A NEW ALGORITHM FOR COMPUTING GRÖBNER BASES[*]

SHUHONG GAO[†], FRANK VOLNY IV[‡], AND MINGSHENG WANG[§]

Abstract. Buchberger's algorithm for computing Gröbner bases was introduced in 1965, and subsequently there have been extensive efforts in improving its efficiency. Major algorithms include F4 (Faugère 1999), XL (Courtois et al. 2000) and F5 (Faugère 2002). F5 is believed to be the fastest algorithm known in the literature. Most recently, Gao, Guan and Volny (2010) introduced an incremental algorithm (G2V) that is simpler and several times faster than F5. In this paper, a new algorithm is presented that can avoid the incremental nature of F5 and G2V. It matches Buchberger's algorithm in simplicity and yet is more flexible. More precisely, given a list of polynomials, the new algorithm computes simultaneously a Gröbner basis for the ideal generated by the polynomials and a Gröbner basis for the leading terms of the syzygy module of the given list of polynomials. For any term order for the ideal, one may vary signature orders (i.e. the term orders for the syzygy module). Under one signature order, the new algorithm specializes to the G2V, and under another signature order, the new algorithm is several times faster than G2V, as indicated by computer experiments on benchmark examples.

Key words. Gröbner basis, Buchberger's Algorithm, Syzygy Module, F5 Algorithm, Module, Quotient Ring

AMS subject classifications. 13P10, 68W10

1. Introduction.

Polynomial systems are ubiquitous in mathematics, science and engineering. Gröbner basis theory is one of the most powerful tools for solving polynomial systems and is essential in many computational tasks in algebra and algebraic geometry. Buchberger introduced in 1965 the first algorithm for computing Gröbner bases, and it has been implemented in most computer algebra systems (e.g., Maple, Mathematica, Magma, Sage, Singular, Macaulay 2, CoCoA, etc).

There has been extensive effort in finding more efficient algorithms for computing Gröbner bases. In Buchberger's original algorithm (1965, [2]), one has to reduce many useless S-polynomials (i.e., those that reduce to 0 via long division), and each reduction is time consuming. It is natural to avoid useless reductions as much as possible. Buchberger [3, 4] discovered two simple criteria for detecting useless S-polynomials. Note that a reduction of an S-polynomial to 0 corresponds to a syzygy (for the initial list of polynomials). Möller, Mora and Traverso (1992, [16]) go a step further to present an algorithm using the full module of syzygies, however, their algorithm is not very efficient. Faugère (2002, [10]) introduced the idea of signatures and rewriting rules that can detect many useless S-polynomials, hence saving a significant amount of time that would be used in reducing them. In fact, for a regular sequence of polynomials, his algorithm F5 detects all useless reductions. By computer experiments, Faugère showed that his algorithm F5 is many times faster than previous algorithms. In fact, Faugère and Joux (2003, [11]) solved the first Hidden Field Equation (HFE) Cryptosystem Challenge which involves a system of 80 polynomial equations with 80

[*] Gao and Volny were partially supported by National Science Foundation under Grants DMS-1005369 and CCF-0830481 and Wang was partially supported by the National Science Foundation of China under Grant 60970134 and by 973 Project under Grant 2007CB311201.

[†] Department of Mathematical Sciences, Clemson University Clemson, SC 29634-0975 USA sgao@clemson.edu

[‡] Department of Mathematical Sciences, Clemson University Clemson, SC 29634-0975 USA fvolny@clemson.edu

[§] Information Security Lab, Institute of Software, Chinese Academy of Sciences, P.O. Box 8718 Beijing 100080, P. R. China mingsheng_wang@yahoo.com.cn

variables over the binary field (1996, [17]). Since F5 seems difficult to understand, there have been several papers trying to simplify and improve F5, see Eder and Perry (2009, [7]), Sun and Wang (2009, [18]), and Hashemi and Ars (2010, [13]).

In another direction of research, one tries to speed up the reduction step. Lazard (1983, [15]) pointed out the connection between Gröbner bases and linear algebra, that is, a Gröbner basis can be computed by Gauss elimination of a Sylvester matrix. The XL algorithm of Courtois et al. (2000, [5]) is an implementation of this Sylvester matrix, which is recently improved by Ding et al. (2008, [6]). A more clever approach is the F4 algorithm of Faugère (1999, [9]). F4 is an efficient method for reducing several S-polynomials simultaneously where the basic idea is to apply fast linear algebra methods to the submatrix of the Sylvester matrix consisting of only those rows that are needed for the reductions of a given list of S-polynomials. This method benefits from the efficiency of fast linear algebra algorithms. The main problem with this approach, however, is that the memory usage grows quickly (compared to, say, F5), even for medium systems of polynomials.

Recently, Gao, Guan and Volny [12] presented an algorithm (G2V) that is incremental in the same fashion as F5 and F5C but is much simpler and faster (in the range of 2 to 10 times faster on some benchmark problems). Incremental algorithms (like F5 and G2V), however, are at a disadvantage as the order of the input generators can have a profound effect on the complexity of the intermediate bases. It sometimes happens that intermediate Gröbner bases are exponential in size yet the final basis is small. In this case, any incremental algorithm will be extremely slow. Our contribution in this paper is to develop a new algorithm (GVW) that can avoid the incremental nature of F5 and G2V. It matches Buchberger's algorithm in simplicity yet is much faster. This new algorithm uses signatures in a similar fashion as F5 and G2V, but allows arbitrary orderings of the signatures (i.e. term orders of the syzygy modules). In this way, a Gröbner basis for $\langle g_1, \ldots, g_m \rangle$ can be computed in one-shot, incrementally, or as a hybrid of the two. Moreover, just as reductions to zero in G2V provide information about the colon ideal (and thus saving further computations), the present algorithm uses reductions to zero to discover more information about the $(g_1, \ldots, g_m)$-syzygy module in order to prevent later reductions to zero.

Upon termination, our algorithm computes a Gröbner basis for the ideal generated by $g_1, \ldots, g_m$ as well as a list of minimal leading terms for the $(g_1, \ldots, g_m)$-syzygy module. Unless one needs to compute the syzygy module under a certain term order, one is free to choose the term order of the syzygy module to maximize performance. A POT order corresponds to an incremental style algorithm while non-elimination orders correspond to one-shot algorithms. In fact, in the incremental mode, our new algorithm specializes to the G2V algorithm, but we find that the one-shot mode can be much faster (another 2 to 10 times faster than incremental mode and therefore several times faster than F5 and F5C).

The paper is organized as follows. In Section 2, we introduce the basic concepts and theory for our algorithm. In particular, we define signatures, regular top-reductions, super top-reductions, and the concept of eventually super top-reducibility. We also introduce J-pairs that are in some sense similar to S-polynomials. Then we characterize Gröbner bases in terms of J-pairs in a similar fashion as Buchberger's characterization in terms of S-polynomials. Our characterization goes a step further, that is, it also tells us when we have a Gröbner basis for the corresponding syzygy module. In Section 3, we present our algorithm and prove its correctness, which proves the correctness of G2V as a special case. The problem of finite termination of our algorithm is left open in this paper, but Huang [14] has completely characterized for which signature orders our algorithm have a finite termination. We present computer experiments of our algorithm that shows how the algorithm performs under different term orders for the syzygy module. Finally, in Section 4, we show how our algorithm can be adapted to compute Gröbner bases for modules and for polynomials over quotient rings, which would allow one to design more flexible incremental algorithms. In the conclusion section, we mention some recent related papers after the current paper was initially submitted.

2. Theory. Let $R = \mathbb{F}[x_1, \ldots, x_n]$ be a polynomial ring over a field $\mathbb{F}$ with n variables. Given polynomials $g_1, \ldots, g_m \in R$, we wish to compute a Gröbner basis for the ideal

$$I = \langle g_1, \ldots, g_m \rangle = \{u_1 g_1 + \cdots + u_m g_m : u_1, \ldots, u_m \in R\} \subseteq R \tag{2.1}$$

with respect to some term order on R. Define

$$\mathbf{H} = \{(u_1, \ldots u_m) \in R^m : u_1 g_1 + \cdots + u_m g_m = 0\}, \tag{2.2}$$

called **the syzygy module** of $\mathbf{g} = (g_1, \ldots, g_m)$. We would like to develop an algorithm that computes Gröbner bases for both I and $\mathbf{H}$. Note that elements of R^m are viewed as row vectors and are denoted by bold letters say $\mathbf{g}, \mathbf{u}$ etc. We consider the following R-submodule of $R^m \times R$:

$$M = \left\{(\mathbf{u}, v) \in R^m \times R : \mathbf{u}\mathbf{g}^t = v\right\}. \tag{2.3}$$

We define $\mathbf{E}_i \in R^m$ to be the i^{th} standard unit vector. Note that a monomial (or a term) in R is of the form

$$x^\alpha = \prod_{i=1}^{n} x_i^{a_i}$$

where $\alpha = (a_1, \ldots, a_n) \in \mathbb{N}^n$ is any vector of non-negative integers, and a term in R^m is of the form

$$x^\alpha \mathbf{E}_i$$

where $1 \leq i \leq m$ and $\alpha \in \mathbb{N}^n$. We say $x^\alpha \mathbf{E}_i$ divides $x^\beta \mathbf{E}_j$ if $i = j$ and x^α divides x^β, with the quotient being $(x^\beta \mathbf{E}_i)/(x^\alpha \mathbf{E}_j) = x^\beta / x^\alpha \in R$. Also, the R-module M is generated by

$$(\mathbf{E}_1, g_1), (\mathbf{E}_2, g_2), \ldots, (\mathbf{E}_m, g_m). \tag{2.4}$$

Fix any term order $\prec_1$ on R and any term order $\prec_2$ on R^m. We emphasize that the order $\prec_2$ may or may not be related to $\prec_1$ in the theory below, though $\prec_2$ is usually an extension of $\prec_1$ to R^m in implementation. For the sake of convenience, we shall use the following convention for leading terms:

$$\mathrm{lm}(v) = \mathrm{lm}_{\prec_1}(v), \quad \mathrm{lm}(\mathbf{u}) = \mathrm{lm}_{\prec_2}(\mathbf{u})$$

for any $v \in R$ and $\mathbf{u} \in R^m$. Note that, for $v \in R$, $\mathrm{lm}(v)$ is a monomial x^α, while, for $\mathbf{u} \in R^m$, $\mathrm{lm}(\mathbf{u})$ is a term $x^\alpha \mathbf{E}_i$ for some $\alpha \in \mathbb{N}^n$ and $1 \leq i \leq m$. We make the convention that if $v = 0$ then $\mathrm{lm}(v) = 0$; similarly for $\mathrm{lm}(\mathbf{u})$. This should not cause any confusion, but the reader should keep the two different orders in mind.

For any $(\mathbf{u}, v) \in R^m \times R$, we call $\mathrm{lm}(\mathbf{u})$ the **signature** of $(\mathbf{u}, v)$. In comparison, F5 defines a signature for a polynomial $v \in I = \langle g_1, \ldots, g_m\rangle$ to be $\mathrm{lm}(\mathbf{u})$ for any $\mathbf{u} \in R^m$ so that $(\mathbf{u}, v) \in M$, hence v may have many signatures. Our definition of signatures for pairs avoids this ambiguity of multiple signatures.

We now define top-reduction, similar to the top-reduction in F5. Let $p_1 = (\mathbf{u}_1, v_1), p_2 = (\mathbf{u}_2, v_2) \in R^m \times R$ be any two pairs. When v_2 is nonzero, we say p_1 is top-reducible by p_2 if the following two conditions are satisfied:

(i) v_1 is nonzero and $\mathrm{lm}(v_2)$ divides $\mathrm{lm}(v_1)$; and

(ii) $\mathrm{lm}(t\mathbf{u}_2) \preceq \mathrm{lm}(\mathbf{u}_1)$ where $t = \mathrm{lm}(v_1)/\mathrm{lm}(v_2)$.

The corresponding **top-reduction** is then

$$p_1 - ctp_2 = (\mathbf{u}_1 - ct\mathbf{u}_2, v_1 - ctv_2), \tag{2.5}$$

where $c = \mathrm{lc}(v_1)/\mathrm{lc}(v_2)$. The effect of a top-reduction is that the leading monomial in the v-part is canceled without increasing the signature of p_1. Such a top-reduction is called **regular**, if

$$\mathrm{lm}(\mathbf{u}_1 - ct\mathbf{u}_2) = \mathrm{lm}(\mathbf{u}_1),$$

and **super** otherwise. So the signature of p_1 remains the same under a regular top-reduction but becomes smaller under a super top-reduction. A super top-reduction happens if

$$\mathrm{lm}(t\mathbf{u}_2) = \mathrm{lm}(\mathbf{u}_1) \text{ and } \frac{\mathrm{lc}(\mathbf{u}_1)}{\mathrm{lc}(\mathbf{u}_2)} = \frac{\mathrm{lc}(v_1)}{\mathrm{lc}(v_2)}.$$

When $v_2 = 0$, we say that p_1 is **top-reducible** by $(\mathbf{u}_2, 0)$ if $\mathbf{u}_1$ and $\mathbf{u}_2$ are both nonzero and $\mathrm{lm}(\mathbf{u}_2)$ divides $\mathrm{lm}(\mathbf{u}_1)$. In this case, we could use $(\mathbf{u}_2, 0)$ to top-reduce p_1 by setting $t = \mathrm{lm}(\mathbf{u}_1)/\mathrm{lm}(\mathbf{u}_2)$ and $c = \mathrm{lc}(\mathbf{u}_1)/\mathrm{lc}(\mathbf{u}_2)$ in the equation (2.5). Such a top-reduction will decrease the signature of p_1 without increasing the leading term of v_1 (even if $v_1 = 0$) and is therefore always called **super**. We note that a pair $(\mathbf{u}_1, 0)$ is never top-reducible by $(\mathbf{u}_2, v_2)$ with $v_2 \neq 0$.

In our algorithm below, we only detect super top-reductions of the two kinds defined here, but never actually perform super top-reductions. We should mention that the top-reductions used in F5 correspond to regular top-reductions in our sense, but some of our regular top-reductions are not allowed in F5 (e.g. when $\mathrm{lm}(\mathbf{u}_1) = t\,\mathrm{lm}(\mathbf{u}_2)$).

We need a concept of J-pairs, similar to S-polynomials in Buchburger's algorithm. Suppose $p_1 = (\mathbf{u}_1, v_1), p_2 = (\mathbf{u}_2, v_2) \in R^m \times R$ are two pairs with v_1 and v_2 both nonzero. We form a joint pair from them as follows. Let

$$t = \mathrm{lcm}(\mathrm{lm}(v_1), \mathrm{lm}(v_2)), \quad t_1 = \frac{t}{\mathrm{lm}(v_1)}, \quad t_2 = \frac{t}{\mathrm{lm}(v_2)}.$$

Let $c = \mathrm{lc}(v_1)/\mathrm{lc}(v_2)$ and $T = \max(t_1\mathrm{lm}(\mathbf{u}_1), t_2\mathrm{lm}(\mathbf{u}_2))$, say $T = t_i\mathrm{lm}(\mathbf{u}_i)$ where $i \in \{1, 2\}$. Suppose

$$\mathrm{lm}(t_1\mathbf{u}_1 - ct_2\mathbf{u}_2) = T. \tag{2.6}$$

Then T is called the **J-signature** of p_1 and p_2, while t_ip_i is called the **J-pair** of p_1 and p_2. We do not define any J-pair for p_1 and p_2 when $\mathrm{lm}(t_1\mathbf{u}_1 - ct_2\mathbf{u}_2) \prec T$, which happens if

$$t_1\mathrm{lm}(\mathbf{u}_1) = t_2\mathrm{lm}(\mathbf{u}_2), \text{ and } \frac{\mathrm{lc}(\mathbf{u}_1)}{\mathrm{lc}(\mathbf{u}_2)} = \frac{\mathrm{lc}(v_1)}{\mathrm{lc}(v_2)}.$$

In comparison to Buchburger's algorithm, the S-polynomial of v_1 and v_2 is $t_1v_1 - ct_2v_2$. In terms of pairs, this corresponds to a reduction:

$$t_1p_1 - ct_2p_2 = (t_1\mathbf{u}_1 - ct_2\mathbf{u}_2, t_1v_1 - ct_2v_2). \tag{2.7}$$

When (2.6) holds, (2.7) is a regular top-reduction of t_ip_i by the other pair, namely $t_{3-i}p_{3-i}$. This means that the J-pair of p_1 and p_2 is defined if and only if (2.7) is a regular top-reduction. Hence the J-pair of p_1 and p_2 is always regular top-reducible by p_1 or p_2. We point out that, in the case of S-polynomials, the goal is to cancel the leading terms of v's. In our J-pair, the leading terms of v's are not cancelled, but will be cancelled in later top-reductions. Also, we never define the J-pair of $p_1 = (\mathbf{u}_1, v_1)$ and $p_2 = (\mathbf{u}_2, v_2)$ when v_1 or v_2 is zero.

Lemma 2.1. *Let t be a monomial in R and $p_1 = (\mathbf{u}_1, v_1), p_2 = (\mathbf{u}_2, v_2) \in R^m \times R$. If tp_1 is regular top-reducible by p_2, where both v_1 and v_2 are nonzero, then t_1p_1 is a J-pair of p_1 and p_2, where*

$$t_1 = \frac{lcm(lm(v_1), lm(v_2))}{lm(v_1)} = \frac{lm(v_2)}{\gcd(lm(v_1), lm(v_2))}$$

and t_1 is a divisor of t. Furthermore, t_1p_1 is regular top-reducible by p_2.

Proof. Since tp_1 is regular top-reducible by p_2 and both v_1 and v_2 are nonzero, there is a monomial s such that

$$t\,\mathrm{lm}(v_1) = s\,\mathrm{lm}(v_2), \quad t\,\mathrm{lm}(\mathbf{u}_1) = \mathrm{lm}(t\,\mathbf{u}_1 - cs\,\mathbf{u}_2), \tag{2.8}$$

where $c = \mathrm{lm}(v_1)/\mathrm{lm}(v_2)$. Let

$$t_2 = \frac{\mathrm{lcm}(\mathrm{lm}(v_1), \mathrm{lm}(v_2))}{\mathrm{lm}(v_2)} = \frac{\mathrm{lm}(v_1)}{\gcd(\mathrm{lm}(v_1), \mathrm{lm}(v_2))}.$$

Then the first equation of (2.8) implies that, for some monomial w,

$$t = \frac{\mathrm{lm}(v_2)}{\gcd(\mathrm{lm}(v_1), \mathrm{lm}(v_2))} w = t_1w, \text{ and}$$
$$s = \frac{\mathrm{lm}(v_1)}{\gcd(\mathrm{lm}(v_1), \mathrm{lm}(v_2))} w = t_2w.$$

Hence the second equation of (2.8) implies that $t_1\,\mathrm{lm}(\mathbf{u}_1) = \mathrm{lm}(t_1\mathbf{u}_1 - ct_2\mathbf{u}_2)$. This shows that t_1p_1 is the J-pair of p_1 and p_2, and t_1p_1 is regular top-reducible by p_2. □

A subset G of M is called a **strong Gröbner basis for** M if every pair M is top-reducible by some pair in G.

Proposition 2.2. *Suppose that $G = \{(\mathbf{u}_1, v_1), (\mathbf{u}_2, v_2), \ldots, (\mathbf{u}_k, v_k)\}$ is a strong Gröbner basis for M (where k could be infinite). Then*

1. *$\mathbf{G}_0 = \{\mathbf{u}_i : v_i = 0, 1 \le i \le k\}$ is a Gröbner basis for the syzygy module of $\mathbf{g} = (g_1, \ldots, g_m)$, and*
2. *$G_1 = \{v_i : 1 \le i \le k\}$ is a Gröbner basis for $I = \langle g_1, \ldots, g_m \rangle$.*

Proof. For any $\mathbf{u} = (u_1, \dots, u_m)$ in the syzygy module of $\mathbf{g}$, we have $(\mathbf{u}, 0) \in M$. By our assumption, $(\mathbf{u}, 0)$ is top-reducible by some pair $(\mathbf{u}_i, v_i)$ in G. Then we must have $v_i = 0$, thus $\mathbf{u}_i \in G_0$ and $\text{lm}(\mathbf{u})$ is reducible by $\text{lm}(\mathbf{u}_i)$. This proves that G_0 is a Gröbner basis for the syzygy module of $\mathbf{g}$.

Now suppose $v \in I$ and is nonzero. Then there exists $\mathbf{u} = (u_1, \dots, u_m) \in R^m$ so that $\mathbf{u}\mathbf{g}^t = v$, hence $(\mathbf{u}, v) \in M$. Among all such $\mathbf{u}$, we pick one so that $\text{lm}(\mathbf{u})$ is minimum. Since $(\mathbf{u}, v) \in M$, it is top-reducible by some $(\mathbf{u}_i, v_i)$ where $1 \le i \le k$. If $v_i = 0$, then we could use $(\mathbf{u}_i, 0)$ to reduce $(\mathbf{u}, v)$ to get a $\mathbf{u}'$ so that $\mathbf{u}'\mathbf{g}^t = v$ and $\text{lm}(\mathbf{u}')$ is smaller than $\text{lm}(\mathbf{u})$, contradicting to the minimality of $\text{lm}(\mathbf{u})$. So $v_i \neq 0$ and $\text{lm}(v_i)$ divides $\text{lm}(v)$. Hence G_1 is a Gröbner basis for I. □

Remark. Note that $M \subset R^m \times R$ has a Gröbner basis in the usual sense as a submodule of R^{m+1} where the leading term of $(\mathbf{u}, v)$ is $\text{lm}(v)\mathbf{E}_{m+1}$ if $v \neq 0$ and $\text{lm}(\mathbf{u})$ if $v = 0$. The above proposition implies that a strong Gröbner basis for M is a Gröbner basis for M as a submodule of R^{m+1}, but the converse may not be true for an arbitrary submodule M of R^{m+1} (as our regular top-reduction must preserve signatures). This is why we call our basis a strong Gröbner basis.

Let S be any set of pairs in $R^m \times R$. We say that a pair $(\mathbf{u}, v) \in R^m \times R$ is regular top-reducible by S if it is regular top-reducible by at least one pair in S. We call $(\mathbf{u}, v)$ **eventually super top-reducible** by S if there is a sequence of regular top-reductions of $(\mathbf{u}, v)$ by pairs in S that reduce $(\mathbf{u}, v)$ to a pair $(\mathbf{u}', v')$ that is no longer regular top-reducible by S but is super top-reducible by at least one pair in S.

THEOREM 2.3. *Suppose G is a subset of M such that, for any term $T \in R^m$, there is a pair $(\mathbf{u}, v) \in G$ and a monomial t such that $T = t\,\text{lm}(\mathbf{u})$. Then the following are equivalent:*

- (a) *G is a strong Gröbner basis for M,*
- (b) *every J-pair of G is eventually super top-reducible by G,*
- (c) *for every J-pair $(\mathbf{u}, v)$ of G, there is a pair $(\mathbf{u}_1, v_1) \in G$ so that $\text{lm}(\mathbf{u}_1)$ divides $\text{lm}(\mathbf{u})$ and $t\,\text{lm}(v_1) \prec \text{lm}(v)$ where $t = \text{lm}(\mathbf{u})/\text{lm}(\mathbf{u}_1)$.*

Proof. $(a) \Rightarrow (b)$ Let $p = (\mathbf{u}, v)$ be any J-pair of G. Then p is in M, hence top-reducible by G. We can perform regular top-reductions to p as much as possible, say to get $p' = (\mathbf{u}', v')$ which is not regular top-reducible. Since p' is still in M, it is top-reducible by G, hence must be super top-reducible by G. Therefore, p is eventually super top-reducible by G.

$(b) \Rightarrow (c)$ Let $p = (\mathbf{u}, v)$ be any J-pair from G. Since p is eventually super top-reducible by G, after a sequence of regular top-reductions of p by G, we can get a $p_0 = (\mathbf{u}_0, v_0) \in M$ such that p_0 is not regular top-reducible by G but is super top-reducible by some pair $p_1 = (\mathbf{u}_1, v_1) \in G$.

If $v_1 = 0$, then $\text{lm}(\mathbf{u}_1) \mid \text{lm}(\mathbf{u}_0) = \text{lm}(\mathbf{u})$ and $tv_1 = 0$ is smaller than $\text{lm}(v)$. So we may assume that $v_1 \neq 0$. Then

$$\frac{\text{lm}(v_0)}{\text{lm}(v_1)} = \frac{\text{lm}(\mathbf{u}_0)}{\text{lm}(\mathbf{u}_1)},$$

which is denoted by t. Note that every J-pair can be regular top-reduced by G, so we have $\text{lm}(v_0) < \text{lm}(v)$ and $\text{lm}(\mathbf{u}_0) = \text{lm}(\mathbf{u})$, the latter implies that

$$t\,\text{lm}(v_1) = \text{lm}(v_0) \prec \text{lm}(v).$$

Hence we have $\text{lm}(\mathbf{u}_1) \mid \text{lm}(\mathbf{u}_0)$ and $t\,\text{lm}(v_1) \prec \text{lm}(v)$ as desired. This shows that (c) is satisfied.

$(c) \Rightarrow (a)$. We prove by contradiction. Assume that there is a pair $p = (\mathbf{u}, v) \in M$ that is not top-reducible by any pair in G. Among all such pairs p we pick one with minimal signature $T = \mathrm{lm}(\mathbf{u})$. Note that $T \neq \mathbf{0}$. Next, we select a pair $p_1 = (\mathbf{u}_1, v_1)$ from G such that

(*i*) $T = t\,\mathrm{lm}(\mathbf{u}_1)$ for some monomial t, and

(*ii*) $t\,\mathrm{lm}(v_1)$ is minimal among all $p_1 \in G$ satisfying (*i*).

We claim that $t(\mathbf{u}_1, v_1)$ is not regular top-reducible by G. To prove this claim, we suppose that $t(\mathbf{u}_1, v_1)$ is regular top-reducible by some $p_2 = (\mathbf{u}_2, v_2) \in G$, so both v_1 and v_1 are nonzero. We want to derive a contradiction to the condition (*ii*). By Lemma 2.1, the J-pair of p_1 and p_2 is $t_1(\mathbf{u}_1, v_1)$ and that $t_1 p_1$ is still regular top-reducible by p_2, where

$$t_1 = \frac{\mathrm{lcm}(\mathrm{lm}(v_1), \mathrm{lm}(v_2))}{\mathrm{lm}(v_1)}, \text{ and } t = t_1 w$$

for some monomial w. As $t_1 p_1$ is a J-pair of G, there is a pair $p_3 = (\mathbf{u}_3, v_3) \in G$ so that $t_3 \mathrm{lm}(v_3) \prec t_1 \mathrm{lm}(v_1)$, where $t_3 = t_1 \mathrm{lm}(\mathbf{u}_1)/\mathrm{lm}(\mathbf{u}_3)$ is a monomial. Then we have

$$T = t\,\mathrm{lm}(\mathbf{u}_1) = w t_1 \mathrm{lm}(\mathbf{u}_1) = w t_3 \mathrm{lm}(\mathbf{u}_3),$$

and

$$w t_3 \mathrm{lm}(v_3) \prec w t_1 \mathrm{lm}(v_1) = t\,\mathrm{lm}(v_1).$$

This violates the condition (*ii*) for the choice of p_1 in G.

Hence we may assume that $t(\mathbf{u}_1, v_1)$ is not regular top-reducible by G. Consider

$$(\overline{\mathbf{u}}, \overline{v}) = (\mathbf{u}, v) - ct(\mathbf{u}_1, v_1), \tag{2.9}$$

where $c = \mathrm{lc}(\mathbf{u})/\mathrm{lc}(\mathbf{u}_1)$ so that $\mathrm{lm}(\overline{\mathbf{u}}) \prec \mathrm{lm}(\mathbf{u}) = T$. Note that $\mathrm{lm}(v) \neq t\,\mathrm{lm}(v_1)$, since otherwise $(\mathbf{u}, v)$ would be top-reducible by p_1 contradicting the choice of $(\mathbf{u}, v)$. Also, as $(\overline{\mathbf{u}}, \overline{v}) \in M$ and $\mathrm{lm}(\overline{\mathbf{u}}) \prec T$, we have that $(\overline{\mathbf{u}}, \overline{v})$ is top-reducible by G. If $(\overline{\mathbf{u}}, \overline{v})$ is top-reducible by some pair $p_2 = (\mathbf{u}_2, v_2) \in G$ with $v_2 = 0$, then we can reduce $(\overline{\mathbf{u}}, \overline{v})$ repeatedly by such pairs to get a new pair $(\tilde{\mathbf{u}}, \overline{v})$ that is not top-reducible by any pair in G with v-part being zero. Note that $(\tilde{\mathbf{u}}, \overline{v})$ is still in M and $\mathrm{lm}(\tilde{\mathbf{u}}) \prec T$. Hence $(\tilde{\mathbf{u}}, \overline{v})$ is top-reducible by some pair $p_2 = (\mathbf{u}_2, v_2) \in G$ with $v_2 \neq 0$. As $\mathrm{lm}(v) \neq t\,\mathrm{lm}(v_1)$, we consider two cases:

- $\mathrm{lm}(v) \prec t\,\mathrm{lm}(v_1)$. Then $\mathrm{lm}(\overline{v}) = t\,\mathrm{lm}(v_1)$, hence $t(\mathbf{u}_1, v_1)$ is regular top-reducible by $(\mathbf{u}_2, v_2)$ (as $\mathrm{lm}(\tilde{\mathbf{u}}) \prec t\,\mathrm{lm}(\mathbf{u}_1)$). Since $t(\mathbf{u}_1, v_1)$ is not regular top-reducible by any pair in G, this case is impossible.
- $\mathrm{lm}(v) \succ t\,\mathrm{lm}(v_1)$. Then $\mathrm{lm}(\overline{v}) = \mathrm{lm}(v)$, and $(\mathbf{u}, v)$ is regular top-reducible by $(\mathbf{u}_2, v_2)$, contradicting the fact that $(\mathbf{u}, v)$ is not top-reducible by any pair in G.

Therefore such a pair $(\mathbf{u}, v)$ does not exist in M, so every pair in M is top-reducible by G. This proves (a). □

Remarks. (*i*) In the original version of this paper, Theorem 2.3 had only (a) and (b). Later, in early November 2010, Huang [14] discovered (c) in a slightly different form as M-pair Criterion, and in December 2010, Arri and Perry [1] discovered (c) as F5 Criterion where the polynomials in G must be S-irreducible in M (not just in G). We decided to include (c) here for three reasons: (1) (c) was actually proved in the original proof of the equivalence of (a) and (b) (the current proof is just a rewording of that proof), (2) our statement and proof are much simpler (compare with Theorem 18 in [1]), and (3) Theorem 2.3 allows J-pairs to be processed in any order, not just in increasing signature order; which is not true for the algorithms of Huang [14] and Arri [1]. For more details, see the comments on related recent works in the last paragraph in Section 5.

(*ii*) For a subset G of M, there are usually many J-pairs of G with the same signature T. We don't need to check the condition (b) or (c) for every such J-pair. Instead, we just store one J-pair whose v-part is minimal and check if this J-pair satisfies (c), which would imply that all other J-pairs also satisfy (c).

(*iii*) Suppose a final strong Gröbner basis for M is

$$(\mathbf{u}_1, v_1), \ldots, (\mathbf{u}_k, v_k). \tag{2.10}$$

At any intermediate step of computation, we only know

$$(\mathbf{u}_1, v_1), \ldots, (\mathbf{u}_p, v_p) \tag{2.11}$$

for some $p < k$. In general, a pair $(\mathbf{u}, v)$ may be eventually super top-reducible by (2.11) but not by (2.10). How can one decide whether $(\mathbf{u}, v)$ is eventually super top-reducible by (2.10) when only (2.11) is known? Similarly for the condition (c). Our strategy is to always pick the J-pair with minimal signature to reduce. Then a pair that is eventually super top-reducible by an intermediate basis is always eventually super top-reducible by the final basis. A more detailed argument will be given in the next section.

3. Algorithm, Term Orderings and Time Comparison. Algorithm and Its Correctness. Our algorithm is based on Theorem 2.3. The basic idea is as follows. Initially, we have the pairs in (2.4) in our Gröbner basis. So the condition of the theorem is satisfied. From these pairs, we form all J-pairs, keeping only one J-pair for each J-signature (the one whose v-part is minimal). We then take the smallest J-pair from the list of J-pairs. Check if the condition (c) is satisfied for this pair. If yes, discard this J-pair; otherwise, repeatedly perform regular top-reductions to this pair until it is no longer regular top-reducible, say to get $(\mathbf{u}, v)$. If the v part of the resulting pair is zero, then the $\mathbf{u}$ part is a syzygy in $\mathbf{H}$, and we store this vector. If the v part is nonzero, then add this $(\mathbf{u}, v)$ pair to the current Gröbner basis and form new J-pairs. Repeat this process until the list of J-pairs is empty.

We make two improvements on this basic algorithm. First, storing and updating syzygies $\mathbf{u} \in \mathbf{H}$ are expensive. In our computation, we shall make all pairs $(\mathbf{u}, v)$ monic, namely, the leading coefficient of $\mathbf{u}$ is 1. Now suppose $(\mathbf{u}_1, v_1)$ and $(\mathbf{u}_2, v_2)$ are any two monic pairs. Then a top-reduction (regular or super) is determined only by $\mathrm{lm}(\mathbf{u}_1)$, $\mathrm{lm}(\mathbf{u}_2)$, v_1 and v_2. The other terms of $\mathbf{u}_1$ and $\mathbf{u}_2$ are not used at all. Let $T_1 = \mathrm{lm}(\mathbf{u}_1)$ and $T_2 = \mathrm{lm}(\mathbf{u}_2)$, the signatures of $(\mathbf{u}_1, v_1)$ and $(\mathbf{u}_2, v_2)$, respectively. Suppose we store only (T_1, v_1) and (T_2, v_2). Then (T_1, v_1) is regular top-reducible by (T_2, v_2) when $v_2 \neq 0$, $\mathrm{lm}(v_1)$ is divisible by $\mathrm{lm}(v_2)$, $tT_2 \prec T_1$, or $tT_2 = T_1$ but $\mathrm{lc}(v_1) \neq \mathrm{lc}(v_2)$. The corresponding top-reduction is

$$v := v_1 - ctv_2$$

where $t = \mathrm{lm}(v_1)/\mathrm{lm}(v_2)$ and $c = \mathrm{lc}(v_1)/\mathrm{lc}(v_2)$, and furthermore, if $tT_2 = T_1$ then we update v as

$$v := v/(1 - c),$$

to keep the $\mathbf{u}$-part monic of $(\mathbf{u}, v)$ where $T_1 = \mathrm{lm}(\mathbf{u})$. Then (T_1, v) is the resulting pair of the reduction, and it replaces (T_1, v_1). Our algorithm below will perform regular top-reductions in this fashion.

Another improvement is to use trivial syzygies. We will store the leading terms of known syzygies in a list called H. Let (T_1, v_1) and (T_2, v_2) be any two pairs from the Gröbner basis computed so far, where v_1 and v_2 are both nonzero. Then, for $1 \leq i \leq 2$, there are $\mathbf{u}_i \in R^m$ such that $\mathrm{lm}(\mathbf{u}_i) = T_i$ and $(\mathbf{u}_i, v_i) \in M$. Then we have

$$v_2(\mathbf{u}_1, v_1) - v_1(\mathbf{u}_2, v_2) = (v_2\mathbf{u}_1 - v_1\mathbf{u}_2, 0) \in M.$$

Hence $v_2\mathbf{u}_1 - v_1\mathbf{u}_2$ is a syzygy of $(g_1, \ldots, g_m)$. Its leading term is

$$T = \max(T_1\mathrm{lm}(v_2), T_2\mathrm{lm}(v_1)),$$

provided that $T_1\mathrm{lm}(v_2) \neq T_2\mathrm{lm}(v_1)$ or $T_1\mathrm{lm}(v_2) = T_2\mathrm{lm}(v_1)$ but $\mathrm{lc}(v_1) \neq \mathrm{lc}(v_2)$. When $T_1\mathrm{lm}(v_2) = T_2\mathrm{lm}(v_1)$ and $\mathrm{lc}(v_1) = \mathrm{lc}(v_2)$, the leading terms in $v_2(\mathbf{u}_1, v_1)$ and $v_1(\mathbf{u}_2, v_2)$ cancel each other. In that case, we don't know the leading term of the syzygy, so we just ignore such a syzygy. In all other cases, our algorithm will add T to the list H. The benefit of H is in detecting useless reductions. That is, whenever a J-pair has a signature that is divisible by a term in H, it is always eventually super top-reducible and hence discarded, thus saving time.

The algorithm is described more precisely in Figure 3.1 below. As mentioned above, we use H to record leading terms of syzygies. In addition to H, our algorithm uses two more lists to store the pairs $(T_1, v_1), (T_2, v_2), \ldots, (T_k, v_k)$ with $v_i \neq 0$ for $1 \leq i \leq k$. This list will be stored as

$$U = [T_1, T_2, \ldots, T_k], \quad V = [v_1, v_2, \ldots, v_k].$$

Then $[U, V]$ represents the whole list $(T_1, v_1), (T_2, v_2), \ldots, (T_k, v_k)$.

THEOREM 3.1. *If the algorithm in Figure 3.1 terminates, then V is a Gröbner basis for $I = \langle g_1, g_2, \ldots, g_m \rangle$ and H is a Gröbner basis for the leading terms of the syzygy module of $(g_1, g_2, \ldots, g_m)$.*

Proof. To prove the correctness of the algorithm, we need to show the following:

- (i) One can delete J-pairs in Steps 4a, and 4b whose signatures are divisible by $\mathrm{lm}(\mathbf{u})$, where $\mathbf{u} \in H$.
- (ii) A pair that is eventually super top-reducible by an intermediate basis will always be eventually super top-reducible by the final basis.
- (iii) One just needs to keep one J-pair for each signature, which follows directly from Theorem 2.3(c).

Our current basis consists of pairs in $[U, V]$ and $[H, 0]$. For (i), let $(\mathbf{u}, v)$ be any pair whose signature $\mathrm{lm}(\mathbf{u})$ is divisible by $\mathrm{lm}(\mathbf{u}')$ for some $\mathbf{u}' \in H$. Then $(\mathbf{u}, v)$ is top-reducible by $(\mathbf{u}', 0)$. Any regular top-reduction of $(\mathbf{u}, v)$ won't change $\mathrm{lm}(\mathbf{u})$, so the pair obtained from $(\mathbf{u}, v)$ by any sequence of regular top-reductions will be super top-reducible by $(\mathbf{u}', 0)$. Hence $(\mathbf{u}, v)$ is eventual super top-reducible by the current basis. This means that we don't need to reduce $(\mathbf{u}, v)$, and so we simply discard it.

To see (ii), suppose the final Gröbner basis computed for M is

$$(\mathbf{u}_1, v_1), \ldots, (\mathbf{u}_k, v_k), \tag{3.1}$$

while at any intermediate step, we only know

$$(\mathbf{u}_1, v_1), \ldots, (\mathbf{u}_p, v_p) \tag{3.2}$$

```
                Algorithm for computing Gröbner bases
Input:      $g_1, \ldots, g_m \in R = \mathbb{F}[x_1, \ldots, x_n]$ and term orders for $R$ and $R^m$
Output:     A Gröbner basis for $I = \langle g_1, \ldots, g_m \rangle$ and a Gröbner basis for
            $\mathrm{lm}(\mathbf{H})$, the leading terms of the syzygy module
Variables:  $U$ a list of terms $T_i$, representing signatures of $(\mathbf{u}_i, v_i) \in M$,
            $V$ a list of polynomials for $v_i$ for $(\mathbf{u}_i, v_i) \in M$,
            $H$ a list for $\mathrm{lm}(\mathbf{u})$ were $\mathbf{u} \in R^m$ is a syzygy found so far,
            $JP$ a list of pairs $(x^\alpha T_i, x^\alpha v_i)$, where $x^\alpha$ is a monomial so that
            $x^\alpha(\mathbf{u}_i, v_i)$ is a J-pair of $(\mathbf{u}_i, v_i)$ and $(\mathbf{u}_j, v_j)$ for some $j \neq i$.
Step 0.     $U = [\ ]$, $V = [\ ]$, and $H = [\ ]$ are all empty lists.
            $JP = [(\mathbf{E}_1, g_1), \ldots, (\mathbf{E}_m, g_m)]$.
Step 1.     Take a minimal (in signature) pair $(T, v_1)$ from $JP$, and delete it
            from $JP$.
Step 2.     If $(T, v_1)$ satisfies Theorem 2.3($c$) with $G = [U, V]$, then discard
            $(T, v_1)$ and go to step 5.
Step 3.     Reduce the pair $(T, v_1)$ repeatedly and as much as possible by the
            pairs in $[U, V]$ using only regular top-reductions, say to get $(T, v)$.
Step 4a.    If $v = 0$, then append $T$ to $H$, and delete every J-pair $(T_2, v_2)$ in
            $JP$ whose signature $T_2$ is divisible by $T$.
Step 4b.    If $v \neq 0$ and $(T, v)$ is not super top-reducible by $[U, V]$, then
              i) Add the leading terms of the principle syzygies, $vT_j - v_jT$ for
                 $1 \leq j \leq |U|$, to $H$,
             ii) Form new J-pairs of $(T, v)$ and $(T_j, v_j)$, $1 \leq j \leq |U|$,
            iii) Insert into $JP$ all such J-pairs whose signatures are not re-
                 ducible by $H$ (storing only one J-pair for each distinct signature
                 $T$, the one with $v$-part minimal), and
             iv) Append $T$ to $U$ and $v$ to $V$.
Step 5.     While JP is not empty, go to step 1.
Return:     $V$ and $H$.
```

FIG. 3.1. *Main algorithm*

for some $p < k$. Suppose that the smallest J-pair from JP is (t, i) (i.e., $t(u_i, v_i)$). If $t(u_i, v_i)$ is eventually super top-reducible by (3.2), then $t(u_i, v_i)$ remains eventually super top-reducible by (3.1), as all $(\mathbf{u}_j, v_j)$, $j > p$, have strictly larger signature than $t(u_i, v_i)$. If $t(u_i, v_i)$ is not eventually super top-reducible by (3.2), then the basis (3.2) is augmented by a new pair $(\mathbf{u}_{p+1}, v_{p+1})$, which is obtained from $t(u_i, v_i)$ via regular top-reductions by (3.2). Hence the J-pair $t(u_i, v_i)$ is eventually super top-reducible by the new basis

$$(\mathbf{u}_1, v_1), \ldots, (\mathbf{u}_p, v_p), (\mathbf{u}_{p+1}, v_{p+1}). \tag{3.3}$$

Note that $(\mathbf{u}_{p+1}, v_{p+1})$ has the same signature as the J-pair $t(\mathbf{u}_i, v_i)$. All new J-pairs formed using $(\mathbf{u}_{p+1}, v_{p+1})$ will have strictly greater signature than that of $(\mathbf{u}_{p+1}, v_{p+1})$ (this is true exactly when $(\mathbf{u}_{p+1}, v_{p+1})$ is fully regular top-reduced with respect to (3.2)). Hence $(\mathbf{u}_{p+1}, v_{p+1})$ can not be top-reducible by any pair $(\mathbf{u}_j, v_j)$, $j > p + 1$, so the J-pair $t(u_i, v_i)$ remains eventually super top-reducible by (3.1). Therefore, any pair that is eventually super top-reducible by our current basis remains so by the final basis. □

Remark. If, in Step 4b, the super top-reducibility of (T, v) is not checked, then one can process J-pairs in *any order*, not necessarily in increasing signature order.

The correctness of the algorithm follows directly from Theorem 2.3.

Finite termination. Our algorithm allows arbitrary term orders $\prec_1$ on R and and $\prec_2$ on R^m. However, it is not clear when it has finite termination. We left this as an open problem in the original version. Later, Huang [14] proved the following nice result.

THEOREM 3.2 (Huang [14]). *The algorithm in Figure 3.1 terminates in finitely many steps if and only if the term orders $\prec_1$ on R and $\prec_2$ on R^m are compatible, which means that $x^\alpha \prec_1 x^\beta$ if and only if $x^\alpha \mathbf{E}_i \prec_2 x^\beta \mathbf{E}_i$ for all $1 \le i \le m$.*

For more details, the interested reader is referred to Huang's paper. Also, we would like to mention that the proofs of finite termination in Hashemi and Ars [13] and Arri and Perry [1] have flaws. In [13], the proof of Proposition 4.1 assumes that the each time a new polynomial is added to the current Gröbner basis, the ideal generated by its leading terms strictly increases (just like Buchburger's algorithm). This is not true in general, as a polynomial may be reducible by the current Gröbner basis in the sense of Buchburger's algorithm but such a reduction may not preserve signature hence not allowed in F5 algorithm.

In [1], they claim finite termination for any term orders $\prec_1$ on R and $\prec_2$ on R^m. That is not correct by Huang's result. Assuming that the two orders are compatible, their proof of Proposition 14 is still flawed. More precisely, they assumed that if an R-module N of $R \times R^m$ is generated by a set of elements of the form

$$(x^{\alpha_j}, x^{\beta_j} E_{i_j}), \quad j = 1, 2, \ldots,$$

then, for every element $(v, \mathbf{u}) \in N$, the element $(\mathrm{lm}(v), \mathrm{lm}(\mathbf{u}))$ is divisible by one of the generators, that is, there is a monomial $t \in R$ and some j so that

$$(\mathrm{lm}(v), \mathrm{lm}(\mathbf{u}) = t\,(x^{\alpha_j}, x^{\beta_j} E_{i_j}).$$

This is not true in general. Here's a counterexample. Let $R = \mathbb{F}[x, y]$ under lex with $x > y$ and R^2 under POT order with $E_1 = (1, 0) > E_2 = (0, 1)$. Consider the R-submodule N generated by

$$(x, E_1), \quad (x, E_2), \quad (y, E_2).$$

Then $(y, E_1) = (x, E_1) - (x, E_2) + (y, E_2) \in N$, but (y, E_1) is not divisible by any of the three generators.

Gröbner bases for the syzygy module. Our algorithm as presented in Figure 3.1 only calculates the leading terms of the syzygy module. While one has the option of modifying the algorithm to compute syzygies instead of leading terms of syzygies, there is a more efficient way. Suppose that the algorithm terminates with lists U, V and H, then we can compute a minimal Gröbner basis for the syzygy module as follows. The m pairs $(\mathbf{E}_i, g_i)$, $1 \le i \le m$, are already in M. Among these pairs, we need to perform regular top-reductions until no one is regular top-reducible by any others. Then we have m pairs

$$(\mathbf{u}_1, v_1), \ldots, (\mathbf{u}_m, v_m) \in M$$

whose signatures are $\mathbf{E}_1, \ldots, \mathbf{E}_m$, respectively, and none of them is regular top-reducible by others in the list. Now order the signatures in $U \setminus \{\mathbf{E}_1, \ldots, \mathbf{E}_m\}$ in increasing order, say

$$T_{m+1}, \ldots, T_\ell.$$

For i from $m+1$ to ℓ, find $j < i$ and a monomial t so that $T_i = t\,\mathrm{lm}(\mathbf{u}_j)$ and $t\,\mathrm{lm}(v_j)$ is minimal, and perform regular top-reductions of $t(\mathbf{u}_j, v_j)$ by

$$(\mathbf{u}_1, v_1), \ldots, (\mathbf{u}_m, v_m), \ldots, (\mathbf{u}_{i-1}, v_{i-1}),$$

until it is not regular top-reducible. Denote the resulted pair by $(\mathbf{u}_i, v_i)$ and proceed to the next i. By the end of this loop, we get ℓ pairs

$$(\mathbf{u}_1, v_1), (\mathbf{u}_2, v_2), \ldots, (\mathbf{u}_m, v_m), \ldots, (\mathbf{u}_\ell, v_\ell) \tag{3.4}$$

in M, whose signatures are exactly those in U.

To get a Gröbner basis for the syzygy module, just do the following. For each term T in H, we recover the $\mathbf{u}$ such that $\mathbf{u}\mathbf{g}^t = 0$ and $\mathrm{lm}(\mathbf{u}) = T$. Find a pair $(\mathbf{u}_i, v_i)$, $1 \le i \le \ell$, so that $T = t\,\mathrm{lm}(\mathbf{u}_i)$ and $t\,\mathrm{lm}(v_i)$ is minimal. Then perform regular top-reductions of $t(\mathbf{u}_i, v_i)$ by (3.4) until the v-part is zero and the $\mathbf{u}$-part is a syzygy with leading term equal to T. If T comes from a trivial syzygy, then no reductions are required. All these syzygies form a minimal Gröbner basis for the $(g_1, \ldots, g_m)$-syzygy module with respect to ordering $\prec_2$.

This algorithm takes advantage of the signatures already computed in U and H, thus saving time that would be used in processing J-pairs and reducing J-pairs that are eventually super top-reducible.

Term Orders. Now we discuss choices of term orders. We use $\prec_1$ to represent a term ordering on R and $\prec_2$ to represent a term ordering on R^m. While computing Gröbner bases for both $\langle g_1, \ldots, g_m\rangle$ and $\mathbf{H}$, one should set $\prec_1$ and $\prec_2$ to the appropriate term orderings for the Gröbner bases desired. Often, however, the Gröbner basis for $\mathbf{H}$ is not needed. Then we only need the leading terms of $\mathbf{H}$ to speed up the computation of $\langle g_1, \ldots, g_m\rangle$. In this case, we have tremendous freedom in the choice of $\prec_2$.

There are many ways that we can construct a term ordering on R^m. We consider four extreme cases below. Let $\prec$ be some term order on R. We extend $\prec$ to R^m as follows.

(POT) The first is called position over term ordering (POT). We say that $x^\alpha \mathbf{E}_i \prec x^\beta \mathbf{E}_j$ if $i < j$ or $i = j$ and $x^\alpha \prec x^\beta$.

(TOP) The second is the term over position ordering (TOP). We say that $x^\alpha \mathbf{E}_i \prec x^\beta \mathbf{E}_j$ if $x^\alpha \prec x^\beta$ or $x^\alpha = x^\beta$ and $i < j$.

($\mathbf{g}1$) Next is the $\mathbf{g}$-weighted degree followed by TOP. We say that $x^\alpha \mathbf{E}_i \prec x^\beta \mathbf{E}_j$ if $\deg(x^\alpha g_i) < \deg(x^\beta g_j)$ or $\deg(x^\alpha g_i) = \deg(x^\beta g_j)$ and $x^\alpha \mathbf{E}_i \prec_{top} x^\beta \mathbf{E}_j$, where deg is for total degree.

($\mathbf{g}2$) Finally, we have $\mathbf{g}$-weighted $\prec$ followed by POT. We say that $x^\alpha \mathbf{E}_i \prec x^\beta \mathbf{E}_j$ if $\mathrm{lm}(x^\alpha g_i) \prec \mathrm{lm}(x^\beta g_j)$ or $\mathrm{lm}(x^\alpha g_i) = \mathrm{lm}(x^\beta g_j)$ and $x^\alpha \mathbf{E}_i \prec_{pot} x^\beta \mathbf{E}_j$.

These signature orders are compatible with the order in R, hence our algorithm has finite termination by Huang's result [14]. We remark that, under the POT order, our new algorithm closely corresponds to the G2V algorithm presented in [12]. The reason being that this new algorithm always first picks J-pairs with signatures containing $\mathbf{E}_1$, then those with $\mathbf{E}_2$, etc. This means that it computes Gröbner bases for $\langle g_1\rangle$, $\langle g_1, g_2\rangle$, …, $\langle g_1, g_2, \ldots, g_m\rangle$, just like G2V and F5. The only difference is that the intermediate bases may not be reduced and non-leading terms are not reduced as in the computing of normal forms.

Another remark is that our algorithm under the $\mathbf{g}1$ order roughly corresponds to the behavior of the F4 and XL algorithms [5]. In the XL algorithm, one performs row reductions on a matrix whose rows correspond to all polynomials $x^\alpha g_i$, $1 \le i \le m$,

Test Case (#generators)	F5	F5C	G2V
Katsura5 (22)	1.48	0.93	0.36
Katsura6 (41)	2.79	2.34	0.37
Katsura7 (74)	30.27	22.76	4.64
Katsura8 (143)	290.97	177.74	29.88
Schrans-Troost (128)	1180.08	299.65	21.34
F633 (76)	30.93	29.87	2.06
Cyclic6 (99)	28.44	22.06	5.65
Cyclic7 (443)	4591.20	2284.05	732.33

Table 3.1

Runtimes in seconds comparing F5, F5C and G2V (GVW under POT ordering) for various test cases in Singular 3110 on an Intel Core 2 Quad 2.66 GHz. This table is reproduced from [12].

Test Case	F5	F5C	G2V	POT	TOP	**g**1	**g**2
Katsura5 (22)	79	66	64	67	64	64	39
Katsura6 (41)	103	77	69	73	97	97	55
Katsura7 (74)	280	218	216	224	189	189	101
Katsura8 (143)	691	492	439	448	368	368	191
Schrans-T (128)	1379	813	461	398	208	208	220
F633 (76)	420	362	288	164	237	225	150
Cyclic 6 (99)	451	338	411	163	1209	1209	216
Cyclic 7 (443)	3905	2581	3108	785	9322	9322	974

Table 3.2

Counts of the J-pairs or S-polynomials processed by F5, F5C (as in [7]), G2V (as in [12]), and GVW under POT, TOP, **g**1 *and* **g**2 *orders*

with total degree of $x^\alpha g_i$ smaller than some bound. Our algorithm basically works with only some of those rows that correspond to J-signatures. So our algorithm needs much less storage.

Performance Comparison. For ease of exposition, we refer to our algorithm as GVW. We implemented GVW in C++ so that $\langle g_1, \ldots, g_m\rangle$ is computed in one-shot, that is, non-incrementally. Because our C++ implementation is vastly different than our F5/C and G2V implementations, we did not compare timings as we did in [12] (see Table 3.1, reproduced here[1] for comparison purposes). Instead, table 3.2 lists the counts of J-pairs or S-polynomials processed by each algorithm. Within Table 3.2, we distinguish between the G2V (as in [12], without theorem 2.3(c)) and GVW under the POT order. But as mentioned earlier, GVW under POT is nearly the G2V algorithm except for the interreduction between increments and theorem 2.3(c).

Just as in [12], various benchmark examples (from [7]) were run for comparison. We collected data from each example under each term ordering for comparison. Table 3.4 list the runtimes in seconds of GVW for each of the four term orderings. In examining the timings, we find that **g2** seems to be a clear winner among the four term orders.

A more computer independent measure would be a count of J-pairs processed and the number of extraneous generators produced. Table 3.2 lists the total number of J-pairs processed for each term ordering. It's analogous to counting the number

[1] with permission from ACM.

Test Case (# gen)	POT/G2V	TOP	**g**1	**g**2
Katsura5 (22)	0	0	0	0
Katsura6 (41)	0	0	0	0
Katsura7 (74)	0	0	0	0
Katsura8 (143)	0	0	0	0
Schrans-Troost (128)	0	0	0	0
F633 (76)	0	0	0	0
Cyclic 6 (99)	0	0	0	0
Cyclic 7 (443)	0	0	0	0

Table 3.3
A count of the super top-reductions (all discarded)

of S-polynomials processed in F5 or Buchberger's algorithm. As with the timings, **g**2 seems to be the most efficient. We remark that in [12], it was observed that G2V and F5 performed very similarly in terms of J-pairs/S-polynomials processed. Therefore, GVW under the **g**2 order with theorem 2.3(c) tends to process fewer J-pairs/S-polynomials.

Table 3.5 lists the sizes of the Gröbner bases produced by GVW with each term ordering. These are the Gröbner bases produced by the algorithm before any interreduction occurs to produce a reduced Gröbner basis. We believe this measure to be significant since fewer extraneous generators means quicker reductions. Again, we see that **g**2 produces less redundancy than the other orderings. In fact, the parenthetical values of each table shows the size of a minimal Gröbner basis for the ideal $\langle g_1, \ldots, g_m\rangle$. Table 3.5 shows that GVW under POT is producing Gröbner bases that are close to minimal. Finally, we mention that as presented in table 3.3, there were no super top-reductions for the examples considered.

One might make the observation that in [12] (or Table 3.1), G2V outperformed F5 and F5C by runtimes of 2 to 10 times[2], while with the present algorithm, GVW under the **g**2 ordering outperforms G2V (GVW under the POT ordering) by another factor of 2 to 10 times. This comparison shows that if GVW under **g**2 were implemented comparably to F5 or F5C, it would compute Gröbner bases around 4 to 20 times faster.

4. Algorithm for Quotient Rings and Modules. Quotient Rings. Let $\mathbb{F}$ be any field and $R = \mathbb{F}[x_1, \ldots, x_n]$ be a polynomial ring. Let J be an ideal of R with Gröbner basis $G = \{f_1, \ldots, f_k\}$. Suppose I is an ideal of R/J generated by $\{g_1, \ldots, g_m\}$ where each g_i is already in normal form with respect to G. We wish to compute a Gröbner basis for $I = \langle g_1, \ldots, g_m\rangle$, and the $(g_1, \ldots, g_m)$-syzygy module.

We represent polynomials in R/J in normal form modulo G. This means that, for any $g \in R$, we have

$$g \equiv \sum_{i=1}^{\ell} c_i x^{\alpha_i} \pmod{G} \tag{4.1}$$

where no term x^{α_i} is divisible by any leading term of G. This expression can be

[2] We mention that while F5 and F5C require homogeneous input polynomials, G2V and GVW do not. In all the tables presented throughout, including table 3.1, G2V and GVW were also given the same homogeneous input polynomials as F5 and F5C. In retrospect, it may have been more fair to G2V and GVW to remove the homogenizing variable from the input polynomials.

Test Case (# gen)	POT/G2V	TOP	**g**1	**g**2
Katsura5 (22)	0.00	0.00	0.00	0.01
Katsura6 (41)	0.02	0.04	0.04	0.04
Katsura7 (74)	0.46	0.36	0.36	0.34
Katsura8 (143)	4.20	2.97	2.99	2.82
Schrans-Troost (128)	1.54	3.72	3.75	3.94
F633 (76)	0.07	0.43	0.36	0.06
Cyclic 6 (99)	0.04	0.66	0.64	0.07
Cyclic 7 (443)	5.40	253.75	252.02	7.49

Table 3.4

Runtimes in seconds using our C++ implementation on an Intel Core 2 Quad 2.66 GHz processor

Test Case (# gen)	POT/G2V	TOP	**g**1	**g**2
Katsura5 (22)	67	64	64	27
Katsura6 (41)	73	91	91	44
Katsura7 (74)	224	175	175	80
Katsura8 (143)	448	343	343	151
Schrans-Troost (128)	398	133	133	134
F633 (76)	135	184	170	106
Cyclic 6 (99)	155	1189	1189	188
Cyclic 7 (443)	749	9237	9237	846

Table 3.5

Sizes of Gröbner bases before any interreduction for different term orders

obtained from g by long division via G. When $g \in R$ is viewed as a polynomial in R/J, the leading term of g is the maximal x^{α_i} that appears in the normal form (4.1) of g. So the leading term of $g \in R/J$ is never divisible by any leading term of G.

We begin by defining a Gröbner basis for an ideal $I \subset R/J$. We say that a generating set $\{g_1, \ldots, g_m\} \subset R/J$ is a Gröbner basis for $I = \langle g_1, \ldots, g_m \rangle$ in R/J if for any $h \in I$, the leading monomial of h is divisible by the leading monomial of one of the generators, that is

$$\mathrm{lm}(g_i) \mid \mathrm{lm}\,(h) \quad \text{for some } 1 \leq i \leq m.$$

In other words, if $\{g_1, \ldots, g_m\}$ is a Gröbner basis for $I \subset R/J$ and $\{f_1, \ldots, f_k\}$ is a Gröbner basis for $J \subset R$, then $\{g_1, \ldots, g_m, f_1, \ldots, f_k\}$ is a Gröbner basis for $\langle g_1, \ldots, g_m, f_1, \ldots, f_k \rangle \subset R$.

The syzygy module for $\mathbf{g} = (g_1, \ldots, g_m) \in (R/J)^m$ is defined as

$$\mathbf{H} = \{(u_1, \ldots, u_m) \in (R/J)^m : u_1 g_1 + \cdots + u_m g_m = 0 \text{ in } R/J\}.$$

If viewed in the original ring R, every $(g_1, \ldots, g_m)$-syzygy in $(R/J)^m$ can be extended to an $(g_1, \ldots, g_m, f_1, \ldots, f_k)$-syzygy in R^{m+k}, which may vary depending on how $u_1 g_1 + \cdots + u_m g_m$ is reduced to 0 by G. In our computation, we only need to store the leading term of $(u_1, \ldots, u_m) \in \mathbf{H}$ where no terms in the u_i's are divisible by the leading terms of G.

Figure 4.1 describes a slight modification to the GVW algorithm that produces a Gröbner basis for $\langle g_1, \ldots, g_m \rangle \subset R/J$ and a Gröbner basis for the leading terms of the syzygy module $\mathbf{H}$, which can be used to calculate an actual Gröbner basis for $\mathbf{H}$.

Test Case (# gen)	POT/G2V	TOP	**g**1	**g**2
Katsura5 (22)	5.16	4.92	4.95	4.62
Katsura6 (41)	5.73	6.44	6.45	5.41
Katsura7 (74)	14.48	13.72	13.70	8.34
Katsura8 (143)	53.17	45.56	46.14	22.94
Schrans-Troost (128)	55.75	17.84	17.84	18.89
F633 (76)	7.91	10.09	8.89	6.05
Cyclic 6 (99)	5.88	26.55	26.86	6.06
Cyclic 7 (443)	43.36	2772.00	2764.00	42.06

Table 3.6
Maximal amount of memory used (MiB) for different term orders

This version of GVW can be used to compute Gröbner bases incrementally, each time adding m polynomials. For example, to compute a Gröbner basis for an ideal $I = \langle g_1, \dots, g_t\rangle \subset R$, one can first compute a Gröbner basis G for $J = \langle g_1, \dots, g_k\rangle \subset R$ where $k < t$. Then compute a Gröbner basis G_1 for $\langle g_{k+1}, \dots, g_t\rangle$ in the quotient ring R/J. Then $G \cup G_1$ is a Gröbner basis for I. And in the process, G is used in the reduction of many polynomials (e.g., the v part of every J-pair). By interpreting any polynomial in R/J as having already been reduced to normal form modulo G, we keep the number of terms in each polynomial to a minimum, thus reducing computational and storage requirements. Also, as the choice for k and m are arbitrary, one can design an algorithm that can compute Gröbner bases in one-shot, incrementally, or some hybrid of the two. This provides a flexible strategy for computing Gröbner bases for large systems of polynomials.

Calculating reduced Gröbner bases at intermediate steps is another advantage to running the algorithm in this mode. The biggest performance difference (after part (c) of theorem 2.3) between G2V and GVW under POT is that G2V is able to calculate a reduced Gröbner basis between each iteration. In fact, this is the advantage provided by F5C [7]. Our quotient ring version of GVW is able to do the same. For this reason, whenever an elimination order is used on R^m, this quotient ring version should improve performance.

Modules. Let $\mathbb{F}$ be a field and $R = \mathbb{F}[x_1, \dots, x_n]$ be a polynomial ring. Let $\mathbf{g}_1, \dots, \mathbf{g}_m$ be elements in R^s. We define an R-linear operator $T : R^m \to R^s$, uniquely determined by $\mathbf{g}_1, \dots, \mathbf{g}_m$, given by

$$(f_1, \dots, f_m) \longmapsto (f_1, \dots, f_m) \begin{bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \vdots \\ \mathbf{g}_m \end{bmatrix}.$$

We wish to determine the image space and kernel of T. Note that the image is the R-submodule $\mathbf{I}$ generated by $\{\mathbf{g}_1, \dots, \mathbf{g}_m\}$ in R^s while the kernel of T corresponds to the $(\mathbf{g}_1, \dots, \mathbf{g}_m)$-syzygy module $\mathbf{H}$ in R^m.

We fix term orders $\prec_1$ on R^s and $\prec_2$ on R^m, and let $\mathbf{u} = (f_1, \dots, f_m) \in R^m$ and $\mathbf{v} = T(\mathbf{u}) \in R^s$. We redefine M as an R-submodule of $R^m \times R^s$ so that

$$M = \{(\mathbf{u}, \mathbf{v}) \in R^m \times R^s : T(\mathbf{u}) = \mathbf{v}\}.$$

We continue to use $\mathbf{E}_i,\ 1 \le i \le m$ as the i^{th} unit vector in R^m, but to avoid confusion we use $\mathbf{F}_j,\ 1 \le j \le s$ as the j^{th} unit vector in R^s. And now, the R-module M is

```
            Algorithm for computing Gröbner bases in quotient rings
Input:      $G = \{f_1, \ldots, f_k\}$ a Gröbner basis for an ideal $J \subset R$, $g_1, \ldots, g_m$
            polynomials in R in normal form modulo $G$, and term orders for $R$
            and $R^m$.
Output:     A Gröbner basis for $\langle g_1, \ldots, g_m \rangle \in R/J$ and a Gröbner basis for
            $\mathrm{lm}(\mathbf{H})$, the leading terms of the syzygy module.
Variables:  $U$ a list of terms $T_i$, representing signatures of $(\mathbf{u}_i, v_i) \in M$.
            $V$ a list of polynomials $v_i$ for $(\mathbf{u}_i, v_i) \in M$,
            $H$ a list of $\mathrm{lm}(\mathbf{u})$ where $\mathbf{u} \in R^m$ is a syzygy found so far,
            $JP$ a list of pairs $(x^\alpha T_i, x^\alpha v_i)$, where $x^\alpha$ is a monomial so that
            $x^\alpha(\mathbf{u}_i, v_i)$ is a J-pair of $(\mathbf{u}_i, v_i)$ and $(\mathbf{u}_j, v_j)$ for some $j \neq i$.
Step 0.     $U = [\mathbf{0}, \ldots, \mathbf{0}]$ with length $k$, and $V = [f_1, \ldots, f_k]$,
            $JP = [(\mathbf{E}_1, g_1) \ldots, (\mathbf{E}_m, g_m)]$ and $H = [\ ]$, the empty list.
Step 1.     Take a minimal (in signature) pair $(T, v_1)$ from $JP$, and delete it
            from $JP$.
Step 2.     If $(T, v_1)$ satisfies theorem 2.3($c$) with $G = [U, V]$ (comparisons
            with respect to normal forms by $G$), then discard $(T, v_1)$ and go to
            step 5.
Step 3.     Reduce the pair $(T, v_1)$ repeatedly and as much as possible by the
            pairs in $[U, V]$ using only regular top-reductions, say to get $(T, v)$.
Step 4a.    If $v = 0$, then append $T$ to $H$, and delete every J-pair $(T_2, v_2)$ in
            $JP$ whose signature $T_2$ is divisible by $T$.
Step 4b.    If $v \neq 0$ and $(T, v)$ is not super top-reducible by $[U, V]$, then
              i) Add the leading terms of the principle syzygies, $vT_j - v_j T$ for
                 $1 \leq j \leq |U|$, to $H$,
             ii) form new J-pairs of $(T, v)$ and $(T_j, v_j)$, $1 \leq j \leq |U|$,
            iii) insert into $JP$ all such J-pairs whose signatures are not re-
                 ducible by $H$ (storing only one J-pair for each distinct signature
                 $T$, the one with $v$-part minimal), and
             iv) append $T$ to $U$ and $v$ to $V$,
Step 5.     While JP is not empty, go to step 1.
Return:     $V$ and $H$.
```

FIG. 4.1. *Algorithm for quotient rings*

generated by

$$(\mathbf{E}_1, \mathbf{g}_1), (\mathbf{E}_2, \mathbf{g}_2), \ldots, (\mathbf{E}_m, \mathbf{g}_m).$$

By now it should be clear that the GVW algorithm is a special case of this situation where $s = 1$ and is immediately applicable. The only differences that arise in this general case are in dealing with the leading monomials of the $\mathbf{v}$ part. Suppose $(\mathbf{u}_1, \mathbf{v}_1)$ and $(\mathbf{u}_2, \mathbf{v}_2)$ are two pairs in $R^m \times R^s$, with $x^\alpha \mathbf{F}_j = \mathrm{lm}(\mathbf{v}_1)$ and $x^\beta \mathbf{F}_k = \mathrm{lm}(\mathbf{v}_2)$. We consider $(\mathbf{u}_2, \mathbf{v}_2)$ as a candidate to top-reduce $(\mathbf{u}_1, \mathbf{v}_1)$ only if $j = k$. Also, we only calculate the J-pair between $(\mathbf{u}_1, \mathbf{v}_1)$ and $(\mathbf{u}_2, \mathbf{v}_2)$ if $j = k$. In this case, assuming $\mathbf{v}_1, \mathbf{v}_2 \neq 0$, we have

$$t = \mathrm{lcm}(x^\alpha, x^\beta), \quad t_1 = \frac{t}{x^\alpha}, \quad t_2 = \frac{t}{x^\beta},$$

and if $t_i \mathbf{u}_i = \max\{t_1 \mathbf{u}_1, t_2 \mathbf{u}_2\}$, then $t_i(\mathbf{u}_i, \mathbf{v}_i)$ is a J-pair. Everything else proceeds as before.

5. Conclusions and related recent works. We have presented a new algorithm for computing Gröbner bases for ideals and modules (including syzygy modules). In terms of simplicity, our algorithm is as simple as Buchberger's algorithm in the sense that we form J-pairs and perform reductions. We can detect useless J-pairs using syzygies and eventual super top-reducibility. Our algorithm is more flexible than F5 and our previous algorithm G2V [12] in that we allow a Gröbner basis to be computed incrementally, in one-shot, or a hybrid of the two. It is in this flexibility that we achieve an efficiency boost over G2V as some signature orderings perform better than others. Indeed, the **g**2 ordering performs better than others and is suggested for general implementation of Gröbner basis algorithms. Also, we believe that F4 style fast reductions are possible within the context of our algorithm, but the question remains as to how to implement it efficiently.

After our paper was submitted in October 2010, several related papers have appeared. Eder and Perry [8] provide a detailed comparison on F5, G2V and Arri's algorithm. As mentioned earlier, Huang [14] have completely characterized when our algorithm has finite termination, and his proof works also for our algorithm for quotient rings and general modules. Sun and Wang [19] generalized the GVW algorithm further and they allow J-pairs be processed in any order, not just in increasing signature orders, which will provide more flexibility in implementation. Huang [14] characterize Gröbner bases in terms of TRB and TRP pairs, while Arri and Perry [1] characterize Gröbner bases in terms of S-irreducible polynomials and S-primitive polynomials. TRB pairs are equivalent to S-irreducible polynomials, and TRP pairs are equivalent to S-primitive polynomials. We note that, in our language, that a pair $(\mathbf{u}, v)$ is a TRB pair if it is not regular top-reducible by any pair in the module M in (2.3), and a TRB pair is a TRP pair if it is not super top-reducible by another TRB pair whose signature is strictly smaller. To be able to check such a property for $(\mathbf{u}, v)$ by a current G, G must contain all TRP pairs whose signatures are smaller than $\text{lm}(\mathbf{u})$. The condition (c) of Theorem 2.3 is stated in Huang [14] as M-pair criterion in terms of TRP pairs and in Arri and Perry [1] as F5 Criterion in terms of S-primitive polynomials. Their algorithms must process J-pairs or S-polynomials in increasing signature order. If J-pairs are processed in increasing order, their algorithms are very similar to ours. We believe that our approach is much simpler, and as remarked after the proof of Theorem 3.1, our algorithm can be easily modified so that J-pairs can be processed in any order, not necessarily in increasing order.

REFERENCES

[1] A. Arri and J. Perry, *The F5 criterion revised*, CoRR, arXiv:1012.3664v3 (2010).

[2] B. Buchberger, *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal*, PhD thesis, Leopold-Franzens University, 1965.

[3] B. Buchberger, *A criterion for detecting unnecessary reductions in the construction of Gröbner bases*, in EUROSAM '79: Proceedings of the International Symposiumon on Symbolic and Algebraic Computation, London, UK, 1979, Springer-Verlag, pp. 3–21.

[4] B. Buchberger, *Gröbner-Bases: An Algorithmic Method in Polynomial Ideal Theory.*, Reidel Publishing Company, Dodrecht - Boston - Lancaster, 1985.

[5] N. Courtois, E. Klimov, J. Patarin, and A. Shamir, *Efficient algorithms for solving overdefined systems of multivariate polynomial equations*, in In Advances in Cryptology, Eurocrypt'2000, LNCS 1807, Springer-Verlag, 2000, pp. 392–407.

[6] J. Ding, J. Buchmann, M. S. E. Mohamed, W. S. A. E. Mohamed, and R.-P. Weinmann, *MutantXL*, in First International Conference on Symbolic Computation and Cryptography, Springer-Verlag, 2008.

[7] C. Eder and J. Perry, *F5C: A variant of Faugère's F5 algorithm with reduced Gröbner bases*, Journal of Symbolic Computation, 45 (2010), pp. 1442 – 1458. MEGA'2009.

[8] C. Eder and J. Perry, *Signature-based algorithms to compute Gröbner bases*, CoRR, arXiv:1101.3589 (2011).

[9] J. C. Faugère, *A new efficient algorithm for computing Gröbner bases (F4)*, Journal of Pure and Applied Algebra, 139 (1999), pp. 61 – 88.

[10] J. C. Faugère, *A new efficient algorithm for computing Gröbner bases without reduction to zero (F5)*, in ISSAC '02: Proceedings of the 2002 international symposium on Symbolic and algebraic computation, New York, NY, USA, 2002, ACM, pp. 75–83.

[11] J. C. Faugère and A. Joux, *Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using gröbner bases*, in In Advances in Cryptology — CRYPTO 2003, Springer, 2003, pp. 44–60.

[12] S. Gao, Y. Guan, and F. Volny IV, *A new incremental algorithm for computing Gröbner bases*, in ISSAC'10: Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation, Munich, Germany, 2010, ACM, pp. 13–19.

[13] A. Hashemi and G. Ars, *Extended F5 criteria*, Journal of Symbolic Computation, 45 (2010), pp. 1330 – 1340. MEGA'2009.

[14] L. Huang, *A new conception for computing Gröbner basis and its applications*, CoRR, arXiv:1012.5425v2 (2010).

[15] D. Lazard, *Gröbner-bases, Gaussian elimination and resolution of systems of algebraic equations*, in EUROCAL '83: Proceedings of the European Computer Algebra Conference on Computer Algebra, London, UK, 1983, Springer-Verlag, pp. 146–156.

[16] H. M. Möller, T. Mora, and C. Traverso, *Gröbner bases computation using syzygies*, in ISSAC '92: Papers from the international symposium on Symbolic and algebraic computation, New York, NY, USA, 1992, ACM, pp. 320–328.

[17] J. Patarin, *Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms*, in EUROCRYPT'96: Proceedings of the 15th annual international conference on Theory and application of cryptographic techniques, Berlin, Heidelberg, 1996, Springer-Verlag, pp. 33–48.

[18] Y. Sun and D. Wang, *A new proof of the F5 algorithm*, CoRR, arXiv:1004.0084 (2010).

[19] Y. Sun and D. Wang, *A generalized criterion for signature related Gröbner basis algorithms*, CoRR, arXiv:1101.3382 (2011).