B \setminus \{\{f,g\}\}$;
$h \leftarrow$ some normal form of $\mathrm{Spol}(f,g)$ with respect to G;
if $h \neq 0$ **then** $B \leftarrow B \cup \{\{h,g'\} \mid g' \in G\}$;
$G \leftarrow G \cup \{h\}$;
end while;
Return G;

We now identify polynomials with vectors respectively rows of matrices. Since finite subsets of $[X]$ are totally ordered, we can use their elements as indices for vectors and for the columns of matrices. Given a polynomial $f = \sum_{t \in [X]} c_t\, t \in \mathbb{K}[X]$, for any finite, nonempty set $A \subset [X]$ with $\mathrm{supp}(f) \subseteq A$ we associate to f the vector $v_A(F) = (c_t)_{t \in A} \in \mathbb{K}^A$. Conversely, to any vector $c = (c_t)_{t \in A} \in \mathbb{K}^A$ we associate the polynomial $pol(c) = \sum_{t \in A} c_t\, t$. We define the leading term and the leading coefficient of a vector c as $\mathrm{lt}(c) := \mathrm{lt}(pol(c))$ and $\mathrm{lc}(c) := c_{\mathrm{lt}(c)}$. When A is clear from the context, we will use interchangeably f and $v_A(f)$, as well as c and $pol(c)$. We will use the convention of writing down the entries of the vectors resp. rows (i.e. the coefficients) from left to right in decreasing order of their corresponding terms w.r.t. $\prec$. For the remaining part of the paper we also fix a total ordering $<$ on $K[X]$.

Definition 9. Let $A \neq \emptyset$ be a finite set. We define $\mathrm{tuple}(A)$ to be the tuple of all the elements in A ordered according to $<$, i.e.

$$\mathrm{tuple}(A) := \begin{cases} \langle a \rangle & \text{if } A = \{a\} \\ \langle a \rangle \frown \mathrm{tuple}(A \setminus \{a\}) & \text{if } |A| \geq 2 \text{ where } a = \min_{<} A. \end{cases}$$

Definition 10 (Matrix corresponding to a tuple/set of polynomials)**.** Let $r \in \mathbb{N} \setminus \{0\}$. For any tuple of polynomials $\langle f_1, \ldots, f_r \rangle \in K[X]^r$ we define the matrix

$$\mathrm{mat}(\langle f_1, \ldots, f_r \rangle) := \begin{pmatrix} v_A(f_1) \\ \vdots \\ v_A(f_r) \end{pmatrix}$$

where $A = \bigcup_{i=1}^r \mathrm{supp}(f_i)$, and for a non-empty, finite set $F \subset K[X]$ we define

$$\mathrm{mat}(F) := \mathrm{mat}(\mathrm{tuple}(F)).$$

By abuse of notation, for a matrix M corresponding to a set of polynomials with column index set A and for a polynomial f with $\mathrm{supp}(f) \subseteq A$, we write $f \in M$ iff $v_A(f)$ is a row of M.

Now recall some basic concepts and knowledge about matrices in general.

Definition 11 (Pivot column)**.** Let $M \in \mathbb{K}^{k\times l}$ with $k, l \in \mathbb{N} \setminus \{0\}$. For a non-zero row $i \in [k]$, the *pivot column* of i in M is the unique $j \in [l]$ for which $M_{i',j} = 0$ for every $i' < i$ and $M_{i,j} \neq 0$. Column $j \in [l]$ is a *pivot column* of M iff there exists a row $i \in [k]$ such that j is its pivot column in M. We define $\mathrm{PC}(M) := \{j \in [l] : j \text{ is a pivot column of } M\}$.

Remark 12. If we consider a matrix corresponding to a set F of polynomials, the pivot column of a non-zero polynomial in F is indexed by its leading term.

Definition 13 (Triangular matrix, triangularization)**.** We call a matrix M *triangular* iff no two rows of M have the same pivot column. A *triangularization step* on a matrix M is defined to be one of the following steps:

1. exchange of two rows in M
2. multiplication of a row in M by a constant $c \in \mathbb{K} \setminus \{0\}$
3. addition of a multiple $c_1\, r_1$ of one row r_1 to a multiple $c_2\, r_2$ of another row r_2, where $c_1, c_2 \in \mathbb{K} \setminus \{0\}$.

We say to *triangularize* a matrix M_1 (to M_2) if we apply a sequence of triangularization steps to M_1 to obtain a triangular matrix (M_2).

Remark 14. One way to triangularize a matrix is by Gaussian elimination. If we consider a matrix corresponding to a set of polynomials, then the reduction of a non-zero row c by a non-zero row d, where $\mathrm{lt}(c) = \mathrm{lt}(d)$, i.e. replacing row c by $\mathrm{red}(c, d)$ is an example of a triangularization step.

Recall the following well known result.

Lemma 15. *Let $M \in \mathbb{K}^{k\times l}$ with $k, l \in \mathbb{N} \setminus \{0\}$ and let M_1, M_2 be matrices obtained from M by triangularization. Then*

$$\mathrm{PC}(M_1) = \mathrm{PC}(M_2).$$

4 Gröbner bases computation and Gaussian elimination

The problem we investigate is the following.

Problem 16.
Given: A finite set $F = \{f_1, \ldots, f_r\} \subset \mathbb{K}[X] \setminus \{0\}$ $(r \geq 2)$.
Question: Do there exist finite $U_1, \ldots, U_r \subseteq [X]$ such that if $S = \bigcup_{i=1}^{r} U_i f_i$ and M is a matrix we obtain by triangularizing $\mathrm{mat}(S)$, M contains a Gröbner basis of $\mathrm{Id}(F)$?

The answer to this question is yes, as we will see.

Convention: From now on we will only perform head reductions when computing a Gröbner basis by the Buchberger algorithm. This neither influences the correctness nor the termination of the algorithm.

Consider the following: If we compute a Gröbner basis of F by the Buchberger algorithm, then every polynomial that is added to the basis during the computation is obtained by successive application of the function red to elements in F multiplied by some terms. This is easily

seen if one observes that $u\,\mathrm{red}(\tilde{f}_1, \tilde{f}_2) = \mathrm{red}(u\tilde{f}_1, u\tilde{f}_2)$ for $u \in [X]$ and $\tilde{f}_1, \tilde{f}_2 \in [X]F$. For example for $u_1, u_2, u_3, v \in [X]$ and $f_1, f_2, f_3 \in F$, if $h_1 = \mathrm{red}(u_1 f_1, u_2 f_2)$ and $h_2 = \mathrm{red}(u_3 f_3, v h_1)$, then $h_2 = \mathrm{red}(u_3 f_3, \mathrm{red}(v u_1 f_1, v u_2 f_2))$. We can associate to every such nested reduction structure with entries in $[X]F$ a nested binary tuple by just putting successively the reductees into the first component of the (sub-)tuples and the reductors into the second while preserving the nested structure. In the example above we associate for example to $\mathrm{red}(u_3 f_3, \mathrm{red}(v u_1 f_1, v u_2 f_2))$ the tuple $\langle u_3 f_3, \langle v u_1 f_1, v u_2 f_2\rangle\rangle$.

If we put the input polynomials and all of the polynomials (represented as the nested tuples described above) that are added to the basis during the computation into a tuple in the order of their computation, we can describe the history of the Gröbner basis computation by this tuple. We give an extended Gröbner basis algorithm which additionally returns the corresponding history tuple of the computation. For this, we define for a term $u \in [X]$ and a tuple s a component wise multiplication $u\,s$.

Algorithm Gröbner basis + history tuple
Input: F finite set of non-zero polynomials
Output: $\langle G, s\rangle$, where G is a Gröbner basis for $\mathrm{Id}(F)$ and s is the history tuple of the computation
$G \leftarrow F$;
$s \leftarrow \langle f_1, \ldots, f_r\rangle$;
$c \leftarrow \langle f_1, \ldots, f_r\rangle$;
$B \leftarrow \{\{f, g\} \mid f, g \in G, f \neq g\}$;
while $B \neq \emptyset$
take a pair $\{f, g\}$ from B;
$B \leftarrow B \setminus \{\{f, g\}\}$;
$h \leftarrow \mathrm{Spol}(f, g)$;
$histh \leftarrow \langle \frac{\mathrm{lcm}(\mathrm{lt}(f),\mathrm{lt}(g))}{\mathrm{lt}(f)} s[i], \frac{\mathrm{lcm}(\mathrm{lt}(f),\mathrm{lt}(g))}{\mathrm{lt}(g)} s[j]\rangle$,
where i is such that $c[i] = f$ and j is such that $c[j] = g$;
while there exists $b \in G$ with $\mathrm{lt}(b) | \mathrm{lt}(h)$
$histh \leftarrow \langle histh, \frac{\mathrm{lt}(h)}{\mathrm{lt}(b)} s[i]\rangle$, where i is such that $c[i] = b$;
$h \leftarrow \mathrm{red}(h, \frac{\mathrm{lt}(h)}{\mathrm{lt}(b)} b)$;
end while;
if $h \neq 0$ **then** $c \leftarrow c \smile \langle h\rangle$;
$s \leftarrow s \smile \langle histh\rangle$;
$B \leftarrow B \cup \{\{h, g'\} \mid g' \in G\}$;
$G \leftarrow G \cup \{h\}$;
end while;
Return $\langle G, s\rangle$;

For the rest of the paper, we fix a computation strategy for the Buchberger algorithm (observe that as it is stated above, several choices for critical pairs and reductor polynomials are possible) and denote the history tuple of the computation with input F by $\mathrm{hist}(F)$.

Let us consider the following example of a Gröbner basis computation.

Example 17. Let $F = \{f_1, f_2, f_3\}$ with $f_1 = y^2$, $f_2 = xy^2 + x^2 + xy$ and $f_3 = x^2 y + y$, and as

admissible ordering let us use the degree lexicographic order with $y \prec x$. We perform the following computations.

- $\mathrm{Spol}(f_3, f_2) = yf_3 - xf_2 = -x^3 - x^2y + y^2 =: f_4$
  We have $f_4 = \mathrm{red}(yf_3, xf_2)$ and $histh = \langle yf_3, xf_2\rangle$.
- $\mathrm{Spol}(f_3, f_4) = -xf_3 - yf_4 = x^2y^2 - y^3 - xy \to_{yf_3} -y^3 - xy - y^2 \to_{yf_1} -xy - y^2 =: f_5$
  We have $f_5 = \mathrm{red}(\mathrm{red}(\mathrm{red}(xf_3, \mathrm{red}(y^2f_3, xyf_2)), yf_3), yf_1)$ and
  $histh = \langle\langle\langle xf_3, \langle y^2f_3, xyf_2\rangle\rangle, yf_3\rangle, yf_1\rangle$.
- $\mathrm{Spol}(f_1, f_2) = xf_1 - f_2 = -x^2 - xy =: f_6$
  We have $f_6 = \mathrm{red}(xf_1, f_2)$ and $histh = \langle xf_1, f_2\rangle$.
- $\mathrm{Spol}(f_3, f_6) = -f_3 - yf_6 = xy^2 - y \to_{xf_1} -y =: f_7$
  We have $f_7 = \mathrm{red}(\mathrm{red}(f_3, \mathrm{red}(xyf_1, yf_2)), xf_1)$ and
  $histh = \langle\langle f_3, \langle xyf_1, yf_2\rangle\rangle, xf_1\rangle$.

All of the other S-polynomials reduce to 0. Hence, we obtain $G = \{f_1, f_2, f_3, f_4, f_5, f_6, f_7\}$ as a Gröbner basis of $\mathrm{Id}(F)$.

The history tuple of this computation is

$$\begin{aligned}\mathrm{hist}(F) \;=\; & \langle f_1, f_2, f_3, \langle yf_3, xf_2\rangle, \langle\langle\langle xf_3, \langle y^2f_3, xyf_2\rangle\rangle, yf_3\rangle, yf_1\rangle,\\ & \langle xf_1, f_2\rangle, \langle\langle f_3, \langle xyf_1, yf_2\rangle\rangle, xf_1\rangle\rangle.\end{aligned}$$

Definition 18 ($\mathrm{matrix}(s)$)**.** Let s be a history tuple of a Gröbner basis computation and let $\mathrm{P}(s)$ be the set of all the polynomials in s. We define

$$\mathrm{matrix}(s) := \mathrm{mat}(\mathrm{P}(s)).$$

Example 19. We continue Example 17. We had $F = \{f_1, f_2, f_3\}$ with $f_1 = y^2$, $f_2 = xy^2 + x^2 + xy$ and $f_3 = x^2y + y$, and

$$\begin{aligned}\mathrm{hist}(F) \;=\; & \langle f_1, f_2, f_3, \langle yf_3, xf_2\rangle, \langle\langle\langle xf_3, \langle y^2f_3, xyf_2\rangle\rangle, yf_3\rangle, yf_1\rangle,\\ & \langle xf_1, f_2\rangle, \langle\langle f_3, \langle xyf_1, yf_2\rangle\rangle, xf_1\rangle\rangle.\end{aligned}$$

Now,

$$\mathrm{P}(\mathrm{hist}(F)) = \{f_1, f_2, f_3, yf_1, xf_1, xyf_1, yf_2, xf_2, xyf_2, yf_3, xf_3, y^2f_3\}$$

and

$$\begin{aligned}\mathrm{matrix}(\mathrm{hist}(F)) \;=\; & \mathrm{mat}(\{f_1, f_2, f_3, yf_1, xf_1, xyf_1, yf_2, xf_2, xyf_2,\\ & yf_3, xf_3, y^2f_3\}).\end{aligned}$$

Mandache attempted to simulate the Gröbner basis computation done via Buchberger algorithm (represented by the history tuple $\mathrm{hist}(F)$) in $\mathrm{matrix}(\mathrm{hist}(F))$ by performing the same reduction steps on these rows as were done in the Gröbner basis computation itself (these steps are recorded in $\mathrm{hist}(F)$). This attempt turned out to fail. She found an example where this is not possible (see [2, p. 51]).

The problem with this approach was that in a matrix, when a row c is reduced by another row d, the row c vanishes. However, it may be that exactly this row is used again at a later step in the course of the Gröbner basis computation. Since it has vanished this new reduction cannot be simulated on the matrix. One approach to get rid of this problem is to put a copy of c into the original matrix (before any reduction steps are performed). By doing this for every row that vanishes during the reduction in the matrix but is still needed at a later step, one can perform all of the reduction steps that are done in the Gröbner basis computation also on the matrix. We obtain a sufficient amount of copies of each row by analyzing the history tuple of the Gröbner basis computation that we want to simulate.

This concept will be used to prove that while the Gröbner basis computation with history tuple $\mathrm{hist}(F)$ cannot always be simulated on $M = \mathrm{matrix}(\mathrm{hist}(F))$, we can still always get a Gröbner basis of $\mathrm{Id}(F)$ by triangularizing M.

In the following definition $\mathrm{Head}(s)$ gives the first element in a tuple s, $\mathrm{Tail}(s)$ the rest of the tuple.

Definition 20 ($\mathrm{Flatten}(s)$)**.** Let s be a nested tuple of polynomials. We define

$$\mathrm{Flatten}(s) := \begin{cases} \langle\rangle & \text{if } s = \langle\rangle, \\ \langle \mathrm{Head}(s)\rangle \frown \mathrm{Flatten}(\mathrm{Tail}(s)) & \text{if } \mathrm{Head}(s) \text{ is a polynomial,} \\ \mathrm{Flatten}(\mathrm{Head}(s)) \frown \mathrm{Flatten}(\mathrm{Tail}(s)) & \text{otherwise.} \end{cases}$$

Example 21. We continue Example 19. We had

$$\begin{aligned} \mathrm{hist}(F) \;=\; & \langle f_1, f_2, f_3, \langle yf_3, xf_2\rangle, \langle\langle\langle xf_3, \langle y^2 f_3, xyf_2\rangle\rangle, yf_3\rangle, yf_1\rangle, \\ & \langle xf_1, f_2\rangle, \langle\langle f_3, \langle xyf_1, yf_2\rangle\rangle, xf_1\rangle\rangle. \end{aligned}$$

Now,

$$\begin{aligned} \mathrm{Flatten}(\mathrm{hist}(F)) \;=\; & \langle f_1, f_2, f_3, yf_3, xf_2, xf_3, y^2 f_3, xyf_2, yf_3, yf_1, \\ & xf_1, f_2, f_3, xyf_1, yf_2, xf_1\rangle. \end{aligned}$$

Observe that xf_1, f_2, f_3 and yf_3 occur twice.

Given a history tuple s, we can now define a matrix with copies of rows on which we can simulate the computation described by s.

Definition 22 ($\mathrm{copymatrix}(s)$)**.** Let s be a history tuple of a Gröbner basis computation. We define

$$\mathrm{copymatrix}(s) := \mathrm{mat}(\mathrm{Flatten}(s)).$$

Definition 23. (Contour) Let M be a triangular matrix associated to a set of polynomials. We define

$$\mathrm{contour}(M) := \{f \in M : f \neq 0 \wedge \underset{\substack{g \in M \\ g \neq 0 \wedge g \neq f}}{\forall} \mathrm{lt}(g) \nmid \mathrm{lt}(f)\}.$$

The following theorem provides a constructive solution to Problem 16. However, since a Gröbner basis is used to build the matrix used for triangularization, it is not suitable for deriving a new algorithm for computing Gröbner bases.

Theorem 24. *Let $F \subseteq \mathbb{K}[X] \setminus \{0\}$ be finite, $|F| > 1$. Then for all M, which can be obtained by triangularizing* matrix(hist(F))*,* contour(M) *is a Gröbner basis of F.*

Proof. Let $s = \text{hist}(F)$. First we show that the computation described by s can be simulated on copymatrix(s). The matrix resulting from the simulation contains a Gröbner basis of F. We then show that we can triangularize this matrix in such a way that the same Gröbner basis is still contained. Afterwards we show that any triangularization of copymatrix(s) yields a Gröbner basis of F and that any such Gröbner basis can also be computed from matrix(s). Finally we prove that the contour of a matrix obtained from matrix(s) by triangularization is a Gröbner basis of F.

First observe that with every triangularization step we stay in the ideal generated by the rows of the initial matrix. Now let $M_1 = \text{copymatrix}(s)$ and P be the set of all the polynomials in s. We label each row in M_1 by the element in P which it represents. Since we might have several copies of rows, one and the same label may be attached to several rows. Now we perform the reductions on the rows of M_1 as described by s, where we do not use those rows anymore which are additionally labeled by "fin" (which rows will be augmented by this label will be described later). Remember that the first $|F|$ entries in s just correspond to the elements in the input basis and do not describe reduction steps. Hence, the simulation starts with the $(|F|+1)$-st entry in s. If for reduction there are several rows with the same label to choose from, we freely choose one of them and proceed with this one. If c and d are the labels of two rows and c is reduced by d, we change the label of the reduced row to $\langle c, d\rangle$. For all $i \in [|s|]$, if the reductions described in $s[i]$ are finished, we augment the row with label $s[i]$ (which is the row that has been produced last) with an additional label "fin". We are finished with simulating the Gröbner basis computation described by s if the last element in s has been computed. Since M_1 contains all of the polynomials in s as often as they occur in s, it is ensured that every reduction step in s can be simulated successfully.

Let M_2 be the matrix resulting from the simulation and let G be the Gröbner basis yielded by the computation described by s. Then we have $G = F \cup Fin$, where Fin is the set of polynomials that correspond to those rows in M_2 that are labeled by "fin".

Now M_2 is not necessarily triangular. But all the elements in G have different leading terms. So we can triangularize M_2 such that all elements of G are rows of the resulting matrix M_3. We accomplish this by using the elements in G exclusively as reductors and not as reductees. By Lemma 15 and Lemma 7 it follows that every triangular matrix obtained from M_1 yields a Gröbner basis of F.

Now we prove that we actually would not have needed the additional copies in M_1. Assume, the row c is contained in M_1 more than once. If we delete this row from M_1, this amounts to deleting a zero-row in a triangular matrix obtained from M_1. This has no effect on the rows that form a Gröbner basis. Hence, we can delete all of the additional copies in M_1, which amounts to using matrix(s) in the first place.

Finally we show that the contour of a matrix M obtained from matrix(s) by triangularization is a Gröbner basis of F. By the arguments above M contains a Gröbner basis of F. Since the other rows are elements of Id(F), the set of all the non-zero rows in M is a Gröbner basis of F. By Lemma 6, contour(M) is a Gröbner basis of F. □

Remark 25. In Theorem 24 we take contour(M) instead of the set of all non-zero polynomials in M because contour(M) is significantly smaller in general.

5 Conclusion

We showed the existence of a matrix M of shifts of the input polynomials in F such that after triangularization of M we obtain a Gröbner basis of F. Although Theorem 24 allows for a construction of M, it is not practical since for the construction we use a Gröbner basis of F in the first place. We are currently working on building such a matrix without the use of a prior Gröbner bases computation.

References

[1] Bruno Buchberger. An Algorithm for Finding the Basis Elements in the Residue Class Ring Modulo a Zero Dimensional Polynomial Ideal (german). Mathematical Institute, University of Innsbruck, Austria. PhD Thesis. 1965. English translation in *J. of Symbolic Computation, Special Issue on Logic, Mathematics, and Computer Science: Interactions*. Vol. 41, Number 3-4, pages 475–511, 2006.

[2] Ana Maria Mandache. Gröbner Bases Computation and Gaussian Elimination. RISC, Johannes Kepler University Linz. PhD Thesis. 1995.

[3] Manuela Wiesinger-Widi. Sylvester Matrix and GCD for Several Univariate Polynomials. Technical report, DK Computational Mathematics, JKU, Linz, Austria, 2011.