A NEW ALGORITHM FOR COMPUTING GRÖBNER BASES*

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

Abstract. The paper presents a simple characterization for so-called strong Gröbner bases which contains Gröbner bases for both ideals and the corresponding syzygy modules (for the given generators of the ideals). This characterization can detect useless S-polynomials without reductions, thus yields an efficient algorithm for computing strong Gröbner bases. Rigorous proofs are given for the correctness and finite termination of this algorithm. For any term order for an ideal, one may vary signature orders (i.e. the term orders for the syzygy module). It is shown by computer experiments on benchmark examples that signature orders based on weighted degree are much better than other signature orders. This is useful for practical computation. Also, since computing Gröbner bases for syzygies is a main computational task for free resolutions in commutative algebra, the algorithm of this paper should be useful for computing free resolutions in practice.

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

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 including 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, [17]) go a step further to present an algorithm using the full module of syzygies, however, their algorithm is not very efficient. Faugère (2002, [9]) introduced the idea of signatures and rewriting rules that can detect many useless S-polynomials, hence saving a significant amount of time. 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, [10]) solved the first Hidden Field Equation (HFE) Cryptosystem Challenge which involves a system of 80 polynomial equations with 80 variables over the binary field (1996, [18]).

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,

* The work presented in this paper was partially the 973 Project (No. 2013CB834203), the National Science Foundation of China under Grant 11171323, and the National Science Foundation of USA under grants DMS-1005369 and CCF-0830481.

†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 Information Engineering, Chinese Academy of Sciences, Beijing 100190, P. R. China mingsheng_wang@yahoo.com.cn

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, [8]), which deals with much smaller matrices. 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 F5 for example), even for medium systems of polynomials.

F5 as presented in [9] is difficult to understand, the proofs of its correctness and finite termination contain significant gaps. Stegers (2006 [20] filled some details of the proofs under the assumption of two conjectures, but one of which was later shown to be false by Gash (2008 [12]). More recently, Arri and Perry (2011 [1]) presented a simpler theory for signature based algorithms. They gave a revised F5 criterion with correct proof, however, their proof of finite termination is flawed (see details in Section 3).

The main contribution of the current paper is to present a new simple theory for computing Gröbner bases. Every list of polynomials defines an ideal and a syzygy module. Most papers in the literature focus on computing Gröbner bases for ideals, while Gröbner bases for syzygies are computed by a totally different method. We show that the two types of Gröbner bases can be computed by a unified framework. We work in a bigger module that contains both the ideal and the syzygy module for a given list of polynomials, and define signatures, J-pairs, and reductions in a natural fashion. A strong Gröbner basis for the big module contains a Gröbner basis for the ideal as well as a Gröbner basis for the syzygy module. Our main result is a simple characterization of strong Gröbner bases (see Theorem 2.4). This characterization has the desirable features that useless J-pairs can be detected without performing any reduction and that J-pairs can be processed in any order. Note that computing Gröbner bases for syzygies is a main computational task for free resolutions in commutative algebra. Our work should be useful for computing free resolutions in practice.

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, J-pairs, and strong Gröbner bases. The main result is Theorem 2.4. As a special case, this provides a proof for the correctness of the G2V algorithm in [11], which was missing there. In Section 3, we present our algorithm and give a simple proof for its finite termination. We also present computer experiments on some benchmark examples and compare the times of our algorithm under different signature orders. In the last section, we discuss how our theory is related to other related works and mention some recent progress since the current paper was initially submitted (in 2010).

2. Theory. Let $R = \mathbb{F}[x_1, \ldots, x_n]$ be a polynomial ring over a field $\mathbb{F}$ with n variables. For any polynomials $g_1, \ldots, g_m \in R$, we define an ideal of R:

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

and a submodule of R^m:

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

which is 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 $\mathbf{H}$ under any given term orders on R and R^m.

To establish the theoretical foundation for our algorithm, we work in the larger R-module $R^m \times R$ which allows us to handle the ideal I and the syzygy module $\mathbf{H}$ simultaneously. 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 subset of $R^m \times R$:

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

It is an R-submodule of $R^m \times R$ because it is closed under addition and multiplication by R, that is, for any $(\mathbf{u}_1, v_1), (\mathbf{u}_2, v_2) \in M$ and any $r_1, r_2 \in R$, we have

$$r_1(\mathbf{u}_1, v_1) + r_2(\mathbf{u}_2, v_2) = (r_1\mathbf{u}_1 + r_2\mathbf{u}_2, r_1v_1 + r_2v_2) \in M.$$

For $1 \leq i \leq m$, let $\mathbf{E}_i \in R^m$ be the i^{th} unit vector whose i-th entry is 1 and other entries are 0. 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

$$(x^\beta \mathbf{E}_i)/(x^\alpha \mathbf{E}_i) = x^{\beta-\alpha} \in R.$$

Also, the R-module M in (2.3) 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 *compatible* with $\prec_1$, that is,

$$x^\alpha \prec_1 x^\beta \quad \text{iff} \quad x^\alpha \mathbf{E}_i \prec_2 x^\beta \mathbf{E}_i \text{ for all } 1 \leq i \leq m.$$

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 $(\mathbf{u}, v)$. Our definition of signatures is different from that of F5 [1, 9] where each $v \in I = \langle g_1, \ldots, g_m \rangle$ is associated with a signature:

$$S(v) = \min\{\mathrm{lm}(\mathbf{u}) : \mathbf{u} \in R^m \text{ with } \mathbf{u}\mathbf{g}^t = v\}.$$

The F5 signature is hard to use in practice, while our signature is natural and easy to use.

We 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 decreased 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 $p_1 - ctp_2$ remains the same as p_1 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 $v_2 = 0$, $p_2 = (\mathbf{u}_2, 0) \in M$ corresponds to the syzygy $\mathbf{u}_2$. We say that p_1 is *top-reducible* by a syszygy $p_2 = (\mathbf{u}_2, 0)$ if $\mathbf{u}_1$ and $\mathbf{u}_2$ are both nonzero and $\mathrm{lm}(\mathbf{u}_2)$ divides $\mathrm{lm}(\mathbf{u}_1)$. A top-reduction by a syzygy is always called *super*. Hence, if $p_1 = (\mathbf{u}_1, v_1)$ is super top-reducible by $p_2 = (\mathbf{u}_2, v_2)$ in either case, then $\mathrm{lm}(\mathbf{u}_2)$ divides $\mathrm{lm}(\mathbf{u}_1)$. We note that a pair $(\mathbf{u}_1, 0)$ is never top-reducible by $(\mathbf{u}_2, v_2)$ with $v_2 \neq 0$, and in our algorithm below, we only detect super top-reductions of the two kinds defined here, but never actually perform super top-reductions.

Definition 2.1. *A subset G of M is called a strong Gröbner basis for M if every nonzero pair in M is top-reducible by some pair in G.*

This definition is similar to the usual definition for Gröbner bases for ideals: a subset G of an ideal I is called a Gröbner basis if the leading term of every polynomial in I is divisible by the leading term of some polynomial in G, that is, every polynomial in I is top-reducible by some polynomial 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. Then*

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

Proof. For any $\mathbf{u} = (u_1, \ldots, u_m)$ 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 $\mathrm{lm}(\mathbf{u})$ is reducible by $\mathrm{lm}(\mathbf{u}_i)$. 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 $\mathbf{u} = (u_1, \ldots, u_m) \in R^m$ so that $\mathbf{u}\mathbf{g}^t = v$, hence $(\mathbf{u}, v) \in M$. Among all such $\mathbf{u}$, we pick one so that $\mathrm{lm}(\mathbf{u})$ is minimum. Since $(\mathbf{u}, v) \in M$, it is top-reducible by some $(\mathbf{u}_i, v_i)$ where $1 \leq i \leq k$. 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 $\mathrm{lm}(\mathbf{u}')$ is smaller than $\mathrm{lm}(\mathbf{u})$, contradicting to the minimality of $\mathrm{lm}(\mathbf{u})$. So $v_i \neq 0$ and $\mathrm{lm}(v_i)$ divides $\mathrm{lm}(v)$. Hence G_1 is a Gröbner basis for I. □

Remark. Note that $M \subset R^m \times R$ has a Gröbner basis in the classical sense of Burchberger as a submodule of R^{m+1} where the leading term of $(\mathbf{u}, v)$ is $\mathrm{lm}(v)\mathbf{E}_{m+1}$ if $v \neq 0$ and $\mathrm{lm}(\mathbf{u})$ if $v = 0$. The above proposition implies that a strong Gröbner basis for M is a classical 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.

Since M is infinite, it is not clear how to check whether a given set of generators for M is a Gröbner basis. We need a characterization in term of G itself similar to Buchberger's criterion. We define 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 $T = \max(t_1\mathrm{lm}(\mathbf{u}_1), t_2\mathrm{lm}(\mathbf{u}_2))$. Without loss of generality, we assume $T = t_1\mathrm{lm}(\mathbf{u}_1)$. If

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

then T is called the *J-signature* of p_1 and p_2, while t_1p_1 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_1p_1 by p_2. 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 any two pairs 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. This seems strange at first glance, but it is useful in saving storage as a J-pair tp_i can be stored simply as a pair (t, i) where i is the index of the pair $p_i = (\mathbf{u}_i, v_i)$, instead of storing the actual pair $(t\mathbf{u}_i, tv_i)$. 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 2.3. *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 (hence 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, we know that both v_1 and v_2 are nonzero and 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,

$$t = \frac{\mathrm{lm}(v_2)}{\gcd(\mathrm{lm}(v_1), \mathrm{lm}(v_2))} w = t_1 w, \text{ and}$$
$$s = \frac{\mathrm{lm}(v_1)}{\gcd(\mathrm{lm}(v_1), \mathrm{lm}(v_2))} w = t_2 w.$$

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 $t_1 p_1$ is the J-pair of p_1 and p_2, and $t_1 p_1$ is regular top-reducible by p_2. □

Let G 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 G if it is regular top-reducible by at least one pair in G. We call $(\mathbf{u}, v)$ *eventually super top-reducible* by G if there is a sequence of regular top-reductions of $(\mathbf{u}, v)$ by pairs in G that reduce $(\mathbf{u}, v)$ to a pair $(\mathbf{u}', v')$ that is no longer regular top-reducible by G but is super top-reducible by at least one pair in G. Also, we say that a pair $(\mathbf{u}, v)$ is *covered* by G if there is a pair $(\mathbf{u}_i, v_i) \in G$ so that $\mathrm{lm}(\mathbf{u}_i)$ divides $\mathrm{lm}(\mathbf{u})$ and $t\,\mathrm{lm}(v_i) \prec \mathrm{lm}(v)$ (strictly smaller) where $t = \mathrm{lm}(\mathbf{u})/\mathrm{lm}(\mathbf{u}_i)$. Note that there is no reduction at all in checking whether a pair is covered by G.

THEOREM 2.4. *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 $T = t\,lm(\mathbf{u})$. 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*) *every J-pair of G is covered by G.*

Proof. (*a*) ⇒ (*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*) ⇒ (*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 $\mathrm{lm}(\mathbf{u}_1) \mid \mathrm{lm}(\mathbf{u}_0) = \mathrm{lm}(\mathbf{u})$ and $tv_1 = 0$ is smaller than $\mathrm{lm}(v)$. So we may assume that $v_1 \neq 0$. Then

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

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

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

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

(*c*) ⇒ (*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 $T \neq 0$. 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 2.3, 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. By the assumption in (c), the J-pair $t_1 p_1$ is covered by G, hence 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). □

The condition (c) of Theorem 2.4 tells us that any J-pair that is covered by G can be discarded (without performing any reductions). This will greatly speed up the algorithm. As special cases, we have the following two criteria.

COROLLARY 2.5 (Syzygy Criterion). *If a J-pair is top-reducible by a syzygy, then it can be discarded.*

COROLLARY 2.6 (Signature Criterion). *Among all J-pairs with an equal signature, one just needs to store one of them (the one with the v-part minimal).*

Remarks. In the original version of this paper (presented in ISSAC 2010 July 25–29, Münich, Germany), Theorem 2.4 had only (a) and (b). Later, Huang (November 2010 [14]) and Arri and Perry (December 2010, [1]) characterize Gröbner bases in term of irreducible and primitive irreducible pairs (or polynomials). In particular, Arri and Perry [1] gave an F5 criterion similar to our condition (b) (see more comments about this in Section 5), but they used the condition (c) (without proof) in their algorithm. When using the condition (b) or the F5 criterion of [1], the J-pairs must be processed in increasing order. Our condition (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.

3. Algorithm and Finite Termination. Our algorithm is based on Theorem 2.4. 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 any J-pair from the list of J-pairs (usually the one with minimal signature). Check if the minimality 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 vectors $\mathbf{u} \in R^m$ are expensive. In our computation, we shall make all pairs $(\mathbf{u}, v)$ monic, namely, the leading coefficient of $\mathbf{u}$ is 1. Then we only store the signature, i.e., the leading term of $\mathbf{u}$. 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 $\mathrm{lm}(v_2)$, and $tT_2 \prec T_1$, 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 of $(\mathbf{u}, v)$ monic 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. There are $\mathbf{u}_i \in R^m$ such that $\mathrm{lm}(\mathbf{u}_i) = T_i$ and $(\mathbf{u}_i, v_i) \in M$ for $1 \leq i \leq 2$. 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 $v_2\mathbf{u}_1$ and $v_1\mathbf{u}_2$ 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 leading terms of these syzygies are obtained free (i.e., without performing any reductions), 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. *Suppose the term order in R is compatible with the term order in R^m. Then the algorithm in Figure 3.1 terminates in finitely many steps with a strong Gröbner basis for M.*

Proof. The correctness of the algorithm follows directly from Theorem 2.4, as Step 2 makes sure the condition (c) is satisfied. We only need to prove the finite termination of the algorithm. For any two pairs $p_1 = (\mathbf{u}_1, v_1), p_2 = (\mathbf{u}_2, v_2) \in M$, we say that p_1 divides p_2 if $\mathrm{lm}(\mathbf{u}_1) \mid \mathrm{lm}(\mathbf{u}_2)$ and $\mathrm{lm}(v_1) \mid \mathrm{lm}(v_2)$. We list the pairs in G in exactly the same order as they were obtained (not including $(T, 0)$ for $T \in H$):

$$(E_1, g_1), (E_2, g_2), \ldots, (E_m, g_m), (T_1, v_1), (T_2, v_2), \ldots, (T_i, v_i), \ldots$$

Then there exist $\mathbf{u}_i \in R^m$ so that $\mathrm{lm}(\mathbf{u}_i) = T_i$ for $i \geq 1$. Let $p_i = (\mathbf{u}_i, v_i)$ for $i \geq 1$. We claim that, for all $i < j$, p_i does not divide p_j. Suppose otherwise, say $p_i = (\mathbf{u}_i, v_i)$ divides $p_j = (\mathbf{u}_j, v_j)$ for some $i < j$. Then there are monomials $t_1, t_2 \in R$ so that

$$\mathrm{lm}(v_j) = t_1\mathrm{lm}(v_i), \quad \mathrm{lm}(\mathbf{u}_j) = t_2\mathrm{lm}(\mathbf{u}_i).$$

Suppose $t_1 \prec t_2$ (in R). Then $t_1\mathrm{lm}(\mathbf{u}_i) \prec t_2\mathrm{lm}(\mathbf{u}_i)$ (since the term orders are compatible). As $t_2\mathrm{lm}(\mathbf{u}_i) = \mathrm{lm}(\mathbf{u}_j)$, we have that p_j is regular top-reducible by p_i, contradicting to the choice of the algorithm. Thus we must have $t_2 \preceq t_1$. Then $t_2\mathrm{lm}(v_i) \preceq t_1\mathrm{lm}(v_i) = \mathrm{lm}(v_j)$. Let $p = (\mathbf{u}, v)$ be the J-pair that was reduced to p_j by the algorithm. Then $\mathrm{lm}(\mathbf{u}) = \mathrm{lm}(\mathbf{u}_j) = T_j$ and $\mathrm{lm}(v_j) \prec \mathrm{lm}(v)$ (as a J-pair is always regular top-reducible). Hence the J-pair p is covered by p_i, hence should have been discarded by the algorithm. Therefore we have a sequence

$$(T_1, \mathrm{lm}(v_1)), (T_2, \mathrm{lm}(v_2)), \ldots, (T_i, \mathrm{lm}(v_i)), \ldots \tag{3.1}$$

with no pair divisible by any previous one.

We introduce new variables

$$y_i = (y_{i1}, y_{i2}, \ldots, y_{in}), \quad 1 \leq i \leq m.$$

Each pair $(x^\alpha E_i, x^\beta)$ corresponds to a term $y_i^\alpha x^\beta$, a monomial in the variables x_i's and y_{ij}'s. Then the pairs in (3.1) gives us a list of monomials in x_i's and y_{ij}'s with the property that no one divisible by any previous one. Since every polynomial ring over a field is Noetherian, the ascending chain condition tells us that this list of monomials must be finite. Therefore, G is finite. □

Remarks on Finite Termination. We would like to make a few remarks about proofs of finite termination that have appeared in the literature.

	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^α, i), where x^α 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.	$U = [\mathbf{E}_1, \ldots, \mathbf{E}_m]$, and $V = [g_1, \ldots, g_m]$. Find the leading terms of the principle syzygies $g_j\mathbf{E}_i - g_i\mathbf{E}_j$ for $1 \leq i < j \leq m$, and add them in H. Compute all the J-pairs of $(\mathbf{E}_1, g_1), \ldots, (\mathbf{E}_m, g_m)$ storing into JP: storing only the J-pairs whose signatures are not reducible by H and storing only one J-pair for each distinct signature.
Step 1.	Take any pair (x^α, i) from JP (say with minimal signature), and delete it from JP. Let $(T, v) = x^\alpha(\mathbf{u}_i, v_i)$.
Step 2.	If (T, v) is covered by $G = [U, V]$, then discard (T, v) and go to Step 5.
Step 3.	Reduce the pair (T, v) repeatedly by G using only regular top-reductions until it is not regular top-reducible, say to get $(T, \tilde{v})$.
Step 4a.	If $\tilde{v} = 0$, then append T to H, and delete every J-pair in JP whose signature is divisible by T.
Step 4b.	If $\tilde{v} \neq 0$ then (b1) Add the leading terms of the principle syzygies, $\tilde{v}T_j - v_jT$ for $1 \leq j \leq \lvert U \rvert$, to H (and delete any redundant ones), (b2) Form new J-pairs between $(T, \tilde{v})$ and (T_j, v_j), $1 \leq j \leq \lvert U \rvert$, and insert into JP all such J-pairs whose signatures are not reducible by H (storing only one J-pair for each distinct signature T, the one with v-part minimal), and (b3) Append $(T, \tilde{v})$ to G (i.e. 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.

(a) We remark that Huang (2010 [14]) is the first person who gave a correct proof of finite termination for signature-based algorithms when the J-pairs are processed in increasing order. The proof for the general case (when J-pairs are processed in arbitrary order) is partly due to Sun and Wang (2012, [16], especially the part for $t_2 \preceq t_1$.

(b) Huang gave a counter example when the orders for R and R^m are not compatible. For convenience of the reader, we reproduce his example here. Let $g_1 = x_2, g_2 = x_1 - x_2 \in R = \mathbb{F}[x_1, x_2]$. Suppose that the term order in R is the lex order with $x_2 \prec_1 x_1$ and the term order for R^2 is defined by position and then the reverse lex order with $E_2 \prec_2 E_1$ and $x_1E_i \prec_2 x_2E_i$ for $i = 1, 2$. So the two orders are not compatible. Starting with the pairs (E_1, x_2) and $(E_2, x_1 - x_2)$, every signature-based algorithm will produce the

infinite sequence:

$$(x_1^k E_1 - (x_1^{k-1}x_2 + x_1^{k-2}x_2^2 + \cdots + x_2^k)E_2, \quad x_2^k), \quad k = 2, 3, \cdots$$

in which no pair is top-reducible by any other pair (including (E_1, x_2) and $(E_2, x_1 - x_2)$). Hence every signature-based algorithm will not have finite termination for these term orders.

(c) 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.

(d) 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 example. Even if 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 $R^m \times R$ is generated by a set of elements of the form

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

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

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

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

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

Then $(E_1, y) = (E_1, x) - (E_2, x) + (E_2, y) \in N$, but (E_1, y) 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 method. 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

$$(\mathbf{u}_1, v_1), (\mathbf{u}_2, v_2), \ldots, (\mathbf{u}_m, v_m), \ldots, (\mathbf{u}_\ell, v_\ell) \tag{3.2}$$

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 (3.2) 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.

4. Term Orderings and Time Comparison. Explicit 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 $i < j$.

(**g**1) Next is the **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.

(**g**2) Finally, we have **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 Theorem 3.1. We remark that, under the POT order, our new algorithm closely corresponds to the G2V algorithm presented in [11]. 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 $\langle g_1\rangle$, $\langle g_1, g_2\rangle$, $\ldots$, $\langle g_1, g_2, \ldots, g_m\rangle$, 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 **g**1 order roughly corresponds to the behavior of the F4 and XL algorithms [5]. In the XL algorithm, one performs row

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 4.1

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 [11].

Test Case	F5	F5C	G2V	POT	TOP	$\mathbf{g}1$	$\mathbf{g}2$
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 4.2

Counts of the J-pairs or S-polynomials processed by F5, F5C (as in [7]), G2V (as in [11]), and GVW under POT, TOP, $\mathbf{g}1$ *and* $\mathbf{g}2$ *orders*

reductions on a matrix whose rows correspond to all polynomials $x^\alpha g_i$, $1 \leq i \leq m$, 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++ . Because our C++ implementation is vastly different than our F5/C and G2V implementations, we did not compare timings as we did in [11] (see Table 4.1, reproduced here[1] for comparison purposes). Instead, table 4.2 lists the counts of J-pairs or S-polynomials processed by each algorithm. Within Table 4.2, we distinguish between the G2V (as in [11], without theorem 2.4(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 2.4(c).

Just as in [11], various benchmark examples (from [7]) were run for comparison. We collected data from each example under each term ordering for comparison. Table 4.3 list the runtimes in seconds of GVW for each of the four term orderings. In examining the timings, we find that $\mathbf{g}2$ seems to be a clear winner among the four term orders.

Table 4.4 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 interre-

[1] with permission from ACM.

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 4.3
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 4.4
Sizes of Gröbner bases before any interreduction for different term orders

duction 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 $\langle g_1, \ldots, g_m \rangle$.

5. Related Works and Conclusions. In this section, we make some detailed comments about how our work is related to other recent works in the literature.

Since F5 algorithm was published in 2002, several papers have been published trying to simply F5 and fills in details in the proofs of its correctness and finite termination including Stegers (2006 [20], Gash (2008, [12]), Eder and Perry (2009,[7]), Sun and Wang (2009 [21]), Hashemi and Ars (2010 ,[13]), and Arri and Perry (2011, [1]).

Gao, Guan and Volny (2010, [11]) give a new incremental approach which is the origin of the current paper. In [11], an algorithm is presented, however, no proof of correctness nor finite termination is given. After the original version of this paper was submitted in October 2010, several related papers have appeared. Eder and Perry (2011, [1]) provide a detailed comparison on F5, G2V and Arri's algorithm. As mentioned earlier, Huang (2010, [14]) completely characterizes when our algorithm has finite termination (when J-pairs are processed in increasing order) and give a counter example when the term orders are not compatible. Sun and Wang (2011–2012 [22, 16]) 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.

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 4.5
Maximal amount of memory used (MiB) for different term orders

Huang [14] characterizes Gröbner bases in terms of TRB and TRP pairs, while Arri and Perry (2011, [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 whether $(\mathbf{u}, v)$ is S-irreducible by using current G, G must contain all TRP pairs whose signatures are smaller than $\mathrm{lm}(\mathbf{u})$. Roune and Stillman (2012, [19]) present implementation details of signature based algorithms.

In the following, we give more technical details of comparison of our work to Faugére's F5 [9] and Arri and Perry's F5 criterion [1], which are most relevant to our work. In our paper, we never define any signature of a polynomial $v \in I$, instead we define the signature of a pair $(\mathbf{u}, v) \in M$. This seems more natural and easier to use. In comparison, in both papers [1, 9], they define signatures as follows. For any $v \in I = \langle g_1, \ldots, g_m \rangle$, the signature of v is defined as

$$S(v) = \min\{\mathrm{lm}(\mathbf{u}) : \mathbf{u} \in R^m \text{ with } \mathbf{u}\mathbf{g}^t = v\}.$$

Their definition is not exactly as above but is equivalent to it. In F5, the term order in R^m is the position-over-term order (POT), hence the algorithm is incremental, that is, it computes the Gröbner basis for each of the ideals $\langle g_i, \ldots, g_m \rangle$ for $i = m, m-1, \ldots, 1$. In [1], the term order in R^m can be arbitrary. In their algorithms, to make sure that each polynomial added to the current Gröbner basis is represented by a minimal $\mathbf{u} \in R^m$, one has to process the critical pairs (or J-pairs in our language) in increasing order. In F5, the generator polynomials $g_1, \ldots, g_m$ are assumed to be homogeneous and the critical pairs are processed from minimal degree to higher degrees, this is more or less equivalent to increasing signature order. However, if the generator polynomials are not homogeneous, their signatures may not be in increasing order any more. So F5 as presented in [9] does not work for nonhomogeneous polynomials. The algorithm in [1] does process critical pairs in increasing order and claim to have $S(v)$ for each $v \in G$. The latter, however, is not rigorously proved. In fact, their algorithm stores a list L of leading terms of syzygies (which come from critical pairs that are reduced to 0, but they forgot to include trivial syzygies). For any pair (T, v), the authors claim that $T = S(v)$ iff T is not divisible by any leading term in L. This is true only if L generates all the leading terms of syzygies up to T which is, however, not justified in their paper.

To simply F5 algorithm and to adapt F5 to general polynomials, Arri and Perry [1] introduce a revised F5 criterion (Proposition 18). In fact, their F5 criterion corresponds to our condition (b). To see this, we recall the main condition of their F5 criterion:

for any $g_1, g_2 \in G$ such that (g_1, g_2) is a normal pair, there exists $g \in G$ and a monomial t such that tg is S-irreducible and

$$S(tg) = S(Spol(g_1, g_2)),$$

where G is a subset of S-irreducible polynomials in I. Let

$$S(g) = T_1 \text{ and } S(\text{Spol}(g_1, g_2)) = T_2.$$

On the one hand, the condition that tg is S-irreducible implies that $S(tg) = tT_1$. On the other hand, the condition that tg is S-irreducible means, in our language, that (T_1, g) is not regular top-reducible by any pair in M. Let v be the polynomial obtained from $\text{Spol}(g_1, g_2)$ via regular top-reductions by M so that (T_2, v) is S-irreducible. Now $S(tg) = S(\text{Spol}(g_1, g_2))$ means that $tT_1 = T_2$. Hence both (tT_1, tg) and (T_2, v) are S-irreducible with the same signature. Then we must have $\text{lm}(tg) = \text{lm}(v)$, thus (T_2, v) is super top-reducible by (T_1, g). This means that the S-pair $(T_2, \text{Spol}(g_1, g_2))$ is eventually super top-reducible by G. Also, the condition that tg is S-irreducible (by M) is hard to check in practice, but it is not required by our condition (b). Therefore, except the requirement of being a normal pair, the F5 criterion is equivalent to our condition (b).

In [9], rewritten rules are introduced to eliminate many critical pairs in F5. In our language, the F5 rewritten rules can be summarized as follows (as described in [1]):

(S) if the signature of a critical pair is divisible by some term in H, then this pair can be discarded, where H is the collection of leading terms of all trivial syzygies and the signatures of critical pairs that are reduced to 0 known so far;

(R) a pair $(t\mathbf{E}_i, v)$ (from a critical pair) is discarded if

(R1) there is a pair $(t_1\mathbf{E}_i, v_1) \in G$ or in JP so that t_1 divides t and v_1 was computed before v.

In [1] (Algorithm 21), they use the rule (S), except that they forgot to include the leading terms of trivial syzygies in H (or L in their notation), and use the rule (R) with (R1) replaced by

(R2) there is a pair $(t_1\mathbf{E}_i, v_1) \in G$ or in JP so that t_1 divides t and $t_2\text{lm}(v_1) \prec \text{lm}(v)$ where $t_2 = t/t_1$.

The mathematical implication of the rule (R1) is not clear, but for homogeneous polynomials, one may interpret (R1) as (R2).

In fact, (R2) is similar to our condition (c) in Theorem 2.4. The condition (R2) is implied by the F5 criterion (Proposition 18 [1]) or by our condition (b). However, Arri and Perry did not prove that (R2) is sufficient to get a Gröbner basis. Using F5 criterion or our condition (b), one has to process J-pairs in increasing order, while the condition (c) has no such constraint at all, one can process J-pairs in any order, which may be useful in practical implementation.

In conclusion, we have presented simple characterizations of strong Gröbner bases that encode Gröbner bases for both ideals and syzygy modules. Computing syzygies has traditionally been approached separately by different methods, however, our paper shows that it can be handled simultaneously with computing of Gröbner bases

for ideals and they each help speed up the computation of the other. Our characterization (b) is natural generalization of Burchberger's criterion for ideals, but (c) is totally different and it is more computing friendly as it detects useless J-pairs without any reduction. We presented a complete proof of correctness and finite termination and showed via benchmark examples that different signature orders may have dramatic impact on the time for computing Gröbner bases for ideals. We hope that the simplicity of our characterizations of strong Gröbner bases is useful for actual implementations in practical Gröbner basis computation for ideals as well as for syzygy modules.

Acknowledgement. The authors would like to thank Dingkang Wang, Yao Sun and Lei Huang for helpful discussions as well as the referees for useful comments.

REFERENCES

[1] A. Arri and J. Perry, *The F5 criterion revised*, J. Symbolic Comput., 46 (2011), pp. 1017–1029.

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

[9] 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.

[10] 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.

[11] 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.

[12] J. Gash, *On efficient computation of gröbner bases*, Ph.D. dissertation, Indiana University, Bloomington, IN, (2008).

[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] X. Ma, Y. Sun, D. Wang, and Y. Zhang, *A signature-based algorithm for computing gröbner bases in solvable polynomial algebras*, in ISSAC'12: Proceedings of the 2012 International Symposium on Symbolic and Algebraic Computation, Grenoble, France, 2012, ACM.

[17] 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.

[18] 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.

[19] B. H. Roune and M. Stillman, *Practical gröbner basis computation*, in ISSAC'12: Proceedings of the 2012 International Symposium on Symbolic and Algebraic Computation, Grenoble, France, 2012, ACM.

[20] T. Stegers, *Faugère's F5 algorithm revisited*, Cryptology ePrint Archive, Report 2006/404 (2006).

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

[22] ———, *A generalized criterion for signature related Gröbner basis algorithms*, CoRR, arXiv:1101.3382 (2011).