Gröbner Bases II

Nicholas Tomlin

May 2, 2019

1 Background

This is a continuation of Tracy Chin's presentation on Gröbner bases, which provides motivation for the construction and develops some initial results. We'll focus here on computational aspects of Gröbner bases, such as how to compute them and their applications to computer science. Recall that our setting is $R = k[x_1, \ldots, x_n]$ for k a field, with $I \subseteq R$ an ideal. Further recall that $\text{LT}(f)$ denotes the leading term of a polynomial $f \in R$, which is defined with respect to a monomial ordering $>$.

2 Preliminaries

We'll begin with some notation, motivated by the division algorithm in multivariate polynomial rings.

Definition 2.1. *Given a family of polynomials $f_1, \ldots, f_s \in R$, then for every $f \in R$, there exist $q_1, \ldots, q_s, r \in R$ by the division algorithm such that:*

$$f = f_1 q_1 + \ldots + f_s q_s + r$$

where no term of r is divisible by $\text{LT}(f_1), \ldots, \text{LT}(f_s)$. We call $r := f^F$ the remainder where $F = \{f_1, \ldots, f_s\}$.

We'll also recall the following theorems from the previous proposition.

Theorem 2.2. *Given $I \subseteq R$ an ideal, then:*

- *There exists a Gröbner basis $G = \{g_1, \ldots, g_m\}$.*
- *The ideal is generated as follows: $I = (g_1, \ldots, g_m)$.*
- *Given $g \in R$, then $g \in I$ iff $g^G = 0$.*

Finally, we'll define the notion of an *S-Polynomial*, which is intended to cancel the leading terms of its constituent polynomials.

Definition 2.3. *Given two polynomials $P_1, P_2 \in R$, we define the S-Polynomial:*

$$S(P_1, P_2) = \frac{lcm(\text{LT}(P_1), \text{LT}(P_2))}{\text{LT}(P_1)} \cdot P_1 - \frac{lcm(\text{LT}(P_1), \text{LT}(P_2))}{\text{LT}(P_2)} \cdot P_2$$

Example 2.4. *To illustrate, we'll compute the S-Polynomial of the following polynomials with lexicographic ordering $x > y$:*

- $P_1 = x^4y - x^2y + x$
- $P_2 = x^2y^2 - y^2$

where $\text{LT}(P_1) = x^4y$ and $\text{LT}(P_2) = x^2y^2$. Therefore, $lcm(\text{LT}(P_1), \text{LT}(P_2)) = x^4y^2$, so computation proceeds as follows:

$$\begin{aligned} S(P_1, P_2) &= \frac{x^4y^2}{x^4y} \cdot \left(x^4y - x^2y + x\right) - \frac{x^4y^2}{x^2y^2} \cdot \left(x^2y^2 - y^2\right) \\ &= (y) \cdot \left(x^4y - x^2y + x\right) - \left(x^2\right) \cdot \left(x^2y^2 - y^2\right) \\ &= \left(x^4y^2 - x^2y^2 + xy\right) - \left(x^4y^2 - x^2y^2\right) \\ &= xy \end{aligned}$$

which cancels the leading terms of both polynomials as desired.

3 Computing Gröbner Bases

Now that we've established some preliminary notions, we will explain how to calculate them with Buchberger's algorithm. But first, we'll present the following theorem without proof:

Theorem 3.1. *Given a set of polynomials $F = \{f_1, \ldots, f_s\} \subset R$ with ideal $I = (f_1, \ldots, f_s)$, then F is a Gröbner basis iff for any $1 \leq i \neq j \leq s$, then the remainder $S(f_i, f_j)^F = 0$.*

We'll now describe *Buchberger's algorithm.* Begin with $F = \{f_1, \ldots, f_s\} \subset R$ family of nonzero polynomials and $I = (f_1, \ldots, f_s)$ generated ideal. Then:

- List pairs $R := \{\{f_i, f_j\} \mid f_i \neq f_j\}$
- While nonempty $R \neq \{\}$:

- Choose arbitrary $\{P_1, P_2\} \in R$
- Remove pair $R := R \setminus \{P_1, P_2\}$
- Calculate remainder $r := S(P_1, P_2)^F$
- If nonzero remainder $r \neq 0$:
  - Add remainder to set $F := F \cup \{r\}$
  - Add corresponding pairs $R := \{\{f, r\} \mid f \in F\}$

- F is a Gröbner basis.

This algorithm does not necessarily produce the smallest Gröbner basis and there are more efficient computational alternatives. However, we will briefly show that it works.

Theorem 3.2. *Buchberger's algorithm produces a Gröbner basis.*

Proof. It suffices to show that Buchberger's algorithm terminates and that the resultant F is a Gröbner basis. The latter claim follows from Theorem 3.1, so we will focus on the algorithm's termination. Indeed, for each loop, let us notate the family of polynomials as F_i such that:

$$F \subsetneq F_1 \subsetneq F_2 \subsetneq F_3 \subsetneq \cdots$$

is an ascending chain. But because we defined the remainder r such that it is not divisible by $\text{LT}(f_1), \ldots, \text{LT}(f_s)$, then we can write another ascending chain as follows:

$$\text{LT}(F) \subsetneq \text{LT}(F_1) \subsetneq \text{LT}(F_2) \subsetneq \text{LT}(F_3) \subsetneq \cdots$$

where $\text{LT}(F_i)$ denotes the ideal generated by the leading terms of polynomials in F_i. But because these ideals exist in R which is Noetherian, then this sequence must stabilize, meaning that the algorithm terminates. □

4 Application to Integer Programming

Now that we've seen how to calculate Gröbner bases, we'll show an application to computer science. Consider the problem of optimizing a function of multiple variables subject to various inequalities and a constraint on the integrality of these variables. For example:

$$\begin{aligned} \text{maximize} \quad & 11x_1 + 15x_2 \\ \text{subject to} \quad & 4x_1 + 5x_2 \leq 37 \\ & 2x_1 + 3x_2 \leq 20 \\ & x_1, x_2 \in \mathbb{Z}^+ \cup \{0\} \end{aligned}$$

We'll give a high-level sketch of how Gröbner bases may be used to solve such problems without proof. This divides into two problems: (1) finding feasible solutions to satisfy the inequalities and (2) finding the maximal such solution in terms of the objective function.

4.1 Feasible Solutions

Continuing with the above example, we'll begin by adding some "slack" variables which allow us to convert our inequalities to equalities:

$$\begin{array}{rll} \text{maximize} & 11x_1 + 15x_2 + 0x_3 + 0x_4 & \\ \text{subject to} & 4x_1 + 5x_2 + x_3 & = 37 \\ & 2x_1 + 3x_2 + x_4 & = 20 \\ & x_1, x_2, x_3, x_4 & \geq 0 \end{array}$$

Next, we'll define a ring homomorphism ϕ according to these variables:

$$\begin{aligned} \phi : k[w_1, w_2, w_3, w_4] &\to k[z_1, z_2] \\ w_1 &\mapsto z_1^4 z_2^2 \\ w_2 &\mapsto z_1^5 z_2^4 \\ w_3 &\mapsto z_1 \\ w_4 &\mapsto z_2 \end{aligned}$$

and we'll claim that values $(x_1, x_2, x_3, x_4) = (A, B, C, D)$ are in the feasible region defined by our inequalities iff $\phi(w_1^A w_2^B w_3^C w_4^D) = z_1^{37} z_2^{20}$. Indeed, expanding this equation as defined above gives:

$$\phi(w_1^A w_2^B w_3^C w_4^D) = z_1^{4A+5B+C} z_2^{2A+3B+D}$$

which holds iff $4A+5B+C = 37$ and $2A+3B+D = 20$ as desired. Therefore, any solution to our system of equations may be expressed as a monomial $w_1^A w_2^B w_3^C w_4^D \in k[w_1, w_2, w_3, w_4]$. Note that this requires $x_1, x_2 \geq 0$, and Laurent polynomials may be used to extend this method to all integers. [2]

4.2 Optimal Solutions

We'll use the algorithm of Conti and Traverso [1] to find an optimal solution. Consider the following ideal:

$$I = \left(z_1^4 z_2^2 - w_1, z_1^5 z_2^3 - w_2, z_1 - w_3, z_2 - w_4\right) \subseteq k[z_1, z_2, w_1, w_2, w_3, w_4]$$

Calculate the Gröbner basis of I and call it G. Then take $f = z_1^{37} z_2^{20}$ and $g = f^G$. By a result in [3], g is a monomial of form $w_1^A w_2^B w_3^C w_4^D$ and therefore denotes a feasible solution. This remainder corresponds to an optimal solution (A, B, C, D) if the following conditions are met.

Definition 4.1. *Given an objective function in $k[w_1, \ldots, w_k]$ and values $A = (A_1, \ldots, A_k)$ and $B = (B_1, \ldots, B_k)$, then define $l(A) > l(B)$ iff the objective function returns a higher value for A than for B.*

Definition 4.2. *The adapted monomial ordering on $k[z_1, \ldots, w_1, \ldots, w_k]$ is defined according to the following rules:*

- *Monomials containing z_i are greater than those containing only w_i*
- *If there exists $A = (A_1, \ldots, A_k)$ and $B = (B_1, \ldots, B_k)$ with*

$$\phi(A_1, \ldots, A_k) = \phi(B_1, \ldots, B_k)$$

and objective valuations $l(A) > l(B)$, then $w_1^{A_1} \cdots w_k^{A_k} > w_1^{B_1} \cdots w_k^{B_k}$.

Theorem 4.3 (Conti and Traverso)**.** *The remainder $g = f^G$ corresponds to an optimal solution if $>$ is an adapted monomial ordering.*

In this way, we can compute optimal solutions to systems of linear integral equations such as the one shown above. For more information and related examples, see the references below.

References

[1] Pasqualina Conti and Carlo Traverso. Buchberger algorithm and integer programming. In *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, pages 130–139. Springer, 1991.

[2] David A Cox, John Little, and Donal O'shea. *Using algebraic geometry*, volume 185. Springer Science & Business Media, 2006.

[3] Bernd Sturmfels. *Gröbner bases and convex polytopes*, volume 8. American Mathematical Society, 1996.