A SHORT INTRODUCTION TO GRÖBNER BASIS

CHRISTOPHER DUPRE

Abstract. In this paper we give a brief introduction to and motivation for the concept of Gröbner basis with eyes towards applications in [1]. We will attempt to only use pre-requisites from linear algebra and elementary abstract algebra. All other results will be cited when needed. The primary exception is the ascending chain condition, which we cite without proof. We very closely follow the exposition as in [2].

1. Introduction

When faced with a system of polynomial equations, it is not always easy to explicitly find the solution set. Oftentimes, the exact representation of our polynomial system can make this easier or harder to see. Consider for instance the following system:

$$x^{10} + z^6xy^2 - 6x^5 + xyz^3 - 2 = 0 \tag{1}$$

$$z^6xy^2 + xyz^3 + x^2y + 3xy^2 = 0 \tag{2}$$

$$x^{10} - 6x^5 - 2x^2y - 6xy^2 - 2 = 0. \tag{3}$$

The task of finding all possible roots to this system seems totally hopeless. However, by rearranging the terms

$$\begin{aligned} 2(1) - 2(2) - (3) &= x^{10} - 6x^5 - 2 = 0 \\ -\big((1) + (2) + (3)\big) &= x^2y + 3xy^2 = 0 \\ (3) - (1) + 2(2) &= xyz^3 + z^6xy^2 = 0. \end{aligned}$$

Notice now that the first equation is quadratic in x^5 and can therefore all roots can be found easily. Further, after specifying x the second is a quadratic in y and again can be easily solved. After specifying both x and y, the third equation is a quadratic in z^3 and so again can be solved. Thus the problem which initially seemed hopeless can be solved with nothing more than the quadratic equation and *picking the right polynomial representatives.*

Let us examine this a bit further. A special case of a system of polynomial equations is a system of linear equations, so let us begin by examining this case. Consider the following linear system

$$\begin{aligned} x + 7y + 8z &= 3 \\ 6x + 2y + 6z &= 5 \\ 8y + 6z &= 4. \end{aligned} \tag{4}$$

We may consider the associated augmented matrix to this system to find a solution. Doing so we achieve

$$\left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 6 & 2 & 6 & 5 \\ 0 & 8 & 6 & 4 \end{array}\right]. \tag{5}$$

We want to simplify this expression using some linear combination such that this augmented system is much easier to solve. Ideally we would like one variable to be isolated such that we can obtain its value without any computation and then use this value to find the next and so on. This leads to the ideas of *Gaussian elimination* and *Reduced Row Echelon Form.* The standard algorithm for finding the reduced row echelon

Date: December 2021.

form involves systematically eliminating variables from each column. For our augmented matrix this would be given as

$$\left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 6 & 2 & 6 & 5 \\ 0 & 8 & 6 & 4 \end{array}\right] \xrightarrow{Row2-6(Row1)} \left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 0 & -40 & -42 & -13 \\ 0 & 8 & 6 & 4 \end{array}\right] \xrightarrow{Row2/(-40)} \left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 0 & 1 & 1.05 & 0.325 \\ 0 & 8 & 6 & 3 \end{array}\right] \tag{6}$$

$$\xrightarrow{Row3-8(Row2)} \left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 0 & 1 & 1.05 & 0.325 \\ 0 & 0 & -2.4 & 1.4 \end{array}\right] \xrightarrow{Row3/(-2.4)} \left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 0 & 1 & 1.05 & 0.325 \\ 0 & 0 & 1 & -0.58\overline{3} \end{array}\right]. \tag{7}$$

From the augmented matrix, we can immediately see that $z = -0.58\overline{3}$ and can quickly find x, y from z. Something which is not often mentioned is the fact that solving columns from left to right is not necessary to get a tractable augmented matrix. Consider the following

$$\left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 6 & 2 & 6 & 5 \\ 0 & 8 & 6 & 4 \end{array}\right] \xrightarrow{(Row2)/2} \left[\begin{array}{ccc|c} 1 & 7 & 8 & 3 \\ 3 & 1 & 3 & 2.5 \\ 0 & 8 & 6 & 4 \end{array}\right] \xrightarrow[Row3-8Row(2)]{Row1-7(Row2)} \left[\begin{array}{ccc|c} -20 & 0 & -13 & -14.5 \\ 3 & 1 & 3 & 2.5 \\ -24 & 0 & -18 & -16 \end{array}\right] \tag{8}$$

$$\xrightarrow{(Row3)/(-24)} \left[\begin{array}{ccc|c} -20 & 0 & -13 & -14.5 \\ 3 & 1 & 3 & 2.5 \\ 1 & 0 & 0.75 & 0.\overline{6} \end{array}\right] \xrightarrow[Row2-3(Row3)]{Row1+20(Row3)} \left[\begin{array}{ccc|c} 0 & 0 & 2 & -1.1\overline{6} \\ 0 & 1 & 0.75 & 0.5 \\ 1 & 0 & 0.75 & 0.\overline{6} \end{array}\right] \tag{9}$$

$$\xrightarrow{Row1/2} \left[\begin{array}{ccc|c} 0 & 0 & 1 & -0.58\overline{3} \\ 0 & 1 & 0.75 & 0.5 \\ 1 & 0 & 0.75 & 0.\overline{6} \end{array}\right] \tag{10}$$

where we again get that $z = -0.58\overline{3}$ and we can solve for x, y quite easily from there. The important thing is not the specific order in which we solve along the columns *but that we have an order at all.* This is really trivial in linear systems as any ordering of the variables will work, but it becomes less obvious with multi-nomials. Out of x^3 and xyz, which is bigger? Which should be solved for first?

2. Orderings on Monomials in $k[x_1, x_2, ..., x_n]$

We saw in the last section that in order to extend the ideas of Gaussian elimination and "nice basis" in linear systems to multinomial systems, we needed some form of ordering on the elements so that we could perform a systematic reduction. After deciding on an ordering of our variables, it is natural to identify each monomial term with its exponent as a multi-index.

$$x_1^{\alpha_1} x_2^{\alpha_2} x_3^{\alpha_3} ... x_n^{\alpha_n} \leftrightarrow (\alpha_1, \alpha_2, \alpha_3, ...\alpha_n) \in \mathbb{Z}^{\mathbb{N}}_{\geq 0}.$$

In order to respect our usual notion of order on univariate polynomials, we will require that multiplication by an arbitrary polynomial cannot reverse the order. All of this inspires the following definition:

Definition 2.1. *A **monomial ordering** $>$ on $k[x_1, x_2, ..., x_n]$ is a relation on $\mathbb{Z}^{\mathbb{N}}_{\geq 0}$ satisfying:*

(i) $>$ is a total ordering on $\mathbb{Z}^{\mathbb{N}}_{\geq 0}$
(ii) If $\alpha > \beta$ and $\gamma \in \mathbb{Z}^{\mathbb{N}}_{\geq 0}$, then $\alpha + \gamma > \beta + \gamma$
(iii) $>$ is a well-ordering on $\mathbb{Z}^{\mathbb{N}}_{\geq 0}$

As a reminder, a well-ordering is an order relation in which every non-empty subset has a least element. For our purposes, the following lemma may be easier to visualize

Lemma 1. *An order relation on $\mathbb{Z}^{\mathbb{N}}_{\geq 0}$ is a well-ordering if and only if every strictly decreasing sequence eventually terminates.*

As a few examples of monomial ordering, we introduce the following orderings on $\mathbb{Z}^{\mathbb{N}}_{\geq 0}$.

Definition 2.2. *The **lexicographic order** (denoted $>_{lex}$) on $\mathbb{Z}^{\mathbb{N}}_{\geq 0}$ is defined as $\alpha >_{lex} \beta$ if the first nonzero entry of $\alpha - \beta$ is positive.*

Proposition 2.3. *The lexicographic order is a monomial order*

Proof. The fact that (i) and (ii) are satisfied by lexicographic order is immediate from it's definition. (iii) follows from the well-ordering of $\mathbb{Z}_{\geq 0}$ and the consecutive consideration of each index in the multi-index. □

Definition 2.4. *The* ***graded lexicographic order*** *(denoted $>_{grlex}$) on $\mathbb{Z}_{\geq 0}^{\mathbb{N}}$ is defined as $\alpha >_{grlex} \beta$ if $|a| > |b|$ or $|a| = |b|$ and $\alpha >_{lex} \beta$.*

Definition 2.5. *Let $f = \sum_{i=1}^{n} c_{\alpha(i)} x^{\alpha(i)}$ be a non-zero polynomial and let $>$ be some monomial ordering. We define the following quantities of this polynomial:*

1. *The* ***multidegree*** *of f (denoted multideg(f)) is equal to the largest power with respect to the given order i.e.*

$$multideg(f) = \max_{>}\{\alpha(i) | c_{\alpha(i)} \neq 0\}.$$

2. *The* ***leading coefficient*** *of f (denoted LC(f)) is the coefficient of the leading term i.e.*

$$LC(f) = c_{multideg(f)}.$$

3. *The* ***leading monomial*** *of f (denoted $LM(f)$ is the monomial in the leading term i.e.*

$$LM(f) = x^{multideg(f)}.$$

4. *The* ***leading term*** *of f (denoted LT(f)) is the product of the leading coefficient and the leading monomial i.e.*

$$LT(f) = LC(f)LM(f)$$

Lemma 2. *Let $f, g \in k[x_1, x_2, ..., x_n]$ be non-zero polynomials. Then:*

1. *multideg(fg) = multideg(f)multideg(g)*
2. *If $f + g \neq 0$, multideg(f + g) $\leq$ max(multideg(f),multideg(g)). If multideg(f)≠multideg(g), then multideg(f + g) = max(multideg(f),multideg(g)).*

3. A Division Algorithim and Limitations

Let us try to use this notion of order to extend our division algorithm in $k[x_1]$ to $k[x_1, x_2, ..., x_n]$. The idea is fairly simple, but is easier to see than to explain directly. Let us see an example. We will attempt to find a representation of $y^3 - x^2$ as $q_1(x, y)(x^2 + y) + q_2(x^2 - 1) + r$. We fix the lexicographic order on the monomials. We will also fix an order of division where we divide by $x^2 + y$ and then $x^2 - 1$. Now, looking at the leading order terms we have that $LT(y^3 - x^2) = -x^2$ and $LT(x^2 + y) = x^2$. These terms divide one another, so we can find their least common multiple and generate a new polynomial $p_1 = \frac{x^2}{-x^2}(y^3 - x^2) - \frac{x^2}{x^2}(x^2 + y) = -y^3 + y$. We now repeat the division using this as our polynomial. Looking at leading order terms we have that $LT(-y^3 + y) = -y^3$ and $LT(x^2 + y) = x^2$ which do not divide each other. Thus we continue to $x^2 + 1$ and note that $LT(x^2 + 1)$ which does not divide $-y^3$. We have now determined that $-y^3$ can no longer be divided and may pass it to the remainder. Repeating this with y shows that we can also move it to the remainder. We thus have completed our division algorithm and have a representative

$$y^3 - x^2 = (x^2 + y) + 0(x^2 - 1) + y^3 - y.$$

Repeating this in the other order gives us that

$$y^3 - x^2 = 0(x^2 + y) + (x^2 - 1) + y^3 - 1.$$

In pseudo-code our procedure is as follows

Algorithm 1 Polynomial division

Input: A polynomial dividend f and an ordered set of polynomial divisors $G = \{g_1, g_2, ..., g_n\}$
Output: A list of q_i and an r such that $f = \sum_{i=1}^{n} q_i g_i + r$ and no term of r is divisible by $LT(g_i)$ for any i.
$p = f,\ q_i = 0,\ r = 0$
while $p \neq 0$ **do**
    if $LT(p)$ is divisible by $LT(g_i)$ **then**
        $q_i = q_i + \frac{LT(p)}{LT(g_i)},\ p = p - \frac{LT(p)}{LT(g_i)} g_i$, return to top of loop
    else if i = n **then**
        $r = r + LT(p),\ p = p - LT(p)$
    else
        $i = i + 1$
    end if
end while
return q_i and r

This leads to a relatively big problem. The remainder term depends on the order of division. In fact, the situation is worse than it would initially appear. Consider the division of $xy+1$ by x^2+y, x^3-1. According to the standard division algorithm with lexicographic order, we would conclude that

$$xy + 1 = 0(x^2 + y) + 0(x^3 - 1) + xy + 1.$$

However we can see by direct calculation that

$$xy + 1 = x(x^2 + y) - (x^3 - 1) + 0.$$

Thus the division algorithm applied to f_1 by $f_2, f_3, ..., f_n$ giving 0 is a sufficient condition for $f_1 \in I(f_2, f_3, ..., f_n)$ *but not necessary.*

If we wish to truly generalize monomial division, we need to improve our method. Notice the terms which we missed were "lower order" terms in our ordering. We did not have enough polynomials to capture these terms. If we could expand our list of polynomials to include these missing lower order terms, we would have a chance at a successful division algorithm.

4. Gröbner Basis

Up until now we have been focusing on the order of monomials, but a generic ideal is very rarely specified in terms of monomials. However, once we fix a monomial order, every polynomial in our ideal has a unique leading term. We may then consider the ideal generated by these terms.

Definition 4.1. *Let $I \subset k[x_1, x_2, ..., x_n]$ different from $\{0\}$ and fix a monomial ordering on $k[x_1, x_2, ..., x_n]$. Then*

(1) $LT(I) = \{LT(f) | f \in I \backslash \{0\}\}$
(2) We denote by $\langle LT(I) \rangle$ the ideal generated by $LT(I)$.

Definition 4.2. *Fix a monomial ordering on $k[x_1, x_2, ..., x_n]$. A finite subset $g_1, g_2, ..., g_n$ of an ideal $I \subset k[x_1, x_2, ..., x_n]$ different from $\{0\}$ is said to be a **Gröbner basis** (or a **standard basis**) if*

$$\langle LT(I) \rangle = \langle LT(g_1), LT(g_2), ..., LT(g_n) \rangle$$

Definition 4.3. *Given an ordered set of polynomials $G = \{g_1, g_2, ..., g_n\}$, a monomial ordering, and a polynomial f we denote the remainder under the division algorithm of f by G as $\overline{f}^G$.*

Theorem 4.4. *Every ideal admits a Gröbner basis. Further, every Gröbner basis is a basis of the ideal.*

Proof. Note that the set of multi-indices in finitely many variable is countable, and thus the set $A = \{\gamma | x^\gamma \in LM(I)\}$ is countable. Let γ_n be an enumeration of A. Then we have the ascending chain of ideals

$$\langle x^{\gamma_1} \rangle \subset \langle x^{\gamma_1}, x^{\gamma_2} \rangle \subset \langle x^{\gamma_1}, x^{\gamma_2}, x^{\gamma_3} \rangle \subset ....$$

By the ascending chain condition, the chain eventually stabilizes. As the elements of $LT(I)$ and $LM(I)$ differ only by a constant, $LM(I) \subset I(LT(I))$ and $LT(I) \subset I(LM(I))$ and thus $I(LM(I)) = I(LT(I))$. Thus

there exists a finite number of monomials such that $\langle LT(I)\rangle = \langle x^{\gamma_1}, x^{\gamma_2}, ..., x^{\gamma_n}.\rangle$. Further, as $x^{\gamma_i} \in LM(I)$, there must be a $g_i \in I$ such that $LM(g_i) = x^{\gamma_i}$. Then the collection $G = \{\frac{g_i}{LC(g_i)}\}_{i=1}^n$ is a Gröbner basis. We claim the G is also a basis of I. Let $f \in I$, and consider $\overline{f}^G$. We claim that $\overline{f}^G = 0$. Suppose not. Then as $\overline{f}^G \in I$, $LT(\overline{f}^G) \in \langle LT(I)\rangle = \langle LT(g_1), LT(g_2), ...LT(g_n)\rangle$. Thus $\overline{f}^G$ cannot possibly be the remainder as we may still divide by at least one more of the polynomials. □

Proposition 4.5. *Let $I \subset k[x_1, x_2, ..., x_n]$ be an ideal and let $G = \{g_1, g_2, ..., g_m\}$ be a Gröbner basis for I. Then given any $f \in k[x_1, x_2, ..., x_n]$ there exists a unique remainder $r \in k[x_1, x_2, ..., x_n]$ such that*

(1) No term of r is divisible by any of $LT(g_1), ..., LT(g_m)$.
(2) There is a $g \in I$ such that $f = g + r$.

In particular, the remainder is independent of the order of the elements of G for the division algorithm and we may write $\overline{f}^G$ without specifying an ordering of G if G is a Gröbner basis.

Proof. The existence of r follows from the division algorithm. To see the uniqueness, consider not. Then $f = g + r = g' + r'$ where $g, g' \in I$. Then $r - r' = g' - g \in I$. If $r - r' \neq 0$, then $LT(r - r') \in LT(I)$. Thus $LT(r - r')$ is a multiple of some $LT(g_i)$, but this contradicts the property of the remainder as $r - r'$ contains only terms in r, r'. Thus the remainder is unique and therefore clearly independent of the order of division. □

Definition 4.6. *Given two polynomials g_1, g_2 and fix a monomial order. Then the **S-poylnomial** (denoted $S(g_1, g_2)$) is defined as*

$$S(g_1, g_2) = \frac{lcm(LM(g_1), LM(g_2))}{LT(g_1)} g_1 - \frac{lcm(LM(g_1), LM(g_2))}{LT(g_2)} g_2$$

Theorem 4.7 (Buchberger's Criterion)**.** *A basis of polynomials $G = \{g_1, g_2, ..., g_m\}$ is a Gröbner basis for $I(g_1, g_2, ..., g_m)$ if and only if $\overline{S(g_i, g_j)}^G = 0$ for every pair of indices i, j*

Proof. If $g_1, g_2, ..., g_m$ is a Gröbner basis, then $\overline{S(g_i, g_j)}^G = 0$ for every pair of indices i, j by the division algorithm.

If $\overline{S(g_i, g_j)}^G = 0$ for every pair of indices i, j, we will search for "the most efficient representation" of $f \in k[x_1, x_2, ..., x_n]$. As $g_1, g_2, ..., g_m$ is a basis for I, for every nonzero $f \in I$ we have a representation $f = \sum_{j=1}^m h_j g_j$. Now consider $M = \max\{\text{multideg}(h_j g_j) | h_j g_j \neq 0\}$ where max is taken with respect to our monomial ordering. Note that as it is the max over a finite number of elements, there is at least one i such that $M = \text{multideg}(h_i g_i) = \text{multideg}(h_i) + \text{multideg}(g_i)$. Note that $\text{multideg}(f) \leq M$. If there exists a representation such that $\text{multideg}(f) = M$, then $LM(f) = LM(h_i)LM(g_i)$ and thus $LM(f) \in \langle LT(g_1), LT(g_2), ..., LT(g_m)\rangle$. If this is true for every $f \in I$, then $g_1, g_2, ..., g_m$ is a Gröbner basis by definition. Assume that there exists an $f \in I$ such that for all representations $\sum_{j=1}^m h_j g_j$ we have that $\max\{\text{multideg}(h_j g_j) | h_j g_j \neq 0\} \geq \delta > \text{multideg}(f)$. Let $\sum_{i=1}^m h_i g_i$ be a representation which achieves $M = \delta$. Then consider the decomposition

$$\sum_{j=1}^m h_j g_j = \sum_{j|\text{multideg}(h_j, g_j)=\delta} LT(h_j) g_j + \sum_{j|\text{multideg}(h_j, g_j)=\delta} (h_j - LT(h_j)) g_j + \sum_{j|\text{multideg}(h_j g_j) h_j g_j < \delta} .$$

Clearly the last two terms have multi-degree less than δ, thus it will suffice to show that we can express the first term in terms of lower order terms. Note that as $(f) < \delta$, the multi-degree of the first sum is less than δ, even though each individual term has multi-degree δ. We will need the following lemma

Lemma 3. *If $\{p_i\}_{i=1}^m \in k[x_1, x_2, ..., x_n]$ satisfy $multideg(p_i) = \delta$ for all i, $multideg\left(\sum_{i=1}^m p_i\right) < \delta$ then $\sum_{i=1}^m p_i = \sum_{i=1}^m h_i S(p_i, p_j)$ where $h_i \in k$ for all j.*

Proof. Note that all p_i have the same leading order monomial. Thus if $LT(p_i) = d_i x^\delta$ then we have that

$$S(p_i, p_j) = \frac{1}{d_i} p_i - \frac{1}{d_j} p_j.$$

As multideg $\left(\sum_{i=1}^m p_i\right) < \delta$, we must have that $\sum_{i=1}^m d_i = 0$. Therefore, we have that

$$\sum_{i=1}^m d_i S(p_i, p_j) = \sum_{i=1}^m p_i + \frac{1}{d_j} p_j \left(\sum_{i=1}^m d_i\right) = \sum_{i=1}^m p_i. \tag{11}$$

□

Using the lemma, we know that

$$\begin{aligned}\sum_{j|\text{multideg}(h_j,g_j)=\delta} LT(h_j)g_j &= \sum_{j|\text{multideg}(h_j,g_j)=\delta} k_j S(LT(h_j)g_j, LT(h_1)g_1) \\ &= \sum_{j|\text{multideg}(h_j,g_j)=\delta} k_j \tfrac{x^\delta}{\text{lcm}(LM(g_1),LM(g_j))} S(g_j, g_1).\end{aligned}$$

As $\overline{S(g_j,g_1)}^G$ for all g_j, there exists a representation $\sum_{j|\text{multideg}(h_j,g_j)=\delta} w_j g_j$ such that

$$\text{multideg}\left(\sum_{j=1}^m h_j g_j - \sum_{j|\text{multideg}(h_j,g_j)=\delta} w_j g_j\right) < \delta \tag{12}$$

$$\sum_{j|\text{multideg}(h_j,g_j)=\delta} w_j g_j = \sum_{j|\text{multideg}(h_j,g_j)=\delta} LT(h_j)g_j = 0. \tag{13}$$

Thus (12) is again a representation of f which has multidegree smaller than δ. This contradicts the defintion of δ, so all $f \in I$ must admit a representation $\sum_{i=1}^m h_i g_i$ such that multideg $\left(\sum_{i=1}^m h_i g_i\right) = \text{multideg}(f)$. Thus $G = \{g_1, g_2, ..., g_n\}$ is a Gröbner basis. □

5. Buchberger's Algorithim

We now have an effective criterion to determine whether or not we currently have a Gröbner basis or not. However, we do not yet have a constructive method for finding a Gröbner basis from an arbitrary basis of an ideal. The basic idea of our algorithm will be to satisfy $\overline{S(g_i,g_j)}^G = 0$ for all i, j by attempting the division the algorithm on $S(g_i, g_j)$ and adding the remainder to G. This then gives a new set of polynomials G' which guarantees that $\overline{S(g_i,g_j)}^{G'} = 0$. We can then iterate this procedure to achieve a Gröbner basis. In pseudo-code this is expressed as

Algorithm 2 Buchberger's Algorithm

```
Input: A basis g_1, g_2, ..., g_n for an ideal I
Output: A Gröbner basis for I
G = {g_1, g_2, ..., g_n}
repeat
    G' = G
    for g_i, g_j ∈ G' do
        if S(g_i, g_j)^{G'} ≠ 0 then
            G = G ∪ S(g_i, g_j)^{G'}
        end if
    end for
until G' = G
```

Note that if the algorithm terminates, then it clearly returns a Gröbner basis for I as $I(G) = I(G')$ at every step and the returned set satisfies Buchberger's Criterion. The only thing left to prove is that it terminates. This follows from the fact that $\langle LT(G')\rangle \subset \langle LT(G)\rangle$ at every step of the iteration. Thus the ideals generated by the leading order terms form an ascending chain, which must eventually terminate by the ascending chain condition.

This is a totally correct but highly inefficient algorithm, and many methods can be used to improve the efficiency of this procedure (see Chapter 9 and 10 of [2] and [3]). Beyond the run time of the algorithm,

notice that many terms in our output Gröbner basis will likely be unnecessary. We want an effective criterion to determine if a term is absolutely necessary. In fact, we have that

Lemma 4. *Let G be a Gröbner basis of I. If $p \in G$ is such that $LT(p) \in \left\langle \{LT(g) \mid g \in G\backslash p\} \right\rangle$. Then $G\backslash p$ is also a Gröbner basis of I.*

Proof. Clearly $\left\langle \{LT(g) \mid g \in G\backslash p\} \right\rangle \subset \langle LT(G) \rangle$. If $LT(p) \in \left\langle \{LT(g) \mid g \in G\backslash p\} \right\rangle$, then $\{LT(G)\} \subset \left\langle \{LT(g) \mid g \in G\backslash p\} \right\rangle$ and therefore by considering the ideal generated by each side we have that $\langle LT(G) \rangle \subset \left\langle \{LT(g) \mid g \in G\backslash p\} \right\rangle$. Thus $\langle LT(G) \rangle = \left\langle \{LT(g) \mid g \in G\backslash p\} \right\rangle = \langle LT(I) \rangle$ and thus $G\backslash p$ is a Gröbner basis. □

We say that a Gröbner basis is **minimal** if no element p can be removed from G such that $G\backslash\{p\}$ is still a Gröbner basis. Intuitively, you can imagine that this is a reduction to upper triangular form. We should not expect minimal basis to be unique, however this is unique up to a leading coefficient.

Definition 5.1. *A **reduced Gröbner basis** for a polynomial ideal I is a Gröbner basis G for I such that*

(1) $LC(g) = 1$ for all $g \in G$

(2) For all $p \in G$, no monomial component of p lies in $\left\langle \{LT(g) \mid g \in G\backslash p\} \right\rangle$.

Theorem 5.2. *Let $I = I(g_1, g_2, ..., g_n) \neq \{0\}$ be a polynomial ideal. For a given monomial ordering, I has a reduced Gröbner basis and it is unique.*

Proof. By Buchberger's Algorithm we know that we can always construct a Gröbner basis from $g_1, g_2, ..., g_n$. We can also always construct a minimal basis from our Gröbner basis G as Lemma 4 is easily checkable. Let $G' = \{f_1, f_2, ..., f_m\}$ be a minimal basis for I. Let $\tilde{f}_i = \overline{f_i}^{G\backslash f_i}$ and let $G'' = \{\frac{\tilde{f}_1}{LC(\tilde{f}_1)}, \frac{\tilde{f}_2}{LC(\tilde{f}_2)}, ..., \frac{\tilde{f}_m}{LC(\tilde{f}_m)}\}$. We claim that G'' is a reduced Gröbner basis. As G' is minimal, $LM(f_i) = LM(\tilde{f}_i)$ and thus $\left\langle LT(\tilde{f}_i) \right\rangle = \langle LT(f_i) \rangle = I$ and thus G'' is a Gröbner basis. It also clearly satisfies $LC(\tilde{f}_i) = 1$ by construction. As it is a remainder, no term of $\tilde{f}_i$ may be divisible by the $\{LT(f_i)\}_{i \neq j} = \{LT(\tilde{f}_i)\}_{i \neq j}$ by Proposition 4.5. Thus G'' is a reduced Gröbner basis.

To see it is unique, consider not. Then there exists two distinct Gröbner bases $G_1 = \{g_1, g_2, ..., g_n\}, G_2 = \{g'_1, g'_2, ..., g'_m\}$ which are both reduced. Fix an $1 \leq i \leq n$. Then $LT(g_i)$ divides $LT(g'_l)$ for some $1 \leq l \leq m$ definition of Gröbner bases and $LT(g'_l)$ divides $LT(g_j)$ for some $1 \leq j \leq n$. Thus $LT(g_i)$ divides $LT(g_j)$. As G_1 is reduced, $i = j$ and $LT(g'_l) = LT(g_i)$. Thus for every element g_i of G_1 there is an element g'_l of G_2 such that $LT(g_i) = LT(g'_l)$. As this must also be true from G_2 to G_1 and both are reduced, G_1 and G_2 have the same number of elements and there is a bi-jection $\phi \in S_n$ such that $LT(g_i) = LT(g'_{\phi(i)})$. Now consider $g_i - g'_{\phi(i)} \in I$. As G_1 is a Gröbner basis $\overline{g_i - g'_{\phi(i)}}^{G_1} = 0$. However, $LT(g_i) = LT(g'_{\phi_i})$ and no other term is divisible by $LT(G_1) = LT(G_2)$ as both are reduced. Thus $\overline{g_i - g'_{\phi(i)}}^{G_1} = g_i - g'_{\phi(i)} = 0$ and therefore $G_1 = G_2$. □

The procedure for generating the reduced Gröbner basis from a given Gröbner basis is often called automatic reduction and is a generalization of reduction to Reduced Row Echelon form. To see this, note that our system in (4) is given by the ideal

$$I = \langle x + 7y + 8z - 3, 6x + 2y + 6z - 5, 8y + 6z - 4 \rangle$$

computing the reduced Gröbner basis with $x > y > z$ gives

$$G = \{7 + 12z, -15 + 16y, -53 + 48x\}$$

where we can again see that $z = -0.58\overline{3}$. In pseudo-code the algorithm for automatic reduction is given by:

Algorithm 3 Automatic Reduction

Input: A Gröbner basis G for an ideal I
Output: The reduced Gröbner basis for I
$G' = \emptyset$
while $G \neq \emptyset$ **do**
    g_{min} = minimum polynomial in G according to the given monomial ordering
    Append g_{min} to G'
    $R = \{g \in G \mid LM(g) \text{ is divisible by } LM(g_{min})\}$
    $G = G \backslash R$
end while
$G'' = \emptyset$
for $g \in G'$ **do**
    Append $\overline{g}^{G \backslash g}$ to G''
end for
Return G''

This is a true generalization of reduced row echelon form from linear algebra in that the reduced Gröbner basis of a linear system given the ordering determined by their placement in the matrix is exactly the Reduced Row Echelon form.

6. First Applications: Ideal Membership/ Ideal Equality Problem

From Proposition 4.5 it is clear that

$$f \in I(g_1, g_2, ...g_n) \iff \overline{f}^G = 0$$

where G is a Gröbner basis of $I(g_1, g_2, ...g_n)$. From Theorem 5.2 we have that two ideals are equal if and only if there reduced Gröbner bases are identical. In particular, we have that $1 \in G$ where G is the reduced Gröbner basis of I if and only if $G = k[x_1, x_2, ..., x_n]$. By the Algebraic-Geometric Correspondence (see Chapter 4 of [2]), this implies that the system of polynomial equations has no solutions. We therefore have an effective criterion to determine if a system of polynomial equations has a solution and an algorithmic way of finding such solutions.

7. Sudoku Boards, n Colorings and Solving Systems of Polynomial Equations

Let us see this in action. We will be following the exposition in [4]. Consider the following Sudoku board.

5		7		4				3
			8	7	6	9	5	4
6	9	4		1	3	7	8	
9	5		7	2	4	6		1
7	2	1	3				9	
	4			8		2	7	5
			4					6
8			6		1	5		
4		5					1	9

For those unfamiliar with the rules of Sudoku, here is a list of the rules:

(1) Each small square in the puzzle needs to be filled with an integer $1 \leq n \leq 9$
(2) Each row of the puzzle needs to have exactly one of each integer $1 \leq n \leq 9$
(3) Each column of the puzzle needs to have exactly one of each integer $1 \leq n \leq 9$

(4) Each 3x3 sub-square surrounded by thick lines of the puzzle needs to have exactly one of each integer $1 \leq n \leq 9$.

Any filling of the puzzle which satisfies these requirements and contains all of the pre-filled squares is considered a solution to the Sudoku puzzle. We would like to turn the rules above into a system of polynomial equations such that we can find a solution. Let us identify the integers i $1 \leq i \leq 9$ with 9-th roots of unity via the correspondence $i \leftrightarrow z_9^i$ where z_9 is the primitive 9-th root of unity. Then if $w_{i,j}$ is the entry at the square in the i-th row and j-th column, then this must satisfy

$$w_{i,j}^9 - 1 = 0$$

in order to satisfy rule number (1) of Sudoku. In order to satisfy rule number (2) fix a row index index i and note that

$$w_{i,j}^9 - w_{i,l}^9 = (w_{i,j} - w_{i,l}) \left(\sum_{k=0}^{8} w_{i,j}^k w_{i,l}^{8-k} \right) = 0.$$

Thus if we wish to demand that $w_{i,j} \neq w_{i,l}$, we can require that

$$\sum_{k=0}^{8} w_{i,j}^k w_{i,l}^{8-k} = 0.$$

This may initially seem only necessary and not sufficient, but combining the above with $w_{i,j}$ is 9-th root of unity. We have that

$$w_{i,j} = w_{i,l} \implies \sum_{k=0}^{8} w_{i,j}^k w_{i,l}^{8-k} = 8 w_{i,j}^8 \neq 0.$$

Thus this condition really is equivalent to requiring that the two values are distinct. We can then create similar equations for the conditions (3) and (4). This gives a list of polynomial equations which all need to be simultaneously satisfied. If we plug in the known values and compute the reduced Gröbner basis using a Computer Algebra Software, we can determine the unique solution of the above puzzle and if there are multiple possible fillings of the puzzle.

Notice that the only important aspects we used in this construction were that the Sudoku squares had a limited number of values and that certain "neighbors" could not be given the same value. Recall that a graph G is a pair of a set of vertices V and an adjacency matrix E. If we look for an n-coloring of the graph we are looking for an assignment of vertices to a set of n colors c_i such that if $v_i, v_j \in c_m$ then $E_{i,j} = 0$. We can identify each color with an $n - th$ primitive root of unity and follow the construction as in Sudoku but with the equations

$$1 \leq i \leq |V|, a_i^n - 1 = 0, E_{i,j} \neq 0 \implies \sum_{k=0}^{n-1} a_i^k a_j^{n-1-k} = 0.$$

Computing a Gröbner basis for the above system of equations then gives a procedural way to determine if a graph is n-colorable or not. For further information and examples, see [5].

References

[1] Jacob Bedrossian and Sam Punshon-Smith. Chaos in stochastic 2d galerkin-navier-stokes. *arXiv preprint arXiv:2106.13748*, 2021.

[2] David Cox, John Little, and Donal OShea. *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media, 2013.

[3] Massimo Caboara and John Perry. Reducing the size and number of linear programs in a dynamic gröbner basis algorithm. *Applicable Algebra in Engineering, Communication and Computing*, 25(1-2):99–117, 2014.

[4] Elizabeth Arnold, Stephen Lucas, and Laura Taalman. Gröbner basis representations of sudoku. *The College Mathematics Journal*, 41(2):101–112, 2010.

[5] Kathleen Iwancio and Michael Singer. Applications of groebner bases.