GROEBNER BASIS CONVERSION USING THE FGLM ALGORITHM

PHILIP BENGE, VALERIE BURKS, NICHOLAS COBAR

1. Introduction

This paper will examine the method of the FGLM algorithm to convert a Groebner basis from one monomial order to another, and how it is used to solve systems of polynomial equations. We proceed as outlined in [1] and [2]. We will begin by studying ideals, which are generated by these systems of polynomials, called bases.

Definition 1.1. Let k be a field. A subset $I \subset k[x_1, \dots, x_n]$ is an **ideal** if it satisfies:

(i) $0 \in I$.
(ii) If $f, g \in I$, then $f + g \in I$.
(iii) If $f \in I$ and $h \in k[x_1, \dots, x_n]$, then $h \cdot f \in I$.

Definition 1.2. If $f_1, \dots, f_s \in k[x_1, ..., x_n]$, then $I = \langle f_1, ..., f_s \rangle$ is an ideal of $k[x_1, ..., x_n]$. We will call $\langle f_1, ..., f_s \rangle$ the **ideal generated by** $f_1, ..., f_s$, where the polynomials $f_1, ..., f_s$ form a **basis** of I.

When we say $\langle f_1, ..., f_s \rangle$ we are referring to all of the elements that can be written as $\sum_{i=1}^{n} r_i f_i$ where the r_i are elements in the polynomial ring and the the f_i are elements in the ideal.

In order to solve a system of polynomial equations, we generate an ideal with the polynomials, and then find the set of common zeros, called the variety of the ideal. While different bases may generate the same ideal, the variety of the ideal will always be the same.

Definition 1.3. Let k be a field, and let $f_1, ..., f_s$ be polynomials in $k[x_1, ..., x_n]$. Then we set $\mathbf{V}\ (f_1, ..., f_s) = \{(a_1, ..., a_n) \in k^n :$ $f_i(a_1, ..., a_n) = 0$ for all $1 \leq i \leq s\}$. We call $\mathbf{V}\ (f_1, ..., f_s)$ the **affine variety** defined by $f_1, ..., f_s$.

For a simple example, consider the set of equations $\{x - z = 0, x + z - y = 0, x + y + z^2 - 4 = 0\}$ and the ideal they generate $I =< x - z, x - y + z, x + y + z^2 - 4 >$. The variety of this ideal is a finite number of points, specifically $\mathbf{V}(I) = \{(-4, -8, -4), (1, 2, 1)\}$. A variety does not have to be a finite set of points, but if it is, the ideal is called **zero-dimensional**.

In order to study these ideals and their bases, a standard must be set for the way in which the polynomials are written. This is essential because the division algorithm, which is associative in single variable calculations, is not associative when multiple variables are involved. Also, what would constitute a Groebner basis under one monomial order would not necessarily be the same under another.

Definition 1.4. *A* **monomial ordering** *on $k[x_1, \dots, x_n]$ is any relation $>$ on $\mathbb{Z}^n_{\geq 0}$, or equivalently, any relation on the set of monomials $x^\alpha, \alpha \in \mathbb{Z}^n_{\geq 0}$, satisfying:*

1. *$>$ is a total (or linear) ordering on $\mathbb{Z}^n_{\geq 0}$.*
2. *If $\alpha > \beta$ and $\gamma \in \mathbb{Z}^n_{\geq 0}$, then $\alpha + \gamma > \beta + \gamma$.*
3. *$>$ is a well-ordering on $\mathbb{Z}^n_{\geq 0}$. This means that every nonempty subset of $\mathbb{Z}^n_{\geq 0}$ has a smallest element under $>$.*

Within polynomials, the monomials are generally written in descending order. Within the monomial itself, individual variables are also written in descending order with $x_1 > x_2 > \dots > x_n$. To determine precedence under a monomial ordering, the exponents of the variables in a monomial are written as an ordered n-tuple to allow the reader to find the vector difference. An example of each type of monomial ordering will be given using the monomials $x_1x_2^2$ and $x_2^3x_3^4$. The n-tuples for these monomials (represented by α and β) are $\alpha = (1, 2, 0)$ and $\beta = (0, 3, 4)$, respectively.

Three commonly used monomial orders are lexicographic, graded lexicographic, and graded reverse lexicographic. In lexicographic order, or lex, a monomial with degree α is greater than another monomial with degree β if and only if the left-most, non-zero component of $\alpha - \beta$ is greater than 0. In graded lexicographic, or grlex, the monomial with the higher total degree is greater than the other. If the monomials' degrees are equal, then we use lex to break the tie. Finally, graded reverse lexicographic, also called grevlex, is exactly like grlex in that it first compares the degrees of the monomials, and the one with the higher degree is taken to be greater. However, in the event of equal degrees, it does not revert back to lex ordering to break ties. Instead a monomial with degree α is greater than another monomial with degree β if the right-most, non-zero component of $\alpha - \beta$ is less than 0.

Example 1.5.

1. $x_1x_2^2 >_{lex} x_2^3x_3^4$ since $\alpha - \beta = (1, -1, -4)$.
2. $x_1x_2^2 <_{grlex} x_2^3x_3^4$ since $\sum \alpha_i = 3$ and $\sum \beta_i = 7$.

(3) $x_1x_2^2 <_{grevlex} x_2^3x_3^4$ since $\sum \alpha_i = 3$ and $\sum \beta_i = 7$.

When studying ideals and bases, it is often necessary to examine relationships between the first terms of the polynomials involved. First, let $f = \sum_\alpha a_\alpha x^\alpha$ be a nonzero polynomial in $k[x_1, \ldots, x_n]$ and let $>$ be a monomial order. Then the **multidegree** of f is $multideg(f) = max\{\alpha \in \mathbb{Z}^n_{\geq 0} : a_\alpha \neq 0\}$ (the maximum is taken with respect to $>$). The **leading coefficient** of f is $LC(f) = a_{multideg(f)} \in$ k, and the **leading monomial** of f is $LM(f) = x^{multideg(f)}$ (with coefficient 1). The **leading term** of f is $LT(f) = LC(f) \cdot LM(f)$. For example, if $f = 3x^3y^2 - x^3y + x^2 - y$, then $multideg(f) = (3, 2, 0)$, $LC(f) = 3$, $LM(f) = x^3y^2$, and the $LT(f) = 3x^3y^2$ with respect to lexicographic order.

Definition 1.6. Fix a monomial order. A finite subset $G = \{g_1, ..., g_t\}$ of an ideal I is said to be a **Groebner basis** if
$< LT(g_1), ..., LT(g_t) > = < LT(I) >$. A Groebner basis is call **reduced** if

(1) $LC(p) = 1$ for all $p \in G$.
(2) For all $p \in G$, no monomial of p lies in $\langle LT(G - \{p\})\rangle$.

This means that the leading term of any element of I must be divisible by one of the $LT(g_i)$ for G to be a Groebner basis. To divide polynomials we use the following **division algorithm**:

Definition 1.7. Fix a monomial order $>$ on $\mathbb{Z}^n_{\geq 0}$, and let $F = (f_1, ..., f_s)$ be an ordered s-tuple of polynomials in $k[x_1, ..., x_n]$. Then every
$f \in k[x_1, ..., x_n]$ can be written as

$$f = a_1f_1 + ... + a_sf_s + r,$$

where $a_i, r \in k[x_1, ..., x_n]$, and either $r = 0$ or r is a linear combination, with coefficients in k, of monomials, none of which is divisible by any of $LT(f_1), ..., LT(f_s)$. We will call r a **remainder** of f on division by F. Furthermore if $a_i, f_i \neq 0$, then we have

$$multideg(f) \geq multideg(a_if_i).$$

For example, consider the ideal $I =< x^2 + y^2 + z^2 - 2x, x^3 - yz - x, x - y + 2z >$. Take the monomial order graded lexicographic. Then a Groebner basis, G, for I is $G = \{x - y + 2z, 2y^2 - 4yz + 5z^2 - 2y + 4z, 3yz^2 + 4z^3 - 10yz + 11z^2, 375z^4 + 974z^3 - 1460yz + 144z^2\}$.

Groebner bases have some very useful properties. In the case of determining if a function $f \in k[x_1, ..., x_n]$ is in an ideal, I, simply divide f by a Groebner basis for I and then $f \in I$ if and only if the remainder is zero. For example, let I and G be as stated above, and

$f = 14xy - 2z + 3$. Then dividing f by $G = \{g_1, g_2, g_3, g_4\}$ using the division algorithm yields $f = 14y \cdot g_1 + 7 \cdot g_2 + r$, where $r = -35z^2 + 14y - 30z + 3$. Since $r \neq 0$, $f \notin I$.

We would like to mention that a reduced Groebner basis is unique with respect to a monomial ordering, and from that, two ideals are equal if and only if their reduced Groebner bases with respect to a monomial ordering are equal.

We note that an ideal I is **zero-dimensional** if and only if it satisfies the following criteria:

Theorem 1.8. *Let V=**V**(I) be an affine variety in $\mathbb{C}^n$ and fix a monomial ordering in $\mathbb{C}[x_1, ..., x_n]$. Then the following statements are equivalent:*

1. *V is a finite set.*
2. *For each i, $1 \leq i \leq n$, there is some $m_i \geq 0$ such that $x_i^{m_i} \in < LT(I) >$.*
3. *Let G be a Groebner basis for I. Then for each i, $1 \leq i \leq n$, there is some $m_i \geq 0$ such that $x_i^{m_i} = LM(g_i)$ for some $g_i \in G$.*
4. *The $\mathbb{C}$-vector space S = span(x^α : $x^\alpha \notin < LT(I) >$) is finite-dimensional.*
5. *The $\mathbb{C}$-vector space $\mathbb{C}[x_1, ..., x_n]/I$ is finite-dimensional.*

The proof of this theorem is non-trivial, however the proof goes beyond the scope of this paper. As we can see, if we have a zero-dimensional ideal, then for each variable x_i, there is a polynomial in the Groebner basis for I with a power of x_i as a leading monomial. Consider the following example of a zero-dimensional ideal.

Example 1.9. Let $I =< xy^3 - x^2, x^3y^2 - y >$ in $\mathbb{R}[x, y]$. Using grlex the Groebner basis is $G = \{x^3y^2 - y, x^4 - y^2, xy^3 - x^2, y^4 - xy\}$ and $\langle LT(I)\rangle = \langle x^3y^2, x^4, xy^3, y^4\rangle$. We can draw a picture in $\mathbb{Z}^2_{\geq 0}$ to represent the exponent vectors of the monomials in $\langle LT(I)\rangle$ and its complement as follows. The vectors

$$\alpha(1) = (3, 2),$$
$$\alpha(2) = (4, 0),$$
$$\alpha(3) = (1, 3),$$
$$\alpha(4) = (0, 4)$$

are the exponent vectors of the generators of $\langle LT(I)\rangle$. Thus, the elements of $((3, 2) + \mathbb{Z}^2_{\geq 0}) \cup ((4, 0) + \mathbb{Z}^2_{\geq 0}) \cup ((1, 3) + \mathbb{Z}^2_{\geq 0}) \cup ((0, 4) + \mathbb{Z}^2_{\geq 0})$ are the exponent vectors of monomials in $\langle LT(I)\rangle$. As a result, we can represent the monomials in $\langle LT(I)\rangle$ by the integer points in the shaded region in $\mathbb{Z}^2_{\geq 0}$ shown below in Figure 1.

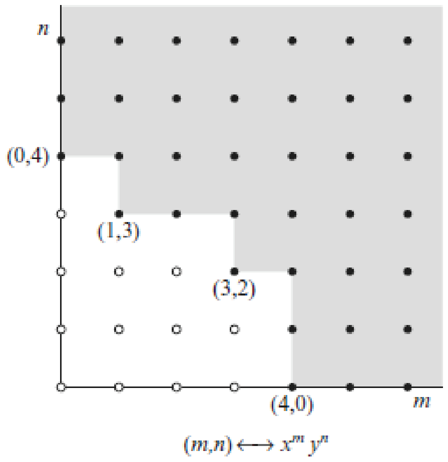

FIGURE 1

2. The FGLM Algorithm

The difficulties that arise from trying to compute reduced Groebner bases with respect to lex ordering can seem insurmountable and enough to render Groebner bases useless. Fortunately there exists a way to circumvent some of the difficulties. The aptly-named FGLM algorithm was developed by J.C. Faugère, P. Gianni, D. Lazard, and T. Mora. As mentioned earlier, it allows one to take a Groebner basis from the relatively easy calculations of a grevlex ordering and convert it to the reduced Groebner basis for the same ideal with respect to another monomial ordering. The drawback to this algorithm is that it only applies to zero-dimensional ideals. This will be explained in a later section. In the algorithm, we will be converting a non-lex Groebner basis to lex Groebner basis, but the FGLM algorithm can be used to convert a Groebner basis from any monomial order to any other monomial order.

First, of course, one must have an initial Groebner basis, G, with respect to an initial monomial order. The algorithm then progresses through three steps for each of the monomials in the ring $k[x_1, \ldots, x_n]$ in increasing lex order. At the beginning of each loop, there will be two sets: G_{lex}, which is initially empty but will become the new Groebner basis for the desired monomial order, and B_{lex}, which is also initially empty but will grow to be the lex monomial basis of the quotient ring $k[x_1, \ldots, x_n]/I$ as a k-vector space.

The FGLM algorithm consists of three main parts.

(1) Main Loop: For this first step the user will take the current input monomial x^α, initially 1, and find the remainder under division by G, denoted $\overline{x^\alpha}^G$. Recall that G is the Groebner basis of the ideal with respect to the original monomial ordering. There are two possible cases for what will happen to the remainder. For the first, if $\overline{x^\alpha}^G$ is *linearly dependent* on the remainders of the other members of B_{lex}, then we have a linear combination such that

$$\overline{x^\alpha}^G - \sum_j c_j \overline{x^{\alpha(j)}}^G = 0$$

where $x^{\alpha(j)} \in B_{lex}$ and $c_j \in k$. This implies that $g = x^\alpha - \sum_j c_j x^{\alpha(j)} \in I$. So we add g to the list G_{lex} as the last element. Because we work through the various x^α in increasing order with respect to the new ordering, a polynomial g that is added to G_{lex} will always have x^α with a coefficient of 1 as its leading term.

In the second case, $\overline{x^\alpha}^G$ is *linearly independent* of the remainders of the items in B_{lex}. In this event, x^α is added to B_{lex}.

If the first case applied and we added a polynomial to G_{lex}, then G_{lex} must be tested to see if it is the desired Groebner basis. To do this, we use the Termination Test, the second part of the FGLM algorithm.

(2) Termination Test: In the event that a new polynomial, g, was added to G_{lex}, the user must compute $LT(g)$. If $LT(g)$ is a power of x_i, where x_i is the greatest variable in the new monomial ordering, then the algorithm terminates. Otherwise, proceed to the third part of the algorithm, the Next Monomial step.

(3) Next Monomial: In this phase of the algorithm, simply replace the x^α that has just been processed with the next monomial with respect to the new order which is not divisible by any of the leading terms of the polynomials in G_{lex}.

The user repeats the steps of this algorithm until the conditions are met for the Termination Test.

Notice that whenever a polynomial g is added to G_{lex}, its leading term is $LT(g) = x^\alpha$ with coefficient 1, hence each basis element must be monic. Also, because the leading term of each basis element is linearly independent of the leading terms of all other elements, the Groebner basis obtained from this algorithm must be reduced. The following example fully demonstrates the algorithm for the reader to clearly see how it works before we present the proof.

3. Example

We will use the FGLM algorithim to find a lexicographic order (with $x > y > z$) Groebner basis for the ideal $I =< x^2 + 2y^2 - y - 2z, x^2 - 8y^2 + 10z - 1, x^2 - 7yz >$ from the graded reverse lexicographic Groebner basis, $G = \{980z^2 - 18y - 201z + 13, 35yz - 4y + 2z - 1, 10y^2 - y - 12z + 1, 5x^2 - 4y + 2z - 1\}$. We start with the least variable in the monomial order, z, and consider it raised to the 0 degree. We then calculate the remainder of z^0 under division by G. We then continue increasing the degree of z and finding remainders under division by G until we find a remainder that is linearly dependent upon the other remainders. This linearly dependent remainder is subtracted from the dividend for which it corresponds and this polynomial is added to the set making up the Groebner basis.

$\overline{z^0}^G = \overline{1}^G = 1$
$\overline{z}^G = z$
$\overline{z^2}^G = \frac{9}{490}y + \frac{201}{980}z - \frac{13}{980}$
$\overline{z^3}^G = \frac{2817}{480200}y + \frac{26653}{960400}z - \frac{2109}{960400}$

Since $\overline{z^3}^G$ is a linear combination of $\overline{1}^G$, $\overline{z}^G$, and $\overline{z^2}^G$,

$$g_1 = z^3 - \frac{313}{980}z^2 + \frac{37}{980}z + \frac{1}{490}$$

is the first polynomial added to G_{lex}.

Now we consider the next variable in the monomial order, y. We again take the remainder with respect to G.

$\overline{y}^G = y$

We find y itself can be expressed as a linear combination, namely $y = \frac{490}{9}\overline{z^2}^G - \frac{67}{6}\overline{z}^G + \frac{13}{18}$ and subsequently

$$g_2 = y - \frac{490}{9}z^2 + \frac{67}{6}z - \frac{13}{18}$$

is added to G_{lex}.

Lastly we consider the greatest variable in the order, x.

$\overline{x}^G = x$
$\overline{x^2}^G = \frac{4}{5}y - \frac{2}{5}z + \frac{1}{5}$

Now $\overline{x^2}^G$ can be expressed in relation to $\overline{y}^G$ and $\overline{z}^G$ and accordingly

$$g_3 = x^2 - \frac{392}{9}z^2 + \frac{28}{3}z - \frac{7}{9}$$

is the final function added to G_{lex}, leaving us with

$$G_{lex} = \{g_1, g_2, g_3\},$$

our desired lex Groebner basis.

Before we show that the FGLM algorithm is a valid method for changing the ordering of a Groebner basis, we will state the following lemma:

Lemma 3.1. *(Dickson's Lemma) Given an infinite list $x^{\alpha(1)}, x^{\alpha(2)}, \ldots$ of monomials in $k[x_1, \ldots, x_n]$, there is an $N \in \mathbb{N}$ such that every $x^{\alpha(i)}$ is divisible by one of $x^{\alpha(1)}, \ldots, x^{\alpha(N)}$.*

Theorem 3.2. *The algorithm described above terminates on every input Groebner basis, G, that generates a zero-dimensional ideal I, and correctly computes a lex Groebner basis, G_{lex}, for I and the lex monomial basis, B_{lex}, for the quotient ring $k[x_1, \ldots, x_n]/I$.*

Proof. We begin with the key observation that monomials are added to the list B_{lex} in strictly increasing lex order. Similarly, if $G_{lex} = \{g_1, \ldots, g_k\}$, then

$$LT(g_1) <_{lex} \ldots <_{lex} LT(g_k),$$

where $>_{lex}$ is the lex order we are using. We also note that when the Main Loop adds a new polynomial g_{k+1} to $G_{lex} = \{g_1, \ldots, g_k\}$, the leading term $LT(g_{k+1})$ is the input monomial in the Main Loop. Since the input monomials are provided by the Next Monomial procedure, it follows that for all k, $LT(g_{k+1})$ is divisible by none of $LT(g_1), \ldots, LT(g_k)$.

We can now prove that the algorithm terminates for all inputs G which generate zero-dimensional ideals. If the algorithm did not terminate for some input G, then the Main Loop would be executed infinitely many times, so one of the two cases in the Main Loop would be chosen infinitely often. If the first alternative were chosen infinitely often, G_{lex} would give an infinite list $LT(g_1), LT(g_2), \ldots$ of monomials. When applied to $LT(g_1), LT(g_2), \ldots$, Dickson's Lemma would contradict the fact that $LT(g_{k+1})$ is divisible by none of $LT(g_1), \ldots, LT(g_k)$. On the other hand, if the second alternative were chosen infinitely often, then B_{lex} would give infinitely many monomials $x^{\alpha(j)}$ whose remainders on division by G were linearly independent in A. This would contradict the assumption that I is zero-dimensional. As a result, the algorithm always terminates if G generates a zero-dimensional ideal I.

Next, suppose that the algorithm terminates with $G_{lex} = \{g_1, \ldots, g_k\}$. By the Termination Test, $LT(g_k) = x_1^{a_1}$, where $x_1 >_{lex} \ldots >_{lex} x_n$. We will prove that G_{lex} is a lex Groebner basis for I by contradiction. Suppose there were some $g \in I$ such that $LT(g)$ is not a multiple of any of the $LT(g_i)$, $i = 1, \ldots, k$. Without loss of generality, we may assume that g is reduced with respect to G_{lex}.

If $LT(g)$ is greater than $LT(g_k) = x_1^{a_1}$, then one easily sees that $LT(g)$ is a multiple of $LT(g_k)$. Hence, this case cannot occur, which means that

$$LT(g_i) < LT(g) \leq LT(g_{i+1})$$

for some $i < k$. But recall that the algorithm places monomials into B_{lex} in strictly increasing order, and the same is true for the $LT(g_i)$. All the non-leading monomials in g must be less than $LT(g)$ in the lex order. They are not divisible by any of $LT(g_j)$ for $j \leq i$, since g is reduced. So, the non-leading monomials that appear in g would have been included in B_{lex} by the time $LT(g)$ was reached by the Next Monomial procedure, and g would have been the next polynomial after g_i included in G_{lex}. This contradicts our assumption on g, which proves that G_{lex} is a lex Groebner basis for I.

To find a monomial basis for $k[x_1, \ldots, x_n]/I$, we need to find all monomials not in $\langle LT(g) \rangle$ for all $g \in G_{lex}$. But B_{lex} contains all such monomials, so B_{lex} forms a monomial basis for the quotient ring as a k-vector space. So B_{lex} consists of all monomials determined by the Groebner basis G_{lex}. □

If I were not a zero dimensional ideal, then some monomial would always yield a linearly independent remainder, and the Main Loop would never terminate. By Moeller and Mora [3] we know the upperbound, call it B, of the total degree of a Groebner basis to be $((n+1)(d+1)+1)^{(n+1)2^{s+1}}$ where n is the number of variables in the ideal, d is the total degree of the ideal, and s is the dimension of the ideal. To assure termination on a positive dimensional ideal, if we know B, then for each monomial x, the main loop would need only to run up to x^B, and terminate if the loop does not do so before x^B. However, because the upper bound of the total degree grows rapidly as the dimension of the ideal increases, it is practical to only use the FGLM algorithm on zero dimensional ideals.

4. Conclusion

Finding Groebner bases with respect to lexicographic order is useful because of the algorithm's ability to eliminate the largest term in at least one of the basis elements which makes solving a system of polynomial equations much easier. Consider a group of polynomial equations $f_1, \ldots, f_s$. These equations determine $I = \langle f_1, ..., f_s \rangle$ and to solve this system, we want to find $\mathbf{V}(I)$. If we compute $\mathbf{V}(I)$ using a lex Groebner basis, then we will get equations $g_1, ..., g_t$ forming the

Groebner basis. Using the property of lexicographic order mentioned above, we can then back substitute to find the solutions to the system. However, finding the lexicographic Groebner basis directly can be computationally expensive. For example, the lexicographic Groebner basis for $I =< x^5 + y^5 + z^5 - 1, x^3 + y^3 + z^2 - 1 >$ contains a polynomial with 415 terms, total degree of 37, with a largest coefficient of 141,592,532,029,352. It is instead more efficient to find a Grobner Basis with respect to the graded reverse lexicographic order, and then use the FGLM algorithm to convert the basis to lexicographic order. The Groebner basis in graded reverse lexicographic order for the above ideal is considerably less pathological: the largest polynomial has 38 terms, total degree of 11, and the largest coefficient is 7. The FGLM algorithm allows us to more efficiently compute lexicographic Groebner bases, which have a wide range of applications.

Acknowledgements

The authors thank Robert Lax for his thought provoking lectures, Jesse Taylor for helpful discussions concerning this paper, and the VIGRE program for our funding.

References

[1] Cox, D., Little, J., and O'Shea, D. (2007). *Ideals, Varieties, and Algorithms* (3^{rd} ed.). New York: Springer.

[2] Cox, D., Little, J., and O'Shea, D. (1998). *Using Algebraic Geometry.* New York: Springer.

[3] Moeller, H. M., and Mora, F. (1984). Upper and lower Bound for the degree of Groebner bases, *Lecture Notes in Computer Science*, 174, 172-183.