Gröbner basis

From Wikipedia, the free encyclopedia

In mathematics, and more specifically in computer algebra, computational algebraic geometry, and computational commutative algebra, a **Gröbner basis** is a particular kind of generating set of an ideal in a polynomial ring $K[x_1, ..., x_n]$ over a field K. A Gröbner basis allows many important properties of the ideal and the associated algebraic variety to be deduced easily, such as the dimension and the number of zeros when it is finite. Gröbner basis computation is one of the main practical tools for solving systems of polynomial equations and computing the images of algebraic varieties under projections or rational maps.

Gröbner basis computation can be seen as a multivariate, non-linear generalization of both Euclid's algorithm for computing polynomial greatest common divisors, and Gaussian elimination for linear systems.[1]

Gröbner bases were introduced in 1965, together with an algorithm to compute them (Buchberger's algorithm), by Bruno Buchberger in his Ph.D. thesis. He named them after his advisor Wolfgang Gröbner. In 2007, Buchberger received the Association for Computing Machinery's Paris Kanellakis Theory and Practice Award for this work. However, the Russian mathematician Nikolai Günther had introduced a similar notion in 1913, published in various Russian mathematical journals. These papers were largely ignored by the mathematical community until their rediscovery in 1987 by Bodo Renschuch *et al.*[2] An analogous concept for multivariate power series was developed independently by Heisuke Hironaka in 1964, who named them **standard bases**. This term has been used by some authors to also denote Gröbner bases.

The theory of Gröbner bases has been extended by many authors in various directions. It has been generalized to other structures such as polynomials over principal ideal rings or polynomial rings, and also some classes of non-commutative rings and algebras, like Ore algebras.

Contents

Tools

Polynomial ring

Gröbner bases are primarily defined for ideals in a polynomial ring $R = K[x_1, \ldots, x_n]$ over a field K. Although the theory works for any field, most Gröbner basis computations are done either when K is the field of rationals or the integers modulo a prime number.

In the context of Gröbner bases, a nonzero polynomial in $R = K[x_1, \ldots, x_n]$ is commonly represented as a sum $c_1 M_1 + \cdots + c_m M_m$, where the c_i are nonzero elements of K, called *coefficients*, and the M_i are monomials (called *power products* by Buchberger and some of his followers) of the form $x_1^{a_1} \cdots x_n^{a_n}$, where the a_i are nonnegative integers. The vector $A = [a_1, \ldots, a_n]$ is called the *exponent vector* of the monomial. When the list $X = [x_1, \ldots, x_n]$ of the variables is fixed, the notation of monomials is often abbreviated as $x_1^{a_1} \cdots x_n^{a_n} = X^A$.

Monomials are uniquely defined by their exponent vectors, and, when a monomial ordering (see below) is fixed, a polynomial is uniquely represented by the ordered list of the ordered pairs formed by an exponent vector and the corresponding coefficient. This representation of polynomials is especially efficient for Gröbner basis computation in computers, although it is less convenient for other computations such as polynomial factorization and polynomial greatest common divisor.

If $F = \{f_1, \ldots, f_k\}$ is a finite set of polynomials in the polynomial ring R, the ideal generated by F is the set of linear combinations of elements of F with coefficients in R; that is the set of polynomials that can be written $\sum_{i=1}^{k} g_i f_i$ with $g_1, \ldots, g_k \in R$.

Monomial ordering

All operations related to Gröbner bases require the choice of a total order on the monomials, with the following properties of compatibility with multiplication. For all monomials M, N, P,

1. $M \leq N \Longleftrightarrow MP \leq NP$
2. $M \leq MP$.

A total order satisfying these condition is sometimes called an *admissible ordering*.

These conditions imply that the order is a well-order, that is, every strictly decreasing sequence of monomials is finite.

Although Gröbner basis theory does not depend on a particular choice of an admissible monomial ordering, three monomial orderings are specially important for the applications:

- *Lexicographical ordering*, commonly called *lex* or *plex* (for pure lexical ordering).
- *Total degree reverse lexicographical ordering*, commonly called *degrevlex*.
- *Elimination ordering*, *lexdeg*.

Gröbner basis theory was initially introduced for the lexicographical ordering. It was soon realised that the Gröbner basis for degrevlex is almost always much easier to compute, and that it is almost always easier to compute a lex Gröbner basis by first computing the degrevlex basis and then using a "change of ordering algorithm". When elimination is needed, degrevlex is not convenient; both lex and lexdeg may be used but, again, many computations are relatively easy with lexdeg and almost impossible with lex.

Basic operations

Leading term, coefficient and monomial

Once a monomial ordering is fixed, the terms of a polynomial (product of a monomial with its nonzero coefficient) are naturally ordered by decreasing monomials (for this order). This makes the representation of a polynomial as a sorted list of pairs coefficient–exponent vector a canonical representation of the polynomials (that is, two polynomials are equal if and only they have the same representation.

The first (greatest) term of a polynomial p for this ordering and the corresponding monomial and coefficient are respectively called the *leading term*, *leading monomial* and *leading coefficient* and denoted, in this article, $\operatorname{lt}(p)$, $\operatorname{lm}(p)$ and $\operatorname{lc}(p)$.

Most polynomial operations related to Gröbner bases involve the leading terms. So, the representation of polynomials as sorted lists make these operations particularly efficient (reading the first element of a list takes a constant time, independently of the length of the list).

Polynomial operations

The other polynomial operations involved by Gröbner basis computations are also compatible with the monomial ordering; that is, they can be performed without reordering the result:

- The addition of two polynomials consists in a merge of the two corresponding lists of terms, with a special treatment in the case of a conflict (that is, when the same monomial appears in the two polynomials).
- The multiplication of a polynomial by a scalar consists of multiplying each coefficient by this scalar, without any

other change in the representation.
- The multiplication of a polynomial by a monomial m consists of multiplying each monomial of the polynomial by m. This does not change the term ordering by definition of a monomial ordering.

Divisibility of monomials

Let $M = x_1^{a_1} \cdots x_n^{a_n}$ and $N = x_1^{b_1} \cdots x_n^{b_n}$ be two monomials, with exponent vectors $A = [a_1, \ldots, a_n]$ and $B = [b_1, \ldots, b_n]$.

One says that M *divides* N, or that N is a *multiple* of M, if $a_i \leq b_i$ for every i; that is, if A is componentwise not greater than B. In this case, the quotient $\frac{N}{M}$ is defined as $\frac{N}{M} = x_1^{b_1 - a_1} \cdots x_n^{b_n - a_n}$. In other words, the exponent vector of $\frac{N}{M}$ is the componentwise subtraction of the exponent vectors of N and M.

The *greatest common divisor* $\gcd(M, N)$ of M and N is the monomial $x_1^{\min(a_1, b_1)} \cdots x_n^{\min(a_n, b_n)}$ whose exponent vector is the componentwise minimum of A and B. The *least common multiple* $\operatorname{lcm}(M, N)$ is defined similarly with max instead of min.

One has

$$\operatorname{lcm}(M, N) = \frac{MN}{\gcd(M, N)}.$$

Reduction

The *reduction* of a polynomial by other polynomials with respect to a monomial ordering is central to Gröbner basis theory. It is a generalization of both row reduction occurring in Gaussian elimination and division steps of the Euclidean division of univariate polynomials.[1] When completed as much as possible, it is sometimes called **multivariate division** although its result is not uniquely defined.

Lead-reduction is a special case of reduction that is easier to compute. It is fundamental for Gröbner basis computation, since general reduction is needed only at the end of a Gröbner basis computation, for getting a reduced Gröbner basis from a non-reduced one.

Let an admissible monomial ordering be fixed, to which refers every monomial comparison that will occur in this section.

A polynomial f is *lead-reducible* by another polynomial g if the leading monomial $\operatorname{lm}(f)$ is a multiple of $\operatorname{lm}(g)$. The polynomial f is *reducible* by g if some monomial of f is a multiple $\operatorname{lm}(g)$. (So, if f is lead-reducible by g, it is also reducible, but f may be reducible without being lead-reducible.)

Suppose that f is *reducible* by g, and let cm be a term of f such that the monomial m is a multiple of $\operatorname{lm}(g)$. A *one-step reduction* of f by g consists of replacing f by

$$\operatorname{red}_1(f,g) = f - \frac{c}{\operatorname{lc}(g)}\,\frac{m}{\operatorname{lm}(g)}\,g.$$

This operation removes the monomial m from f without changing the terms with a monomial greater than m (for the monomial ordering). In particular, a *one step lead-reduction* of f produces a polynomial whose all monomials are smaller than $\operatorname{lm}(f)$.

Given a finite set G of polynomials, one says that f is *reducible* or *lead-reducible* by G if it is reducible or lead-reducible, respectively, by at least one element g of G. In this case, a one-step reduction (resp. one-step lead-reduction) of f by G is any one-step reduction (resp. one-step lead-reduction) of f by an element of G.

The (complete) reduction (resp. lead-reduction) of f by G consists of iterating one-step reductions (respect. one-step lead reductions) until getting a polynomial that is irreducible (resp. lead-irreducible) by G. It is sometimes called a *normal form* of f by G. In general this form is not uniquely defined because there are, in general, several elements of G that can be used for reducing f; this non-uniqueness is the starting point of Gröbner basis theory.

The definition of the reduction shows immediately that, if h is a normal form of f by G, one has

$$f = h + \sum_{g\in G} q_g\, g,$$

where h is irreducible by G and the q_g are polynomials such that $\operatorname{lm}(q_g\, g) \leq \operatorname{lm}(f)$. In the case of univariate polynomials, if G consists of a single element g, then h is the remainder of the Euclidean division of f by g, and q_g is the quotient. Moreover, the division algorithm is exactly the process of lead-reduction. For this reason, some authors use the term *multivariate division* instead of reduction.

Non uniqueness of reduction

In the example that follows, there are exactly two complete lead-reductions that produce two very different results. The fact that the results are irreducible (not only lead-irreducible) is specific to the example, although this is rather common with such small examples.

In this two variable example, the monomial ordering that is used is the lexicographic order with $x > y$, and we consider the reduction of

$$f = 2x^3 - x^2y + y^3 + 3y$$

by $G = \{g_1, g_2\}$, with

$$g_1 = x^2 + y^2 - 1,$$
$$g_2 = xy - 2.$$

For the first reduction step, either the first or the second term of *f* may be reduced. However, the reduction of a term amounts to removing this term at the cost of adding new lower terms; if it is not the first reducible term that is reduced, it may occur that a further reduction adds a similar term, which must be reduced again. It is therefore always better to reduce first the largest (for the monomial order) reducible term; that is, in particular, to lead-reduce first until getting a lead-irreducible polynomial.

The leading term $2x^3$ of *f* is reducible by g_1 and not by g_2. So the first reduction step consists of multiplying g_1 by –2*x* and adding the result to *f*:

$$f \xrightarrow{-2xg_1} f_1 = f - 2xg_1 = -x^2y - 2xy^2 + 2x + y^3 + 3y.$$

The leading term $-x^2y$ of f_1 is a multiple of the leading monomials of both g_1 and g_2, So, one has two choices for the second reduction step. If one chooses g_2, one gets a polynomial that can be reduced again by g_2:

$$f \xrightarrow{-2xg_1} f_1 \xrightarrow{xg_2} -2xy^2 + y^3 + 3y \xrightarrow{2yg_2} f_2 = y^3 - y.$$

No further reduction is possible, so f_2 is a complete reduction of *f*.

One gets a different result with the other choice for the second step:

$$f \xrightarrow{-2xg_1} f_1 \xrightarrow{yg_1} -2xy^2 + 2x + 2y^3 + 2y \xrightarrow{2yg_2} f_3 = 2x + 2y^3 - 2y.$$

Again, the result f_3 is irreducible, although only lead reductions were done.

In summary, the complete reduction of f can result in either $f_2 = y^3 - y$ or $f_3 = 2x + 2y^3 - 2y$.

This is for dealing with the problems set by this non-uniqueness that Buchberger introduced Gröbner bases and S-polynomials. Intuitively, $0 = f - f$ may be reduced to $f_2 - f_3$. This implies that $f_2 - f_3$ belongs to the ideal generated by G. So, this ideal is not changed by adding $f_3 - f_2$ to G, and this allows more reductions. In particular, f_3 can be reduced to f_2 by $f_3 - f_2$ and this restores the uniqueness of the reduced form.

Here Buchberger's algorithm for Gröbner bases would begin by adding to G the polynomial

$$g_3 = yg_1 - xg_2 = 2x + y^3 - y.$$

This polynomial, called S-polynomial by Buchberger is the difference of the one-step reductions of the least common multiple x^2y of the leading monomials of g_1 and g_2 (in this example, one has $g_3 = f_3 - f_2$). This does not complete Buchberger's algorithm, as xy gives different results, when reduced by g_2 or g_3.

S-polynomial

The S-polynomial, also called *critical pair*, with respect of a given monomial ordering, of two polynomials f and g is the polynomial

$$\begin{aligned} S(f,g) &= \operatorname{red}_1(\operatorname{lcm}, g) - \operatorname{red}_1(\operatorname{lcm}, f) \\ &= \frac{1}{\operatorname{lc}(f)} \frac{\operatorname{lm}(g)}{\operatorname{gcd}} f - \frac{1}{\operatorname{lc}(g)} \frac{\operatorname{lm}(f)}{\operatorname{gcd}} g, \end{aligned}$$

where lcm and gcd denote respectively the least common multiple and the greatest common divisor of the leading monomials of f and g.

As the monomials that are reducible by both f and g are exactly the multiples of lcm, one can deal with all cases of non-uniqueness of the reduction by considering only the S-polynomials. This is a fundamental fact for Gröbner basis theory and all algorithms for computing them.

Definition

Let $R = F[x_1, \ldots, x_n]$ be a polynomial ring over a field F. In this section, we suppose that an admissible monomial ordering has been fixed.

Let G be a finite set of polynomials in R that generates an ideal I. The set G is a Gröbner basis (with respect to the monomial ordering), or, more precisely, a Gröbner basis of I if

1. the ideal generated by the leading monomials of the polynomials in I equals the ideal generated by the leading monomials of G,

or, equivalently,

2. the leading monomial of every polynomial in I is a multiple of the leading monomial of some polynomial in G.

There are many characterizing properties, which can each be taken as an equivalent definition of Gröbner bases. For conciseness, in the following list, the notation "one-word/another word" means that one can take either "one-word" or "another word" for having two different characterizations of Gröbner bases. All the following assertions are characterizations of Gröbner bases:

3. a polynomial f is in I, if and only if some/every complete lead-reduction/reduction of f by G produces the zero polynomial;
4. for every S-polynomial s of elements of G, some/every complete lead-reduction/reduction of s by G produces zero;
5. all complete reductions of an element of R produce the same result;
6. the monomials that are irreducible by G form a basis of the F-vector space R/I.

Counting the above definition, this provides 12 characterizations of Gröbner bases. The fact that so many characterizations are possible makes Gröbner bases very useful. For example, condition 3 provides an algorithm for testing ideal membership; condition 4 provides an algorithm for testing whether a set of polynomials is a Gröbner basis and forms the basis of Buchberger's algorithm for computing Gröbner bases; conditions 5 and 6 allow computing in R/I in a way that is very similar to modular arithmetic.

Existence of Gröbner bases. For every admissible monomial ordering and every finite set G of polynomials, there is a Gröbner basis that contains G and generates the same ideal. Moreover, such a Gröbner basis may be computed with Buchberger's algorithm.

This algorithm uses condition 4, and proceeds roughly as follows: add to G all nonzero results of a complete reduction by G of a S-polynomial of two elements of G; repeat this operation with the new elements of G included until, eventually, all reductions produce zero. The algorithm terminates always because of Dickson's lemma or because polynomial rings are Noetherian (Hilbert's basis theorem). Condition 4 insures that the result is a Gröbner basis.

Reduced Gröbner bases

A Gröbner basis is **minimal** if all leading monomials of its elements are irreducible by the other elements of the basis. Given a Gröbner basis of an ideal I, one gets a minimal Gröbner basis of I by removing the polynomials whose leading monomials are multiple of the leading monomial of another element of the Gröbner basis. However, if two polynomials of the basis have the same leading monomial, only one must be removed. So, every Gröbner basis contains a minimal Gröbner basis as a subset.

All minimal Gröbner bases of a given ideal (for a fixed monomial ordering) have the same number of elements, and the same leading monomials, and the non-minimal Gröbner bases have more elements than the minimal ones.

A Gröbner basis is **reduced** if every polynomial in it is irreducible by the other elements of the basis, and has 1 as leading coefficient. So, every reduced Gröbner basis is minimal, but a minimal Gröbner basis needs not to be reduced.

Given a Gröbner basis of an ideal I, one gets a reduced Gröbner basis of I by first removing the polynomials that are lead-reducible by other elements of the basis (for getting a minimal basis); then replacing each element of the basis by the result of the complete reduction by the other elements of the basis; and, finally, by dividing each element of the basis by its leading coefficient.

All reduced Gröbner bases of an ideal (for a fixed monomial ordering) are equal. It follows that two ideals are equal if and only if they have the same reduced Gröbner basis.

Sometimes, reduced Gröbner bases are defined without the condition on the leading coefficients. In this case, the uniqueness of reduced Gröbner bases is true only up to the multiplication of polynomials by a nonzero constant.

When working with polynomials over the field $\mathbb{Q}$ of the rational numbers, it is useful to work only with polynomials with integer coefficients. In this case, the condition on the leading coefficients in the definition of a reduced basis may be replaced by the condition that all elements of the basis are primitive polynomials with integer coefficients, with positive leading coefficients. This restores the uniqueness of reduced bases.

In most implementations of Gröbner basis computation, the Gröbner bases that are output are always reduced.

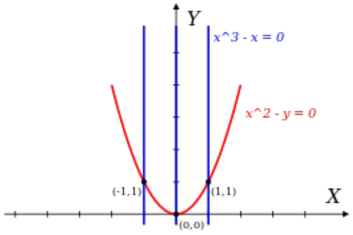
Special cases

For every monomial ordering, the empty set of polynomials is the unique Gröbner basis of the zero ideal.

For every monomial ordering, a set of polynomials that contains a nonzero constant is a Gröbner basis of the unit ideal (the whole polynomial ring). Conversely, every Gröbner basis of the unit ideal contains a nonzero constant. The reduced Gröbner basis of the unit is formed by the single polynomial 1.

In the case of polynomials in a single variable, there is a unique admissible monomial ordering, the ordering by the degree. The minimal Gröbner bases are the singletons consisting of a single polynomial. The reduced Gröbner bases are the monic polynomials.

Example and counterexample



The zeroes of f form the red parabola; the zeroes of g form the three blue vertical lines. Their intersection consists of three points.

Let $R = \mathbb{Q}[x, y]$ be the ring of bivariate polynomials with rational coefficients and consider the ideal $I = \langle f, g \rangle$ generated by the polynomials

$$f = x^2 - y,$$
$$g = x^3 - x.$$

By reducing g by f, one obtains a new polynomial k such that $I = \langle f, k \rangle$:

$$k = g - xf = xy - x.$$

None of f and k is reducible by the other, but xk is reducible by f, which gives another polynomial in I:

$$h = xk - (y - 1)f = y^2 - y.$$

Under lexicographic ordering with $x > y$ we have

$$\mathrm{lt}(f) = x^2$$
$$\mathrm{lt}(k) = xy$$
$$\mathrm{lt}(h) = y^2$$

As f, k and h belong to I, and none of them is reducible by the others, none of $\{f, k\}$, $\{f, h\}$, and $\{h, k\}$ is a Gröbner basis of I.

On the other hand, $\{f, k, h\}$ is a Gröbner basis of I, since the S-polynomials

$$\begin{aligned} yf - xk &= y(x^2 - y) - x(xy - x) = f - h \\ yk - xh &= y(xy - x) - x(y^2 - y) = 0 \\ y^2 f - x^2 h &= y(yf - xk) + x(yk - xh) \end{aligned}$$

can be reduced to zero by f, k and h.

The method that has been used here for finding h and k, and proving that $\{f, k, h\}$ is a Gröbner basis is a direct application of Buchberger's algorithm. So, it can be applied mechanically to any similar example, although, in general, there are many polynomials and S-polynomials to consider, and the computation is generally too large for being done without a computer.

Properties and applications of Gröbner bases

Unless explicitly stated, all the results that follow[3] are true for any monomial ordering (see that article for the definitions of the different orders that are mentioned below).

It is a common misconception that the lexicographical order is needed for some of these results. On the contrary, the lexicographical order is, almost always, the most difficult to compute, and using it makes impractical many computations that are relatively easy with graded reverse lexicographic order (grevlex), or, when elimination is needed, the elimination order (lexdeg) which restricts to grevlex on each block of variables.

Equality of ideals

Reduced Gröbner bases are *unique* for any given ideal and any monomial ordering. Thus two ideals are equal if and only if they have the same (reduced) Gröbner basis (usually a Gröbner basis software always produces reduced Gröbner bases).

Membership and inclusion of ideals

The reduction of a polynomial f by the Gröbner basis G of an ideal I yields 0 if and only if f is in I. This allows to test the membership of an element in an ideal. Another method consists in verifying that the Gröbner basis of $G \cup \{f\}$ is equal to G.

To test if the ideal I generated by $f_1, ..., f_k$ is contained in the ideal J, it suffices to test that every f_I is in J. One may also test the equality of the reduced Gröbner bases of J and $J \ \cup \ \{f_1, ...,f_k\}$.

Solutions of a system of algebraic equations

Any set of polynomials may be viewed as a system of polynomial equations by equating the polynomials to zero. The set of the solutions of such a system depends only on the generated ideal, and, therefore does not change when the given generating set is replaced by the Gröbner basis, for any ordering, of the generated ideal. Such a solution, with coordinates in an algebraically closed field containing the coefficients of the polynomials, is called a *zero of the ideal*. In the usual case of rational coefficients, this algebraically closed field is chosen as the complex field.

An ideal does not have any zero (the system of equations is inconsistent) if and only if 1 belongs to the ideal (this is Hilbert's Nullstellensatz), or, equivalently, if its Gröbner basis (for any monomial ordering) contains 1, or, also, if the corresponding reduced Gröbner basis is [1].

Given the Gröbner basis G of an ideal I, it has only a finite number of zeros, if and only if, for each variable x, G contains a polynomial with a leading monomial that is a power of x (without any other variable appearing in the leading term). If this is the case, then the number of zeros, counted with multiplicity, is equal to the number of monomials that are not multiples of any leading monomial of G. This number is called the *degree* of the ideal.

When the number of zeros is finite, the Gröbner basis for a lexicographical monomial ordering provides, theoretically, a solution: the first coordinate of a solution is a root of the greatest common divisor of polynomials of the basis that depend only on the first variable. After substituting this root in the basis, the second coordinate of this solution is a root of the greatest common divisor of the resulting polynomials that depend only on the second variable, and so on. This solving process is only theoretical, because it implies GCD computation and root-finding of polynomials with approximate coefficients, which are not practicable because of numeric instability. Therefore, other methods have been developed to solve polynomial systems through Gröbner bases (see System of polynomial equations for more details).

Dimension, degree and Hilbert series

The **dimension** of an ideal I in a polynomial ring R is the Krull dimension of the ring R/I and is equal to the dimension of the algebraic set of the zeros of I. It is also equal to number of hyperplanes in general position which are needed to have an intersection with the algebraic set, which is a finite number of points. The **degree** of the ideal and of its associated algebraic set is the number of points of this finite intersection, counted with multiplicity. In particular, the degree of a hypersurface is equal to the degree of its definition polynomial.

Both degree and dimension depend only on the set of the leading monomials of the Gröbner basis of the ideal for any monomial ordering.

The dimension is the maximal size of a subset S of the variables such that there is no leading monomial depending only on the variables in S. Thus, if the ideal has dimension 0, then for each variable x there is a leading monomial in the Gröbner basis that is a power of x.

Both dimension and degree may be deduced from the Hilbert series of the ideal, which is the series $\sum_{i=0}^{\infty} d_i t^i$, where d_i is the number of monomials of degree i that are not multiple of any leading monomial in the Gröbner basis. The Hilbert series may be summed into a rational fraction

$$\sum_{i=0}^{\infty} d_i t^i = \frac{P(t)}{(1-t)^d},$$

where d is the dimension of the ideal and $P(t)$ is a polynomial such that $P(1)$ is the degree of the ideal.

Although the dimension and the degree do not depend on the choice of the monomial ordering, the Hilbert series and the polynomial $P(t)$ change when one changes the monomial ordering.

Most computer algebra systems that provide functions to compute Gröbner bases provide also functions for computing the Hilbert series, and thus also the dimension and the degree.

Elimination

The computation of Gröbner bases for an *elimination* monomial ordering allows computational elimination theory. This is based on the following theorem.

Consider a polynomial ring $K[x_1, \ldots, x_n, y_1, \ldots, y_m] = K[X, Y]$, in which the variables are split into two subsets X and Y. Let us also choose an elimination monomial ordering "eliminating" X, that is a monomial ordering for which two monomials are compared by comparing first the X-parts, and, in case of equality only, considering the Y-parts. This implies that a monomial containing an X-variable is greater than every monomial independent of X. If G is a Gröbner basis of an ideal I for this monomial ordering, then $G \cap K[Y]$ is a Gröbner basis of $I \cap K[Y]$ (this ideal is often called the *elimination ideal*). Moreover, $G \cap K[Y]$ consists exactly of the polynomials of G whose leading terms belong to $K[Y]$ (this makes very easy the computation of $G \cap K[Y]$, as only the leading monomials need to be checked).

This *elimination property* has many applications, some described in the next sections.

Another application, in algebraic geometry, is that *elimination* realizes the geometric operation of projection of an affine algebraic set into a subspace of the ambient space: with above notation, the (Zariski closure of) the projection of the algebraic set defined by the ideal I into the Y-subspace is defined by the ideal $I \cap K[Y]$.

The lexicographical ordering such that $x_1 > \cdots > x_n$ is an elimination ordering for every partition $\{x_1, \ldots, x_k\}, \{x_{k+1}, \ldots, x_n\}$. Thus a Gröbner basis for this ordering carries much more information than usually necessary. This may explain why Gröbner bases for the lexicographical ordering are usually the most difficult to compute.

Intersecting ideals

If I and J are two ideals generated respectively by $\{f_1, ..., f_m\}$ and $\{g_1, ..., g_k\}$, then a single Gröbner basis computation produces a Gröbner basis of their intersection $I \cap J$. For this, one introduces a new indeterminate t, and one uses an elimination ordering such that the first block contains only t and the other block contains all the other variables (this means that a monomial containing t is greater than every monomial that does not contain t). With this monomial ordering, a Gröbner basis of $I \cap J$ consists in the polynomials that do not contain t, in the Gröbner basis of the ideal

$$K = \langle tf_1, \ldots, tf_m, (1-t)g_1, \ldots, (1-t)g_k \rangle.$$

In other words, $I \cap J$ is obtained by *eliminating* t in K. This may be proven by remarking that the ideal K consists in the polynomials $(a-b)t+b$ such that $a \in I$ and $b \in J$. Such a polynomial is independent of t if and only if $a=b$, which means that $b \in I \cap J$.

Implicitization of a rational curve

A rational curve is an algebraic curve that has a set of parametric equations of the form

$$\begin{aligned} x_1 &= \frac{f_1(t)}{g_1(t)} \\ &\vdots \\ x_n &= \frac{f_n(t)}{g_n(t)}, \end{aligned}$$

where $f_i(t)$ and $g_i(t)$ are univariate polynomials for $1 \leq i \leq n$. One may (and will) suppose that $f_i(t)$ and $g_i(t)$ are coprime (they have no non-constant common factors).

Implicitization consists in computing the implicit equations of such a curve. In case of $n = 2$, that is for plane curves, this may be computed with the resultant. The implicit equation is the following resultant:

$$\operatorname{Res}_t(g_1x_1 - f_1, g_2x_2 - f_2).$$

Elimination with Gröbner bases allows to implicitize for any value of n, simply by eliminating t in the ideal $\langle g_1x_1 - f_1, \ldots, g_nx_n - f_n\rangle$. If $n = 2$, the result is the same as with the resultant, if the map $t \mapsto (x_1, x_2)$ is injective for almost every t. In the other case, the resultant is a power of the result of the elimination.

Saturation

When modeling a problem by polynomial equations, it is often assumed that some quantities are non-zero, so as to avoid degenerate cases. For example, when dealing with triangles, many properties become false if the triangle degenerates to a line segment, i.e. the length of one side is equal to the sum of the lengths of the other sides. In such situations, one cannot deduce relevant information from the polynomial system unless the degenerate solutions are ignored. More precisely, the system of equations defines an algebraic set which may have several irreducible components, and one must remove the components on which the degeneracy conditions are everywhere zero.

This is done by *saturating* the equations by the degeneracy conditions, which may be done via the elimination property of Gröbner bases.

Definition of the saturation

The localization of a ring consists in adjoining to it the formal inverses of some elements. This section concerns only the case of a single element, or equivalently a finite number of elements (adjoining the inverses of several elements is equivalent to adjoining the inverse of their product). The *localization* of a ring R by an element f is the ring $R_f = R[t]/(1 - ft)$, where t is a new indeterminate representing the inverse of f. The *localization* of an ideal I of R is the ideal R_fI of I_f. When R is a polynomial ring, computing in R_f is not efficient because of the need to manage the denominators. Therefore, localization is usually replaced by the operation of *saturation*.

The **saturation** with respect to f of an ideal I in R is the inverse image of R_fI under the canonical map from R to R_f. It is the ideal $I : f^\infty = \{g \in R \mid (\exists k \in \mathbb{N}) f^k g \in I\}$ consisting in all elements of R whose product with some power of f belongs to I.

If J is the ideal generated by I and $1-ft$ in $R[t]$, then $I : f^\infty = J \cap R$. It follows that, if R is a polynomial ring, a Gröbner basis computation eliminating t produces a Gröbner basis of the saturation of an ideal by a polynomial.

The important property of the saturation, which ensures that it removes from the algebraic set defined by the ideal I the irreducible components on which the polynomial *f* is zero, is the following: *The primary decomposition of* $I : f^\infty$ *consists of the components of the primary decomposition of I that do not contain any power of f.*

Computation of the saturation

A Gröbner basis of the saturation by *f* of a polynomial ideal generated by a finite set of polynomials *F*, may be obtained by eliminating t in $F \cup \{1 - tf\}$, that is by keeping the polynomials independent of t in the Gröbner basis of $F \cup \{1 - tf\}$ for an elimination ordering eliminating t.

Instead of using *F*, one may also start from a Gröbner basis of *F*. Which method is most efficient depends on the problem. However, if the saturation does not remove any component, that is if the ideal is equal to its saturated ideal, computing first the Gröbner basis of *F* is usually faster. On the other hand, if the saturation removes some components, the direct computation may be dramatically faster.

If one wants to saturate with respect to several polynomials $f_1, \ldots, f_k$ or with respect to a single polynomial which is a product $f = f_1 \cdots f_k$, there are three ways to proceed which give the same result but may have very different computation times (it depends on the problem which is the most efficient).

- Saturating by $f = f_1 \cdots f_k$ in a single Gröbner basis computation.
- Saturating by f_1, then saturating the result by f_2, and so on.
- Adding to *F* or to its Gröbner basis the polynomials $1 - t_1 f_1, \ldots, 1 - t_k f_k$, and eliminating the t_i in a single Gröbner basis computation.

Effective Nullstellensatz

Hilbert's Nullstellensatz has two versions. The first one asserts that a set of polynomials has no common zeros over an algebraic closure of the field of the coefficients, if and only if 1 belongs to the generated ideal. This is easily tested with a Gröbner basis computation, because 1 belongs to an ideal if and only if 1 belongs to the Gröbner basis of the ideal, for any monomial ordering.

The second version asserts that the set of common zeros (in an algebraic closure of the field of the coefficients) of an ideal is contained in the hypersurface of the zeros of a polynomial *f*, if and only if a power of *f* belongs to the ideal. This may be tested by saturating the ideal by *f*; in fact, a power of *f* belongs to the ideal if and only if the saturation by *f* provides a Gröbner basis containing 1.

Implicitization in higher dimension

By definition, an affine rational variety of dimension k may be described by parametric equations of the form

$$\begin{aligned} x_1 &= \frac{p_1}{p_0} \\ &\vdots \\ x_n &= \frac{p_n}{p_0}, \end{aligned}$$

where $p_0, \ldots, p_n$ are $n+1$ polynomials in the k variables (parameters of the parameterization) $t_1, \ldots, t_k$. Thus the parameters $t_1, \ldots, t_k$ and the coordinates $x_1, \ldots, x_n$ of the points of the variety are zeros of the ideal

$$I = \langle p_0 x_1 - p_1, \ldots, p_0 x_n - p_n \rangle \, .$$

One could guess that it suffices to eliminate the parameters to obtain the implicit equations of the variety, as it has been done in the case of curves. Unfortunately this is not always the case. If the p_i have a common zero (sometimes called *base point*), every irreducible component of the non-empty algebraic set defined by the p_i is an irreducible component of the algebraic set defined by I. It follows that, in this case, the direct elimination of the t_i provides an empty set of polynomials.

Therefore, if $k>1$, two Gröbner basis computations are needed to implicitize:

1. Saturate I by p_0 to get a Gröbner basis G
2. Eliminate the t_i from G to get a Gröbner basis of the ideal (of the implicit equations) of the variety.

Algorithms and implementations

Buchberger's algorithm is the oldest algorithm for computing Gröbner bases. It has been devised by Bruno Buchberger together with the Gröbner basis theory. It is straightforward to implement, but it appeared soon that raw implementations can solve only trivial problems. The main issues are the following ones:

1. Even when the resulting Gröbner basis is small, the intermediate polynomials can be huge. It result that most of the computing time may be spent in memory management. So, specialized memory management algorithms may be a fundamental part of an efficient implementation.
2. The integers occurring during a computation may be sufficiently large for making fast multiplication algorithms

and multimodular arithmetic useful. For this reason, most optimized implementations use the GMPlibrary. Also, modular arithmetic, Chinese remainder theorem and Hensel lifting are used in optimized implementations

3. The choice of the S-polynomials to reduce and of the polynomials used for reducing them is devoted to heuristics. As in many computational problems, heuristics cannot detect most hidden simplifications, and if heuristic choices are avoided, one may get a dramatic improvement of the algorithm efficiency.
4. In most cases most S-polynomials that are computed are reduced to zero; that is, most computing time is spent to compute zero.
5. The monomial ordering that is most often needed for the applications (pure lexicographic) is not the ordering that leads to the easiest computation, generally the ordering *degrevlex*.

For solving 3. many improvements, variants and heuristics have been proposed before the introduction of F4 and F5 algorithms by Jean-Charles Faugère. As these algorithms are designed for integer coefficients or with coefficients in the integers modulo a prime number, Buchberger's algorithm remains useful for more general coefficients.

Roughly speaking, F4 algorithm solves 3. by replacing many S-polynomial reductions by the row reduction of a single large matrix for which advanced methods of linear algebra can be used. This solves partially issue 4., as reductions to zero in Buchberger's algorithm correspond to relations between rows of the matrix to be reduced, and the zero rows of the reduced matrix correspond to a basis of the vector space of these relations.

F5 algorithm improves F4 by introducing a criterion that allows reducing the size of the matrices to be reduced. This criterion is almost optimal, since the matrices to be reduced have full rank in sufficiently regular cases (in particular, when the input polynomials form a regular sequence). Tuning F5 for a general use is difficult, since its performances depend on an order on the input polynomials and a balance between the incrementation of the working polynomial degree and of the number of the input polynomials that are considered. To date (2022), there is no distributed implementation that is significantly more efficient than F4, but, over modular integers F5 has been used successfully for several cryptographic challenges; for example, for breaking HFE challenge.

Issue 5. has been solved by the discovery of basis conversion algorithms that start from the Gröbner basis for one monomial ordering for computing a Gröbner basis for another monomial ordering. FGLM algorithm is such a basis conversion algorithm that works only in the zero-dimensional case (where the polynomials have a finite number of complex common zeros) and has a polynomial complexity in the number of common zeros. A basis conversion algorithm that works is the general case is the *Gröbner walk algorithm*.[4] In its original form, FGLM may be the critical step for solving systems of polynomial equations because FGML does not take into account the sparsity of involved matrices. This has been fixed by the introduction of *sparse FGLM algorithms*.[5]

Most general-purpose computer algebra systems have implementations of one or several algorithms for Gröbner bases, often also embedded in other functions, such as for solving systems of polynomial equations or for simplifying trigonometric functions; this is the case, for example, of CoCoA, GAP , Macaulay 2, Magma, Maple, Mathematica, SINGULAR, SageMath and SymPy. When F4 is available, it is generally much more efficient than Buchberger's algorithm. The implementation techniques and algorithmic variants are not always documented, although they may have a dramatic effect on efficiency.

As of 2022, the fastest implementations of F4 and (sparse)-FGLM seem to be those of the library *Msolve*.[6] Beside Gröbner algorithms, Msolve contains fast algorithms for real-root isolation, and combines all these functions in an algorithm for the real solutions of systems of polynomial equations that outperforms dramatically the other software for this problem (Maple and Magma).[6] Msolve is available on GitHub, and is interfaced with Julia, Maple and SageMath; this means that Msolve can be used directly from within these software environments.

Complexity

The complexity of the Gröbner basis computations is commonly evaluated in term of the number n of variables and the maximal degree d of the input polynomials.

In the worst case, the main parameter of the complexity is the maximal degree of the elements of the resulting reduced Gröbner basis. More precisely, if the Gröbner basis contains an element of a large degree D, this element may contain $\Omega(D^n)$ nonzero terms whose computation requires a time of $\Omega(D^n) > D^{\Omega(n)}$. On the other hand, if all polynomials in the reduced Gröbner basis a homogeneous ideal have a degree of at most D, the Gröbner basis can be computed by linear algebra on the vector space of polynomials of degree less than $2D$, which has a dimension $O(D^n)$. So, the complexity of this computation is $O(D^n)^{O(1)} = D^{O(n)}$.

The worst-case complexity of a Gröbner basis computation is doubly exponential in n. More precisely, the complexity is upper bounded by a polynomial in d^{2^n}. Using little o notation, it is therefore bounded by $d^{2^{n+o(n)}}$. On the other hand, examples have been given of reduced Gröbner bases containing polynomials of degree $d^{2^{\Omega(n)}}$, or containing $d^{2^{\Omega(n)}}$ elements. As every algorithm for computing a Gröbner basis must write its result, this provides a lower bound of the complexity.

Generalizations

The concept and algorithms of Gröbner bases have been generalized to submodules of free modules over a polynomial ring. In fact, if L is a free module over a ring R, then one may consider the direct sum $R \oplus L$ as a ring by defining the product of two elements of L to be 0. This ring may be identified with $R[e_1, \ldots, e_l] / \left\langle \{e_i e_j | 1 \leq i \leq j \leq l\} \right\rangle$, where $e_1, \ldots, e_l$ is a basis of L. This allows identifying a submodule of L generated by $g_1, \ldots, g_k$ with the ideal of

$R[e_1, \ldots, e_l]$ generated by $g_1, \ldots, g_k$ and the products $e_i e_j$, $1 \leq i \leq j \leq l$. If R is a polynomial ring, this reduces the theory and the algorithms of Gröbner bases of modules to the theory and the algorithms of Gröbner bases of ideals.

The concept and algorithms of Gröbner bases have also been generalized to ideals over various rings, commutative or not, like polynomial rings over a principal ideal ring or Weyl algebras.

Areas of applications

Error-Correcting Codes

Gröbner basis has been applied in the theory of error-correcting codes for algebraic decoding. By using Gröbner basis computation on various forms of error-correcting equations, decoding methods were developed for correcting errors of cyclic codes,[7] affine variety codes,[8] algebraic-geometric codes and even general linear block codes.[9] Applying Gröbner basis in algebraic decoding is still a research area of channel coding theory.

See also

- Graver basis
- Janet basis
- Regular chains, an alternative way to represent algebraic sets

References

1. Lazard, Daniel (1983). "Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations". *Computer Algebra*. Lecture Notes in Computer Science. Vol. 162. pp. 146–156. doi:10.1007/3-540-12868-9_99 (https://doi.org/10.1007%2F3-540-12868-9_99). ISBN 978-3-540-12868-7.
2. Renschuch, Bodo; Roloff, Hartmut; Rasputin, Georgij G.; Abramson, Michael (June 2003). "Contributions to constructive polynomial ideal theory XXIII: forgotten works of Leningrad mathematician N. M. Gjunter on polynomial ideal theory" (http://www.risc.jku.at/Groebner-Bases-Bibliography/gbbib_files/publication_776.pdf) (PDF). *SIGSAM Bull*. **37** (2): 35–48. doi:10.1145/944567.944569 (https://doi.org/10.1145%2F944567.944569). S2CID 1819694 (https://api.semanticscholar.org/CorpusID:1819694).
3. Cox, David A.; Little, John; O'Shea, Donal (1997). *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. Springer. ISBN 0-387-94680-2.
4. Collart, Stéphane; Kalkbrener, Michael; Mall, Daniel (1997). "Converting bases with the Gröbner walk" (https://doi.org/10.1006%2Fjsco.1996.0145). *Journal of Symbolic Computation*. Elsevier. **24** (3–4): 465–469. doi:10.1006/jsco.1996.0145 (https://doi.org/10.1006%2Fjsco.1996.0145).

5. Faugère, Jean-Charles; Chenqi, Mou (2017). "Sparse FGLM algorithms" (https://doi.org/10.1016%2Fj.jsc.2016.07.025). *Journal of Symbolic Computation*. Elsevier. **80**: 538–569. doi:10.1016/j.jsc.2016.07.025 (https://doi.org/10.1016%2Fj.jsc.2016.07.025). S2CID 149627 (https://api.semanticscholar.org/CorpusID:149627).
6. Berthomieu \first1=Jérémy; Eder, Christian; Safey El Din, Mohab (2021). *Msolve: a library for solving polynomial systems* (https://hal.sorbonne-universite.fr/hal-03191666). 2021 International Symposium on Symbolic and Algebraic Computation. 46th International Symposium on Symbolic and Algebraic Computation. Saint Petersburg, Russia. doi:10.1145/3452143.3465545 (https://doi.org/10.1145%2F3452143.3465545).
7. Chen, X.; Reed, I.S.; Helleseth, T.; Truong, T.K. (1994). "Use of Gröbner bases to decode binary cyclic codes up to the true minimum distance" (https://ieeexplore.ieee.org/document/333885). *IEEE Transactions on Information Theory*. **40** (5): 1654–61. doi:10.1109/18.333885 (https://doi.org/10.1109%2F18.333885).
8. Fitzgerald, J.; Lax, R.F. (1998). "Decoding affine variety codes using Gröbner bases". *Designs, Codes and Cryptography*. **13** (2): 147–158. doi:10.1023/A:1008274212057 (https://doi.org/10.1023%2FA%3A1008274212057). S2CID 2515114 (https://api.semanticscholar.org/CorpusID:2515114).
9. Bulygin, S.; Pellikaan, R. (2009). "Decoding linear error-correcting codes up to half the minimum distance with Gröbner bases". *Gröbner Bases, Coding, and Cryptography*. Springer. pp. 361–5. ISBN 978-3-540-93805-7.

Further reading

- Adams, William W.; Loustaunau, Philippe (1994). *An Introduction to Gröbner Bases*. Graduate Studies in Mathematics. Vol. 3. American Mathematical Society. ISBN 0-8218-3804-0.
- Li, Huishi (2011). *Gröbner Bases in Ring Theory*. World Scientific. ISBN 978-981-4365-13-0.
- Becker, Thomas; Weispfenning, Volker (1998). *Gröbner Bases*. Graduate Texts in Mathematics. Vol. 141. Springer. ISBN 0-387-97971-9.
- Buchberger, Bruno (1965). *An Algorithm for Finding the Basis Elements of the Residue Class Ring of a Zero Dimensional Polynomial Ideal* (http://www.ricam.oeaw.ac.at/Groebner-Bases-Bibliography/gbbib_files/publication_706.pdf) (PDF) (PhD). University of Innsbruck. — (2006). Translated by Abramson, M. "Bruno Buchberger's PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal" (https://doi.org/10.1016%2Fj.jsc.2005.09.007). *Journal of Symbolic Computation*. **41** (3–4): 471–511. doi:10.1016/j.jsc.2005.09.007 (https://doi.org/10.1016%2Fj.jsc.2005.09.007). [This is Buchberger's thesis inventing Gröbner bases.]
- Buchberger, Bruno (1970). "An Algorithmic Criterion for the Solvability of a System of Algebraic Equations" (http://www.ricam.oeaw.ac.at/Groebner-Bases-Bibliography/gbbib_files/publication_699.pdf) (PDF). *Aequationes Mathematicae*. **4**: 374–383. doi:10.1007/BF01844169 (https://doi.org/10.1007%2FBF01844169). S2CID 189834323 (https://api.semanticscholar.org/CorpusID:189834323). (This is the journal publication of Buchberger's thesis.)Burchberger, B.; Winkler, F., eds. (26 February 1998). "An Algorithmic Criterion for the Solvability of a System of Algebraic Equations" (https://books.google.com/books?id=tfa7dpQf1OIC&pg=PA535). *Gröbner Bases and Applications*. London Mathematical Society Lecture Note Series. Vol. 251. Cambridge

University Press. pp. 535–545. ISBN 978-0-521-63298-0.

- Buchberger, Bruno; Kauers, Manuel (2010). "Gröbner Bases" (https://doi.org/10.4249%2Fscholarpedia.7763). *Scholarpedia*. **5** (10): 7763. Bibcode:2010SchpJ...5.7763B (https://ui.adsabs.harvard.edu/abs/2010SchpJ...5.7763B). doi:10.4249/scholarpedia.7763 (https://doi.org/10.4249%2Fscholarpedia.7763).
- Fröberg, Ralf (1997). *An Introduction to Gröbner Bases*. Wiley. ISBN 0-471-97442-0.
- Sturmfels, Bernd (November 2005). "What is ... a Gröbner Basis?" (http://math.berkeley.edu/~bernd/what-is.pdf) (PDF). *Notices of the American Mathematical Society*. **52** (10): 1199–1200, a brief introduction
- Shirshov, Anatoliĭ I. (1999). "Certain algorithmic problems for Lie algebras" (http://www.ricam.oeaw.ac.at/Groebner-Bases-Bibliography/gbbib_files/publication_835.pdf) (PDF). *ACM SIGSAM Bulletin*. **33** (2): 3–6. doi:10.1145/334714.334715 (https://doi.org/10.1145%2F334714.334715). S2CID 37070503 (https://api.semanticscholar.org/CorpusID:37070503). (translated from Sibirsk. Mat. Zh. Siberian Mathematics Journal **3** (1962), 292–296).
- Aschenbrenner, Matthias; Hillar, Christopher (2007). "Finite generation of symmetric ideals" (http://www.ams.org/tran/2007-359-11/S0002-9947-07-04116-5/home.html). *Transactions of the American Mathematical Society*. **359** (11): 5171–92. doi:10.1090/S0002-9947-07-04116-5 (https://doi.org/10.1090%2FS0002-9947-07-04116-5). S2CID 5656701 (https://api.semanticscholar.org/CorpusID:5656701). (on infinite dimensional Gröbner bases for polynomial rings in infinitely many indeterminates).

External links

- Faugère's own implementation of his F4 algorithm (https://web.archive.org/web/20120724023838/http://www-calfor.lip6.fr/~jcf/Software/FGb/Download/index.html)
- "Gröbner basis" (https://www.encyclopediaofmath.org/index.php?title=Gröbner_basis), *Encyclopedia of Mathematics*, EMS Press, 2001 [1994]
- Buchberger, B. (2003). "Gröbner Bases: A Short Introduction for Systems Theorists" (http://www.risc.uni-linz.ac.at/people/buchberg/papers/2001-02-19-A.pdf) (PDF). In Moreno-Diaz, R.; Buchberger, B.; Freire, J. (eds.). *Computer Aided Systems Theory — EUROCAST 2001: A Selection of Papers from the 8th International Workshop on Computer Aided Systems Theory* (https://books.google.com/books?id=HKxqCQAAQBAJ&pg=PA1). Springer. pp. 1–19. ISBN 978-3-540-45654-4.
- Buchberger, B.; Zapletal, A. "Gröbner Bases Bibliography" (http://www.ricam.oeaw.ac.at/Groebner-Bases-Bibliography/search.php).
- Comparative Timings Page for Gröbner Bases Software (http://magma.maths.usyd.edu.au/users/allan/gb)
- Prof. Bruno Buchberger (http://www.risc.jku.at/people/buchberg/) Bruno Buchberger
- Weisstein, Eric W. "Gröbner Basis" (https://mathworld.wolfram.com/GroebnerBasis.html). *MathWorld*.
- Gröbner basis introduction (http://www.scholarpedia.org/article/Gröbner_basis) on Scholarpedia