# Gröbner Bases Algorithm

Iyad A. Ajwa  Zhuojun Liu  Paul S. Wang*
Institute for Computational Mathematics
Department of Mathematics & Computer Science
Kent State University
Kent, Ohio 44242, U.S.A.

February 27, 2003

**Abstract.** Gröbner Bases is a technique that provides algorithmic solutions to a variety of problems in Commutative Algebra and Algebraic Geometry. In this introductory tutorial the basic algorithms as well as their generalization for computing Gröbner basis of a set of multivariate polynomials are presented. The Gröbner basis technique is applied to solve systems of polynomial equations in several variables. This technical report investigates this application.

## 1 Introduction

The origins of algebra and algorithms date back to the ninth century. Working on polynomial equations, the mathematician *Mohammed ibn Musa al-Khawarizmi* wrote his famous book *kitab al-jabr wa'l muqabala* in Baghdad. The book discusses symbolic methods for solving polynomial equations. The words *algebra* and *algorithms* are actually the westernization of the words *al-jabr* and *al-Khawarizmi'yah* respectively [1]. Until the nineteen sixties, Algebra was concerned with constructive methods. With the discovery of computers and their development, algebraic algorithms have been recognized to play a central role in computer science. The recent advances in computer technology coupled with the ancient interest in algebraic algorithms have made it necessary to study computer related topics to algorithms, such as their efficiency, implementation, hardware and software needs and so on. This has lead to the establishment of *Computer Algebra*, a field of study that extends deeply into both mathematics and computer science.

Over the years, new concepts and results have developed in the area of Computer Algebra and computer algebraists made significant contributions to the fields of Mathematics and Computer Science. Among these contributions, an outstanding example is the theory and algorithms for Gröbner .

*Work reported herein has been supported in part by the National Science Foundation under Grant CCR-9201800

The concept of Gröbner Bases was introduced by Bruno Buchberger (1965) in the context of his work on performing algorithmic computations in residue classes of polynomial rings. Buchberger's algorithm for computing Gröbner Bases is a powerful tool for solving many important problems in polynomial ideal theory. It has been extensively studied, developed, refined, and it has been implemented on most computer algebra systems.

This tutorial is divided into six sections. We start by giving the reader the necessary background for understanding the theory of Gröbner basis. Mathematical notations and definitions follow in Section 2. The important concept of monomial ordering and *polynomial reduction*, a corner stone in the Gröbner basis algorithm, are explained in Sections 3 and 4. Buchberger's original algorithm and a modified version of it are given in Section 5. The last section presents an important application of the Gröbner basis algorithm: solution of systems of polynomial equations in several variables.

## 2 Mathematical Notations

The theory of Gröbner bases is centered around the concept of ideals generated by finite sets of multivariate polynomials. Studying polynomials is essential to understand the relationship between algebra and geometry. Therefore, we start our discussion by defining some basic algebraic structures, and move on to the notion of ideals.

**Definition.** A *commutative ring* $\langle R, +, \cdot \rangle$ is a set $R$ with the two binary operations addition $(+)$ and multiplication $(\cdot)$ defined on $R$ such that $\langle R, + \rangle$ is a commutative group, $\cdot$ is commutative and associative, and the distributive law $a \cdot (b+c) = a \cdot b + a \cdot c$ holds $\forall a, b, c \in R$.

**Example.** $\langle Z, +, \cdot \rangle$ is a commutative ring.
Shortly, we will define the most important ring to this tutorial.

**Definition.** Let $\langle R, +, \cdot \rangle$ be a commutative ring with a multiplicative identity. $\langle R, +, \cdot \rangle$ is called a *field* if every nonzero element of $R$ has a multiplicative inverse in $R$.

**Example.** $\langle \mathrm{Q}, +, \cdot \rangle$, $\langle \mathrm{R}, +, \cdot \rangle$, $\langle \mathrm{C}, +, \cdot \rangle$ are fields. However, $\langle \mathrm{Z}, +, \cdot \rangle$ is not a field.

**Definition.** Let $N$ denote the non-negative integers. Let $\alpha = (\alpha_1, \ldots, \alpha_n)$ be a power vector in $N^n$, and let $x_1, x_2, \ldots, x_n$ be any $n$ variables. Then a *monomial* $x^\alpha$ in $x_1, x_2, \ldots, x_n$ is defined as the product $x^\alpha = x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_n^{\alpha_n}$. Moreover, the *total degree* of the monomial $x^\alpha$ is defined as $|\alpha| = \alpha_1 + \cdots + \alpha_n$.

**Example.** $x^5y^2z$, $x^4y^3$, $y^5$, and $x^2z$ are monomials in $x, y, z$. They are of total degrees 8, 7, 5, and 3 respectively.

In this tutorial, unless otherwise specified, all monomials will be in $x_1, x_2, \ldots, x_n$.

**Definition.** A multivariate *polynomial* $f$ in $x_1, x_2, \ldots, x_n$ with coefficients in a field $k$ is a finite linear combination, $f(x_1, x_2, \ldots, x_n) = \sum_\alpha a_\alpha x^\alpha$, of monomials $x^\alpha$ and coefficients $a_\alpha \in k$. The *total degree* of the polynomial $f$ is defined as the maximum $|\alpha|$ such that $a_\alpha \neq 0$.

**Example.** $f(x, y, x) = x^5y^2z - x^4y^3 + y^5 + x^2z - y^3z + xy + 2x - 5z + 3$ is a polynomial in $x, y, z$. The total degree of $f$ is 8 since $|\alpha| = |(5, 2, 1)| = 8$ is maximum of all power vectors of monomials with nonzero coefficients. It is the power vector of the monomial $x^5y^2z$.

**Example.** The set of all multivariate polynomials in $x_1, x_2, \ldots, x_n$ with coefficients in a field $k$ is denoted by $k[x_1, x_2, \ldots, x_n]$. It is easy to verify that $k[x_1, x_2, \ldots, x_n]$ forms a commutative ring. Hence, it will be called a *polynomial ring*.

**Definition.** Let $\langle R, +, \cdot \rangle$ be a commutative ring. A nonempty subset $I \subset R$ is called an *ideal* if $I$ is closed under addition and is closed under inside-outside multiplication.

**Definition.** Let $F = \{f_1, \ldots, f_s\}$ be a set of multivariate polynomials. Then the ideal generated by $F$, denoted by $I = \langle F \rangle$, is given by:

$$\{\textstyle\sum_{i=1}^{s} h_i f_i : h_1, \ldots, h_s \in k[x_1, x_2, \ldots, x_n]\}.$$

The polynomials $f_1, \ldots, f_s$ are called a *basis* for the ideal they generate and, since $F$ is finite, we say the ideal is finitely generated.

Thus, the ideal generated by a family of generators consists of the set of linear combinations of these generators, with polynomial coefficients. Moreover, we say that two polynomials are equivalent with respect to an ideal if their difference belongs to the ideal.

Hilbert Bases Theorem proves that every ideal is finitely generated. Obviously, there are several bases for one ideal. We can always add any linear combination of the generators, or suppress one of them if it is a linear combination of the others. However, among the different bases of an ideal, stands a very useful basis: Gröbner basis.

As we will see, a basic ingredient to the theory of Gröbner basis is the idea of polynomial reduction to compute a suitably defined normal form of a given polynomial. Before one can talk about polynomial reduction, the notion of monomial ordering should be introduced. This important idea is studied next.

## 3 Monomial Ordering

Let us consider polynomials in the variables $x_1, \ldots, x_n$, with coefficients in a field $k$. We will assume the following ordering on the variables $x_1, \ldots, x_n$:

$$x_1 > x_2 > \cdots > x_{n-1} > x_n$$

As we will see, the computation of Gröbner bases varies substantially when we use different monomial orderings.

**Definition.** A total ordering, $>$, on $N^n$ is called *admissible* if the following two conditions are satisfied:

1. $\forall \alpha \in N^n, \alpha > 0$.
2. $\forall \alpha, \beta, \gamma \in N^n, \alpha > \beta \Longrightarrow \alpha + \gamma > \beta + \gamma$.

Clearly, an admissible ordering establishes a one-to-one correspondence between $N^n$ and the monomials $x^\alpha = x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ in $k[x]$. In other words, if $>$ is an admissible ordering on $N^n$ then $>$ is an ordering on the monomials. i.e., $\alpha > \beta \Longrightarrow x^\alpha > x^\beta$.
There are several monomial orederings. Following is the description of most important three:

**Definition.** Let $\alpha$ and $\beta$ be in $N^n$.

1. **Lexicographic Order:** $\alpha >_{lex} \beta$ if and only if the left-most nonzero entry in $\alpha - \beta$ is positive.
2. **Graded Lex Order:** $\alpha >_{grlex} \beta$ if and only if $|\alpha| > |\beta|$ or ($|\alpha| = |\beta|$ and $\alpha >_{lex} \beta$).
3. **Graded Reverse Lex Order:** $\alpha >_{grevlex} \beta$ if and only if $|\alpha| > |\beta|$ or ($|\alpha| = |\beta|$ and the right-most nonzero entry in $\alpha - \beta$ is negative.

**Example.**

- $(3, 2, 1) = \alpha >_{lex} \beta = (1, 2, 4)$ since in $\alpha - \beta = (2, 0, -3)$ the left-most nonzero entry is positive. Hence $x^3y^2z >_{lex} xy^2z^4$.
- $(2, 4, 1) = \alpha >_{grlex} \beta = (1, 6, 0)$ since $|\alpha| = 7 = |\beta|$ and $\alpha >_{lex} \beta$ and left-most nonzero entry in $\alpha - \beta = (1, -2, 1)$ is positive. Hence $x^2y^4z >_{grlex} xy^6$.
- $(1, 3, 1) = \alpha >_{grevlex} \beta = (1, 2, 2)$ since $|\alpha| = 5 = |\beta|$ and the right-most entry in $\alpha - \beta = (0, 1, -1)$ is negative. Hence

**Definition.** Assume an arbitrary admissible ordering $>$ is fixed. Given a nonzero polynomial $f \in k[x_1, x_2, \ldots, x_n]$, we define:

- The *multidegree* of $f$ as: $\text{multideg}(f) = \max(\alpha \in N^n : a_\alpha \neq 0)$.
- The *leading monomial* of $f$ as: $\text{LM}(f) = x^{multideg(f)}$.
- The *leading coefficient* of $f$ as: $\text{LC}(f) = a_{multideg(f)}$.
- The *leading term* of $f$ as: $\text{LT}(f) = \text{LC}(f) \cdot \text{LM}(f)$.

**Example.** Consider the polynomial $f(x, y, z) = 2x^2y^8 - 3x^5yz^4 + xyz^3 - xy^4$ in $k[x, y, z]$. Then

- with respect to the lex order, $f$ is reordered in decreasing order as:
  $f(x,y,z) = -3x^5yz^4 + 2x^2y^8 - xy^4 + xyz^3$
  multideg($f$) = (5,1,4), LM($f$) = $x^5yz^4$, LC($f$)=$-3$, LT($f$) = $-3x^5yz^4$.

- With respect to the grlex order, $f$ is reordered in decreasing order as:
  $f(x,y,z) = -3x^5yz^4 + 2x^2y^8 - xy^4 + xyz^3$
  multideg($f$) = (5,1,4), LM($f$) = $x^5yz^4$, LC($f$)=$-3$, LT($f$) = $-3x^5yz^4$.

- With respect to the grevlex order, $f$ is reordered in decreasing order as:
  $f(x,y,z) = 2x^2y^8 - 3x^5yz^4 - xy^4 + xyz^3$
  multideg($f$) = (2,8,0), LM($f$) = $x^2y^8$, LC($f$) = 2, LT($f$) = $2x^2y^8$.

# 4 Polynomial Reduction

As we will see, polynomial reduction is the corner stone in the Gröbner bases algorithm. It is the most computationally intensive part of the algorithm. Buchberger [3] viewed polynomial reduction as a one step in a generalized division. A polynomial $g$ *reduces* to another polynomial $h$ modulo some polynomial set $F$, denoted as $g \longrightarrow_F h$, if and only if the LT($g$) can be deleted by the subtraction of an appropriate multiple of: an appropriate polynomial $f$ in $F$, a monomial $u$ where $u$ =LM($g$)/LM($f$), and a scalar $b$ in $k$, where $b$ =LC($g$)/LC($f$), yielding $h$.

**Definition.** $g \longrightarrow_F h$ if and only if there exists $f \in F$, $b$, and $u$ such that $h = g - buf$. Otherwise, $g$ is called *irreducible* modulo $F$.

In other words, $g$ is called *irreducible* modulo $F$ if no leading monomial of an element of $F$ divides the leading monomial of $g$. On the other hand if $g$ is reducible modulo $F$ then we can subtract from it a multiple of an element of $F$ to eliminate its leading monomial and to get a new leading monomial less than the leading monomial of $g$. This new polynomial is equivalent to $g$ with respect to the ideal generated by $F$.

**Example.** Let $F = \{f_1, f_2\}$, where $f_1 = xy^2 - x$ and $f_2 = x - y^3$. Consider the polynomial $g = x^7y^2 + x^3y^2 - y + 1$. These polynomials are ordered with respect to the lex order.
Choose $f = f_1, u = x^6, b = 1$, we obtain the polynomial $h = g - buf = x^7 + x^3y^2 - y + 1$.
Thus, $g \longrightarrow_F h$.

**Definition.** A polynomial $h$ is a *normal form* of $g$ if and only if $g \longrightarrow_F^+ h$ and $h$ is irreducible modulo $F$.

**Example.** Let $F = \{f_1, f_2\}$, where $f_1 = xy - 1$ and $f_2 = y^2 - 1$. Consider the polynomial $g = x^2y + xy^2 + y^2$. These polynomials are ordered with respect to the lex order with $x > y$. Clearly, $g$ is reducible modulo $F$. If we continue the reduction process, we end up with the polynomial $h = x + y + 1$ which is irreducible modulo $F$. Thus $h$ is a normal form of $g$.

The definition of polynomial reduction involves only the leading term of $g$. However, it is possible to eliminate some other monomials of $g$ to make the linear combination smaller. This

leads to the following definition.

**Definition.** A polynomial $g$ is *completely reduced* with respect to $F$ if no term in $g$ is divisible by any of the $\mathrm{LT}(f_i)$ for all $f_i \in F$.

As we see, one way to reduce a polynomial modulo a polynomial set $F$ is to generalize the division algorithm for $k[x_1]$ which states as follows:

**Proposition [4].** If $f, g \in k[x_1]$ and $g \neq 0$, then $\exists$ unique $q, r \in k[x_1]$ such that $f = qg + r$ and either $r = 0$ or $deg(r) < deg(g)$.

The proof is straightforward and can be found in ordinary algebra texts. To generalize this division algorithm, we need to divide a polynomial $f \in k[x_1, x_2, \ldots, x_n]$ by a set of polynomials $F = f_1, \ldots, f_s \subset k[x_1, x_2, \ldots, x_n]$. Following is the formal statement:

**Theorem [4].** Let $F = (f_1, \ldots, f_s)$ be an ordered $s$-tuple of polynomials in $k[x_1, x_2, \ldots, x_n]$. Then if $f$ is a polynomial in $k[x_1, x_2, \ldots, x_n]$, then $\exists q_1, \ldots, q_s, r \in k[x_1, x_2, \ldots, x_n]$ such that $f = q_1 f_1 + \cdots + q_s f_s + r$ and either $r = 0$ or $r$ is a completely reduced polynomial.

In the following section, this division algorithm is given.
Now, we turn to a term that plays an essential role in the theory of Gröbner bases: *S-polynomials*. Recall that the l.c.m. of two monomials is the product of all the variables, each to a power which is maximum of its powers in the two monomials.

**Definition.** Given two polynomials $f, g \in k[x_1, x_2, \ldots, x_n]$. Let $J = l.c.m(LM(f), LM(g))$. We define the *S-polynomial* of $f$ and $g$ as the linear combination

$$\textit{S-poly(f,g)} = \frac{J}{LT(f)} \cdot f - \frac{J}{LT(g)} \cdot g$$

Since $J/LT(f)$ and $J/LT(g)$ are monomials, then the *S-poly(f,g)* is a linear combination with polynomial coefficients of $f$ and $g$, and belongs to the same ideal generated by $f$ and $g$. The above mentioned definition indicates that $S$-polynomials are cross product of leading terms and are constructed to cancel leading terms. The leading terms of the two components of *S-poly(f,g)* are equal and therefore, cancel each other.

**Example.** Let $F = \{f_1, f_2\}$, where $f_1 = xy^2z - xyz$ and $f_2 = x^2yz - z^2$. These polynomials are ordered with respect to the pure lexicographic order. $LM(f_1) = xy^2z$, and the $LM(f_2) = x^2yz$. Let $J = l.c.m.(xy^2z, x^2yz) = x^2y^2z$. Then

$$\begin{aligned} \textit{S-poly}(f_1, f_2) &= \tfrac{J}{LT(f_1)} \cdot f_1 - \tfrac{J}{LT(f_2)} \cdot f_2 \\ &= \tfrac{x^2y^2z}{xy^2z} \cdot f_1 - \tfrac{x^2y^2z}{x^2yz} \cdot f_2 \\ &= x \cdot f_1 - y \cdot f_2 \\ &= -x^2yz + yz^2 \end{aligned}$$

Now, let us take three examples. In the first example, let us consider two polynomials $f_1$ and $f_2$ such that $LT(f_1)$ and $LT(f_2)$ are relatively prime and $LC(f_1) = LC(f_2) = 1$. As we will see, the *LM(S-poly)*$(f_1, f_2)$ is a multiple of the $LM(f_1)$ or the $LM(f_2)$.
In the second and third examples, we will consider two polynomials $f_1$ and $f_2$ such that the $gcd(f_1, f_2) = 1$. As we will see, the above mentioned observation may or may not work in this case.

**Example.** Let $F = \{f_1, f_2\}$, where $f_1 = xy + z^3$ and $f_2 = z^2 - 3z$. These polynomials are ordered with respect to the pure lexicographic order. $LM(f_1) = xy$, and the $LM(f_2) = z^2$. Let $J = l.c.m.(xy, z^2) = xyz^2$. Then

$$\begin{aligned} \textit{S-poly}(f_1, f_2) &= \tfrac{xyz^2}{xy} \cdot f_1 - \tfrac{xyz^2}{z^2} \cdot f_2 \\ &= -3xyz + z^5 \end{aligned}$$

**Example.** Let $F = \{f_1, f_2\}$, where $f_1 = 4x^2z - 7y^2$ and $f_2 = xyz^2 + 3xz^4$. These polynomials are ordered with respect to the pure lexicographic order. $LM(f_1) = x^2z$, and the $LM(f_2) = xyz^2$. Let $J = l.c.m.(x^2z, xyz^2) = x^2yz^2$. One can check that the $gcd(f_1, f_2) = 1$. Then

$$\begin{aligned} \textit{S-poly}(f_1, f_2) &= \tfrac{x^2yz^2}{x^2z} \cdot f_1 - \tfrac{x^2yz^2}{xyz^2} \cdot f_2 \\ &= -12x^2z^4 - 7y^3z \end{aligned}$$

**Example.** Let $F = \{f_1, f_2\}$, where $f_1 = x^4y - z^2$ and $f_2 = 3xz^2 - y$. These These polynomials are ordered with respect to the pure lexicographic order. $LM(f_1) = x^4y$, and the $LM(f_2) = xz^2$. Let $J = l.c.m.(x^4y, xz^2) = x^4yz^2$. One can check that the $gcd(f_1, f_2) = 1$. Then

$$\begin{aligned} \textit{S-poly}(f_1, f_2) &= \tfrac{x^4yz^2}{x^4y} \cdot f_1 - \tfrac{x^4yz^2}{3xz^2} \cdot f_2 \\ &= x^3y^2 - 3z^4 \end{aligned}$$

There are several equivalent definitions for Gröbner bases. Following is the definition of a Gröbner basis as it was originally given by Buchberger [3].

**Definition.** Given a finite set of polynomials, $F$. Then $F$ is a Gröbner basis if and only if $\forall g, h_1, h_2$ if $h_1$ and $h_2$ are normal forms of $g$ modulo $F$ then $h_1 = h_2$.

**Theorem.** Let $F = \{f_1, \ldots, f_s\}$ be a finite set of polynomials. Let $I$ be the ideal generated by $F$. The following are equivalent:

- F is a Gröbner basis.
- $\forall f_i, f_j \in F$ : *S-poly*$(f_i, f_j)$ reduces to zero modulo $F$.
- Every reduction of an $f$ of $I$ to a reduced polynomial with respect to $F$ always gives zero.

Indeed, this criterion is the driving force behind Gröbner bases method. To check if a basis is a Gröbner basis, all we have to do is compute all *S-poly*$(f_i, f_j)$ and see if all reduce to zero or not. Another useful use of this theorem is that it gives us the tools to construct a Gröbner basis. If one *S-poly* does not reduce to zero, it still can be added to the basis without changing the ideal generated. This is because it is a linear combination of two polynomials of the basis. Once this *S-poly* is added to the basis, it will reduce to zero. However, there will be new *S-polynomials* to be considered. This process comes to an end as it was shown by Buchberger. This is the essence of Buchberger's algorithm to compute Gröbner bases. In the next section, we discuss this algorithm.

# 5 Algorithms

The above mentioned theorem formulates the algorithmic criterion for Gröbner bases. The following algorithm is the original algorithm given by Buchberger in his Ph.D. dissertation. Buchberger and interested researchers extensively worked on the algorithm and several developments were given. The refined algorithm will be discussed later.

**Algorithm Buchberger [3].**
*Input*: A polynomial set $F = (f_1, \ldots, f_n)$ that generates an ideal $I$.
*Output*: A Gröbner basis $G = (g_1, \ldots, g_t)$ that generates the same ideal $I$ with $F \subset G$.

$G := F$
$M := \{\{f_i, f_j\} | f_i, f_j \in G \text{ and } f_i \neq f_j\}$
Repeat
    $\{p, q\} :=$ a pair in $M$
    $M := M - \{\{p, q\}\}$
    *S := Spoly(p,q)*
    *h := NormalForm(S,G)*
    IF $h \neq 0$ THEN
        $M := M \bigcup \{\{g, h\}\ \forall g \in G\}$
        $G := G \bigcup \{h\}$
Until $M = \emptyset$

So, the idea of the algorithm is fairly clear. We initialize the Gröbner basis, $G$, to the original set of polynomials. Form the set, $M$, of all pairs of polynomials in $G$. Pick a pair {p,q} from $M$. Compute the *S-poly(p,q)* and reduce it modulo $G$ to the polynomial $h$. If $h$ is nonzero, then we add it to the basis, $G$, and we update the set of pairs, $M$, by forming and adding new pairs, $\{h,g\}$, for all $g \in G$. We repeat this process until any computed $h$ equals zero.

The computation of *S-polynomials* is straightforward. The computation of the $Normalform(S, G)$ is possible through any normal form algorithm. One algorithm that has been implemented in several computer algebra systems is the generalized division algorithm mentioned in the previous section. The $Normalform(S, G)$ can be taken as the remainder $r$ of dividing the polynomial $S$ by the set $G$. Following is this algorithm.

**Algorithm Generalized Division [4].**
*Input*: A polynomial set $F = (f_1, \ldots, f_s)$, and any nonzero polynomial $f$ in $k[x_1, x_2, \ldots, x_n]$.
*Output*: The remainder, $r$, of dividing $f$ by $F$.
The quotients $q_1, q_2, \ldots, q_s$ such that $f = q_1 f_1 + \cdots + q_s f_s + r$ with either $r = 0$ or $r$ is a completely reduced polynomial with respect to $F$.

$q_i := 0$; for $i := 1, \ldots, s$
$r := 0$
$p := f$
Repeat
    $i := 1$
    dividing := true
    While $(i \leq s)$ and (dividing) do
        If LT($f_i$) divides LT($p$) then
            $u :=$LT($p$)/LT($f_i$)
            $q_i := q_i + u$
            $p := p - u \cdot f_i$
            dividing := false
        else
            $i := i + 1$
    If not dividing then
        $r :=$ $r+$ LT($p$)
        $p :=$ $p-$ LT($p$)
Until $p = 0$

Indeed, this algorithm is a generalized form of the high school division algorithm. As long as the LT of a divisor divides the LT of an intermediate dividend, the algorithm proceeds as in the one-variable case. If no LT($f_i$) divides LT($p$), then the algorithm removes LT($p$) from $p$ and adds it to $r$.
It should be noted here that unlike the division algorithm in $k[x_1]$, the generalized division algorithm does not have several of the nice properties: the remainder is not uniquely characterized by the requirement that if it is nonzero then none of its terms is divisible by LT($f_i$), the $q_i$ are not unique and they change if the $f_i$ are rearranged. However, this algorithm has

the properties of its one-variable counterpart when it is coupled with Gröbner bases. Let us take an example on how to compute Gröbner basis for a basis.

**Example.** Let $G = \{g_1, g_2\}$, where $g_1 = 4x^2z - 7y^2$ and $g_2 = xyz^2 + 3xz^4$. Let us use the pure lexicographic order with $x > y > z$. The only *S-polynomial* to be considered is *S-poly*$(g_1, g_2) = -12x^2z^4 - 7y^3z$. This polynomial is non-zero and its remainder on division by $G = (g_1, g_2)$ is $g_3 = -21z^3y^2 - 7y^3z$, which is non-zero. Hence, $g_3$ should be included in our generating set. Hence, $G = (g_1, g_2, g_3)$, and the *S-polynomials* to be considered are *S-poly*$(g_1, g_2)$, *S-poly*$(g_1, g_3)$, *S-poly*$(g_2, g_3)$.
Now, *S-poly*$(g_1, g_2)$ reduces to 0 modulo the new $G = (g_1, g_2, g_3)$, and does not have to be considered. *S-poly*$(g_1, g_3) = 49y^5 + 84x^2z^3y^2$ and reduces to $g_4 = 49y^5 + 1323z^6y^2$ which is non-zero. Hence, we must add $g_4$ to our generating set. If we let $G = (g_1, g_2, g_3, g_4)$, then *S-poly*$(g_1, g_2)$, *S-poly*$(g_1, g_3)$ *S-poly*$(g_1, g_4)$, *S-poly*$(g_2, g_3)$, *S-poly*$(g_2, g_4)$, and *S-poly*$(g_3, g_4)$ all reduce to zero modulo the new $G$. Hence, we stop and the Gröbner basis for the ideal generated by the old $G$ is the new $G = (g_1, g_2, g_3, g_4)$.

## 6 Improved Gröbner Bases

Now, we know the idea of Gröbner bases, we turn to the improvements of Buchberger's original algorithm. Following are some observations.

1. As we saw in the above example, if one $S$-polynomial reduces to 0, the algorithm still recomputes its remainder in each iteration of the loop thereafter. This should not happen because once an $S$-polynomial reduces to 0, it is going to reduce to 0 even if we adjoin new polynomials to $F$.

2. Buchberger's Criterion I [2]: In the process of picking a pair $\{f_i, f_j\}$, choose a pair $\{f_i, f_j\}$ such that LCM(LM($f_i$),LM($f_j$)) is minimal among all the pairs.

3. Also, as we saw in an example in previous section, there are $S$-polynomials that may be ignored and we do not need to compute their normal form because they are guaranteed to reduce to zero modulo $F$. If the LM($f_i$) and LM($f_j$) are relatively prime, then $S$-poly$(f_i, f_j)$ reduces to 0 modulo $F$. Thus, pick a pair $\{f_i, f_j\}$ such that LM($f_i$) and LM($f_j$) are not relatively prime. This has become to be known as Buchberger's Criterion II [2].

4. Buchberger's Criterion III [2]: If there is an element $f_k$ of the basis such that the LM($f_k$) divides LCM(LM($f_i$),LM($f_j$)) and if the $S$-poly$(f_i, f_k)$ and the $S$-poly$(f_j, f_k)$ have already been considered, then $S$-poly$(f_i, f_j)$ reduces to zero and hence could be ignored.

5. The output $G$ is not a minimal (reduced) Gröbner basis. There are redundant polynomials that can be eliminated. If LT($g_i$) divides LT($g_j$) then $g_j$ can be eliminated from the basis and $G - \{g_j\}$ is still a Gröbner basis. So, each time a new polynomial is adjoined to the basis, all the other polynomials may be reduced using also the new

polynomial. This results in many polynomials deleted and the resulting basis will be a reduced Gröbner basis.

With these observations in mind, the following modified algorithm computes the Gröbner basis.

**Algorithm Buchberger (modified) [4]:**
*Input*: A polynomial set $F = (f_1, \ldots, f_n)$ that generates an ideal $I$.
*Output*: A Gröbner basis $G = (g_1, \ldots, g_t)$ that generates the same ideal $I$ with $F \subset G$.

$G := F$
$M := \{\{f_i, f_j\} | 1 \leq i < j \leq s\}$
$t := s$
Repeat
    $\{f_i, f_j\}$ := a pair in $M$
    IF (LCM(LM($f_i$),LM($f_j$)) $\neq$ LM($f_i$)·LM($f_j$)) AND NOT(Criterion($f_i, f_j$,M) then
        $S$ := *S-polynomial*($f_i, f_j$)
        $h$ := NormalForm*(G,S)*
        IF $h \neq 0$ THEN
            $t := t + 1$
            $f_t := h$
            $M := M \bigcup \{\{i, t\} \forall 1 \leq < i \leq t - 1\}$
            $G := G \bigcup \{f_t\}$
        $M := M - \{\{f_i, f_j\}\}$
Until $M = \emptyset$

where Criterion($f_i, f_j$,M) is true provided the conditions in (4) above are met.

# 7 Applications

Gröbner bases algorithm has been intensively studied and more applications have been exploited. One of the most important applications is the use of Gröbner bases algorithm for solving systems of polynomial equations and answering questions about the solvability of such systems.

We assume that the systems of equations we are dealing with are in the variables $x_1, \ldots, x_n$ with the lexicographic order $x_1 > \ldots > x_n$. We begin by this definition.

**Definition.** Given $I = \langle f_1, \ldots, f_s \rangle \subset k[x_1, \ldots, x_n]$, the $i$th elimination ideal $I_i$ is the ideal of $k[x_i + 1, \ldots, x_n]$ defined by

$$I_i = I \bigcap k[x_{i+1}, \ldots, x_n].$$

Thus, $I_i$ consists of all consequences of $f_1 = \cdots = f_s = 0$ which eliminate the variables $x_1, \ldots, x_i$. We see that eliminating $x_1, \ldots, x_i$ means finding nonzero polynomials in the $i$th elimination ideal $I_i$, and the significance of Gröbner bases for solving systems of equations stems from the fact that, for Gröbner bases, it is simple to construct all of the elimination

ideals. During the construction of the Gröbner bases, variables are eliminated successively. Also, the order of elimination corresponds to the ordering of the variables: $x_1$ is eliminated first, then $x_2$ is eliminated second, and so on.

To solve a system $F$ of polynomial equations (which determines the ideal $I = \langle F \rangle$) we proceed as follows:

1. Compute the Gröbner basis, $G$, of $I$ with respect to the lex order.
2. Find the roots of the generator in $x_n$ by applying one-variable techniques.
3. Apply back substitution to find the roots of all generators in $G$.
4. Roots of generators in $G$ are extended to solutions of the original equations.

**Example.** Consider the following system of equations:

$$\begin{aligned} x^2 + y^2 + z^2 &= 1 \\ x^2 + y^2 + z^2 &= 2x \\ 2x - 3y - z &= 0 \end{aligned}$$

Let $I$ be the ideal

$$I = \langle x^2 + y^2 + z^2 - 1, x^2 + y^2 + z^2 - 2x, 2x - 3y - z \rangle,$$

then a Gröbner basis for $I$ with respect to the lex order is $G = (g_1, g_2, g_3)$, where
$g_1 = 2x - 1$,
$g_2 = 3y + z - 1$,
$g_3 = 40z^2 - 8z - 23$.

If we examine the generators in $G$, we notice the following:

- We have exactly one generator, $g_3$, in the variable $x_n = z$ alone. The other variables have been eliminated during the process of finding the Gröbner basis. This polynomial has a finite number of roots which can be determined using any one-variable technique.
- There is exactly one generator in the variables $x_{n-1} = y$ and $x_n = z$. Since we have all possible roots of $z$, we can determine the roots of $y$. It is possible that we have a generator in $x_{n-1}$ alone.
- Generator $g_1$ is in $x$ alone. All roots of $x$ can be computed.
- The process of back substitution continues until all roots of generators are determined.
- We also note that when Gröbner bases are computed using the lex order, the variables are eliminated in a nice fashion.

**Example.** Consider the system of equations:

$$\begin{aligned} x^2y - z^3 &= 0 \\ 2xy - 4z &= 1 \\ z - y^2 &= 0 \\ x^3 - 4zy &= 0 \end{aligned}$$

Let $I$ be the ideal

$$I = \langle x^2y - z^3, 2xy - 4z - 1, z - y^2, x^3 - 4zy \rangle,$$

then a Gröbner basis for $I$ with respect to the lex order is $G = (1)$. It is well known that a system has a solution if and only if 1 is not a member of the ideal generated by the set of polynomials in the system. Buchberger proved that a system $F$ is unsolvable if and only if $1 \in G$, where $G$ is the Gröbner basis generated by the set of polynomials in the system.
Next, we consider a system with infinitely many solutions.

**Example.** Consider the system of equations:

$$\begin{aligned} t^2 + x^2 + y^2 + z^2 &= 0 \quad (1) \\ t^2 + 2x^2 - xy - z^2 &= 0 \quad (2) \\ t + y^3 - z^3 &= 0 \quad (3) \\ & \quad (4) \end{aligned}$$

Let $I$ be the ideal

$$I = \langle t^2 + x^2 + y^2 + z^2, t^2 + 2x^2 - xy - z^2, t + y^3 - z^3 \rangle,$$

then a Gröbner basis for $I$ with respect to the lex order is $G = (g_1, g_2, g_3, g_4, g_5)$, where
$g_1 = x^2 + y^2 + z^2 + y^6 - 2y^3z^3 + z^6$
$g_2 = 2y^2 + 3z^2 + y^6 - 2y^3z^3 + z^6 + xy$
$g_3 = -5y^3 - 7yz^2 - 5y^7 + 10y^4z^3 - 3yz^6 + 6z^5y^2 + 4y^8z^3 - 5y^5z^6 + 2z^9y^2 - 3y^5z^2 - y^{11} + 3xz^2 + xz^6$
$g_4 = t + y^3 - z^3$
$g_5 = 13y^2z^2 + 9z^4 + 6y^6z^2 - 12z^5y^3 + 6z^8 + 5z^6y^2 + 6z^6y^6 - 4z^9y^3 + z^{12} + 5y^8 - 10y^5z^3 - 4y^9z^3 + y^{12}$

Let us examine these generators carefully:

- There is no generator in the variable $x_n = z$ alone. $g_5$ is in $y$ and $z$.
- Consider the leading monomials of $g_1, g_2, g_3, g_4$, and $g_5$. They are $x^2, xy, xz^6, t$, and $y^{12}$ respectively. We note that a monomial of the form $x_i^h$ occurs among these leading monomials for $x_i = t, x_i = x, x_i = y$ but there is no monomial of the form $x_i^h$ for $x_i = z$. Buchberger [3] proved the following fact:
  *A system,* F*, has finitely many solutions if and only if for all* $i (1 \leq i \leq n)$*: a power product of the form* $x_i^h$ *occurs among the leading power products of the polynomials in* $G = GB(F)$*, where* $n$ *is the number of polynomials in* F*.*

# 8 Conclusion

In conclusion, we have presented the reader with enough information on this important technique of Gröbner bases. The reader is referred to the home-page for *ICM* on *SymbolicNet* using Mosaic for more applications of Gröbner bases and a comprehensive list of references. The URL to be used is: `http://symbolicnet.mcs.kent.edu/areas/groebner/index.html`.

# References


[1] T. Becker and V. Weispfenning. Gröbner Bases: A Computational Approach to Commutative Algebra, Springer-Verlag, 1993.

[2] B. Buchberger. A Criterion for Detecting Unnecessary Reductions in the Construction of Gröbner Bases. *Lecture Notes in Computer Science, vol. 72*. Springer-Verlag, 1979.

[3] B. Buchberger. Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory. In *Recent Trends in Multidimensional Systems Theory*, edited by N. K. Bose. Chapter 6, pp. 184-232. D. Reidel Publishing Company, 1986.

[4] D. Cox, J. Little, and D. O'Shea. Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra. Springer-Verlag, 1991.

[5] J. H. Davenport, Y. Siret, abd E. Tournier. Computer Algebra: Systems and Algorithms for Algebraic Computation. Second Edition. Chapter 3, pp. 111-122. Academic Press, 1993.