1987-03-00-

Research
Institute for
Symbolic
Computation

L I N Z
Johannes Kepler University, A-4040 Linz, Austria (Europe)

Publications / Reports

Editors: RISC-LINZ faculty

(B. Buchberger, A. Leitsch, F. Lichtenberger, P. Paule,
H. Rolletschek, F. Winkler, H. Zassenhaus)

Applications of Gröbner Bases in Non-Linear Computational Geometry

B. Buchberger

Proc. Workshop on Scientific Software (invited lecture), IMA, Minneapolis, USA March 23-26, 1987, pp. 59–88, IMA Volumes in Mathematics and its Applications, vol. 14, Springer

RISC-LINZ Series no. 87-15.0

Sponsored by:
Copyright Notice: This paper is published elsewhere. As a courtesy to the publisher distribution of this paper is strictly limited.

APPLICATIONS OF GRÖBNER BASES IN NON-LINEAR COMPUTATIONAL GEOMETRY

BRUNO BUCHBERGER[1]

Abstract

Gröbner bases are certain finite sets of multivariate polynomials. Many problems in polynomial ideal theory (algebraic geometry, non-linear computational geometry) can be solved by easy algorithms after transforming the polynomial sets involved in the specification of the problems into Gröbner basis form. In this paper we give some examples of applying the Gröbner bases method to problems in non-linear computational geometry (inverse kinematics in robot programming, collision detection for superellipsoids, implicitization of parametric representations of curves and surfaces, inversion problem for parametric representations, automated geometrical theorem proving, primary decomposition of implicitly defined geometrical objects). The paper starts with a brief summary of the Gröbner bases method.

1 Introduction

Traditionally, computational geometry deals with geometrical and *combinatorial* problems on *linear objects* and simple non-linear objects, see for example (Preparata, Shamos 1985). These methods are not appropriate for recent advanced problems arising in geometrical modeling, computer-aided design, and robot programming, which are more *algebraic* in nature and involve *non-linear geometrical objects.* Real and complex algebraic geometry is the natural framework for most of these non-linear problems. Unfortunately, in the past decades, algebraic geometry was very little concerned with the algorithmic solution of problems. Rather, *non-constructive* proofs of certain geometrical phenomena and mere existence proofs for certain geometrical objects was, and still is, the main emphasis.

The method of Gröbner bases is an *algorithmic* method that can be used to attack a wide range of problems in commutative algebra (polynomial ideal theory) and (complex) algebraic geometry. It is based on the concept of Gröbner bases and on an algorithm for constructing Gröbner bases introduced in (Buchberger 1965, 1970). In recent years the method has been refined and analyzed and more

[1] RISC-LINZ (Research Institute for Symbolic Computation) Johannes Kepler University, A4040 Linz, Austria. This research is supported by a grant from VOEST-ALPINE, Linz, (Dipl. Ing. H. Exner), and a grant from SIEMENS, München, (Dr. H. Schwärtzel)

applications have been studied. (Buchberger 1985) is a tutorial and survey on the Gröbner bases method.

The present paper starts with a brief summary of the *basic concepts and results of Gröbner bases theory* (Section 2). If the reader accepts these basic concepts and results as black boxes, the main part of the paper is self-contained. The internal details of the black boxes together with extensive references to the literature are given in the tutorial (Buchberger 1985).

The main part of the paper explains various applications of the Gröbner bases method for problems in non-linear computational geometry as motivated by advanced *applications* in computer-aided design, geometrical modeling and robot programming. The sequence for the presentation of these applications is quite random. Each of them relies on one or several of the basic properties of Gröbner bases summarized in Section 2.

2 Summary of the Gröbner Bases Method

The reader who is interested only in the applications may skip this section and come back in case he needs a specific notation, concept or theorem.

2.1 General Notation

$\mathbf{N}$	set of natural numbers including zero
$\mathbf{Q}$	set of rational numbers
$\mathbf{R}$	set of real numbers
$\mathbf{C}$	set of complex numbers
K	typed variable for arbitrary fields
$\overline{K}$	algebraic closure of K
i, j, k, l, m, n	typed variables for natural numbers
$K[x_1, \ldots, x_n]$	ring of n-variate polynomials over the coefficient field K
$K(x_1, \ldots, x_n)$	field of n-variate rational rational expressions over the coefficient field K
a, b, c, d	typed variables for elements in coefficient fields
f, g, h, p, q	typed variables for polynomials
s, t, u	typed variables for power products, i. e. polynomials of the form $x_1^{i_1} \ldots x_n^{i_n}$
$\mathrm{C}(f, u)$	the coefficient at power product u in polynomial f
F, G	typed variables for finite sets of polynomials
H	typed variable for finite sequences of polynomials
$\mathrm{Ideal}(F)$	the ideal generated by F, i. e. the set $\{\sum_i h_i.f_i \mid h_i \in K[x_1, \ldots, x_n], f_i \in F\}$
$\mathrm{Radical}(F)$	the radical of the ideal generated by F, i. e. $\{f \mid f$ vanishes on all common zeros of $F\}$ or, equivalently, $\{f \mid f^k \in \mathrm{Ideal}(F)$ for some $k\}$
$f \equiv_F g$	f is congruent to g modulo Ideal(F)
$K[x_1, \ldots, x_n]/\mathrm{Ideal}(F)$	the residue class ring modulo Ideal(F)
$[f]_F$	the residue class of f modulo Ideal(F)

In the definition of Ideal(F), it is sometimes necessary to explicitly indicate the polynomial ring from which the h_i are taken. If the polynomial ring is not clear from the context, we will use an index:

$\mathrm{Ideal}_{K[x_1,\ldots,x_n]}(F)$

In the definition of Radical(F), by a common zero of the polynomials in F we mean a common zero in the algebraic closure of the coefficient field.

2.2 Polynomial Reduction

The basic notion of Gröbner bases theory is *polynomial reduction.* The notion of polynomial reduction depends on a linear ordering on the set of power products that can be extended to a partial ordering on the set of polynomials. The set of "*admissible orderings*" that can be used for this purpose can be characterized by two easy axioms. The *lexical ordering* and the *total degree ordering* are the two admissible orderings used most often in examples. These two orderings are completely specified by fixing a linear ordering on the set of indeterminates $x_1, \ldots, x_n$ in the polynomial ring. Roughly, f reduces to g modulo F iff g results from f by subtracting a suitable multiple $a.u.h$ of a polynomial $h \in F$ such that g is lower in the admissible ordering than f. Reduction may be conceived as a generalization of the subtraction step that appears in univariate polynomial division. For all details, see (Buchberger 1985). We use the following notation:

$\succ$	typed variable for admissible orderings
LP(f)	leading power product of f (w. r. t. $\succ$)
LC(f)	leading coefficient of f (w. r. t. $\succ$)
MLP(F)	the set of "multiples of leading powerproducts in F", i. e. $\{u \mid (\exists f \in F)(u$ is a multiple of LP$(f)\}$
$f \to_F g$	f reduces to g modulo F
$\overset{*}{\to}_F$	reflexive-transitive closure of $\to_F$
$\overset{*}{\leftrightarrow}_F$	reflexive-symmetric-transitive closure of $\to_F$
$\underline{f}_F$	f is in normal form modulo F, i. e. there does not exist any g such that $f \to_F g$

A binary relation $\to$ on a set M is called "*noetherian*" iff there does not exist any infinite sequence $x_1 \to x_2 \to x_3 \to \ldots$ of elements x_i in M.

Lemma 2.2.1 (Basic Properties of Reduction)
(Noetherianity)
For all F: $\to_F$ *is noetherian.*
(Reduction Closure = Congruence)
For all F: $\equiv_F = \overset{*}{\leftrightarrow}_F$.
(Normal Form Algorithm)
There exists an algorithm NF *("Normal Form") such that for all* F, g:
(NF1) $g \overset{*}{\to}_F \mathrm{NF}(F,g)$,
(NF2) $\underline{\mathrm{NF}(F,g)}_F$.
(Cofactor Algorithm)

There exists an algorithm COF *("cofactors") such that for all F, g:*
COF(F, g) *is a sequence H of polynomials indexed by F satisfying*

$$g = \mathrm{NF}(F, g) + \textstyle\sum_{f \in F} H_f.f.$$

Note that, for fixed F, f, there may exist many different g such that $f \to_F^* g$ and $\underline{g}_F$ i. e. , in general, "normal forms for polynomials f are not unique modulo F". A normal form algorithm NF, by successive reduction steps, singles out one of these g for each F and f.

COF proceeds by "collecting" the multiples $a.u.h$ of polynomials $h \in F$ that are subtracted in the reduction steps when applying the normal form algorithm NF to g. Actually, COF can be required to satisfy additional properties, for examples, certain restrictions on the leading power products of the polynomials $H_f.f$.

2.3 Gröbner Bases and the Main Theorem

Definition 2.3.1 (Buchberger 1965, 1970)
F is a Gröbner basis (w. r. t. $\succ$) iff
"normal forms modulo F are unique", i.e.
for all f, g_1, g_2:
if $f \to_F^* g_1, f \to_F^* g_2, \underline{g_1}_F, \underline{g_2}_F$, then $g_1 = g_2$.

Note that $\to_F$ depends on the underlying admissible ordering $\succ$ on the power products. Therefore, also the definition of Gröbner basis depends on the underlying $\succ$. Whenever $\succ$ is clear from the context, we will not explicitly mention $\succ$. Gröbner bases whose polynomials are monic (i. e. have leading coefficient 1) and are in normalform modulo the remaining polynomials in the basis are called "*reduced Gröbner bases*". As we will see, Gröbner bases have a number of useful properties that establish easy algorithms for important problems in polynomial ideal theory. Therefore the main question is how Gröbner bases can be algorithmically constructed. The algorithm needs the concept of "*S-polynomials*". The S-polynomial of two polynomials f and g is the difference of certain multiples $u.f$ and $v.g$. For details see (Buchberger 1985). We use the notation

SP(f, g) the S-polynomial of f and g.

Theorem 2.3.1 (Main Theorem, Buchberger 1965, 1970)
(Algorithmic Characterization of Gröbner Bases)
F is a Gröbner basis iff
for all $f, g \in F$: $\mathrm{NF}(F, \mathrm{SP}(f, g)) = 0$.
(Algorithmic Construction of Gröbner Bases)
There exists an algorithm GB *such that for all F*
(GB1) Ideal(F) = Ideal(GB(F))
(GB2) GB(F) *is a (reduced) Gröbner basis.*

The proof of the (Algorithmic Characterization) is completely combinatorial and quite involved. The whole power of the Gröbner bases method is contained in this proof. The algorithm GB is based on the (Algorithmic Characterization), i. e. it involves successive computation of normal forms of S-polynomials. This algorithm is structurally simple. However, it is complex in terms of time and space consumed. In some sense, this is necessarily so because the problems that can be solved by the Gröbner bases method are intrinsically complex as has been shown by various authors. Still, the algorithm allows to tackle interesting and non-trivial practical problems for which no feasible solutions were known by other methods. Also, various theoretical and practical improvements of the algorithm have enhanced the scope of applicability.

2.4 The Gröbner Bases Algorithm in Software Systems

The Gröbner bases algorithm GB is available in almost all major computer algebra systems, notably in the SAC-2, SCRATCHPAD II, REDUCE, MAPLE, MACSYMA and muMATH systems. The introduction of (Buchberger, Collins, Loos 1982) contains the addresses of institutions from which these systems can be obtained. In these systems at least the algorithms SP, NF, (COF,) and GB are accessible to the user. In most systems, also a number of other auxiliary routines and variants of these basic algorithms are available and the user can experiment with different coefficient domains, admissible orderings and strategies for tuning the algorithms.

The implementations vary drastically in their efficiency mostly because of the varying amount of theory that has been taken into account. Also, computation time and space depends drastically on the admissible orderings used, on permutations of variables, on treating indeterminates as ring or field variables, on strategies for selecting pairs in the consideration of S-polynomials and on many other factors. Thus if one seriously considers solving problems of the type described in this paper one should try different systems and various orderings, strategies etc.

The rest of the paper is written with the goal in mind that the reader should be able to apply the methods as soon as he has access to an implementation of the basic algorithms NF, COF, SP, and GB viewed as "black boxes".

2.5 Properties of Gröbner Bases

In the following theorem we summarize the most important properties of Gröbner bases on which the algorithmic solution of many fundamental problems in polynomial ideal theory (algebraic geometry, non-linear computational geometry) can be based. Actually, not all of these properties are used in the later sections of the paper. However, since the results on Gröbner bases are quite scattered in the literature, the summary may help the reader who perhaps wants to try the Gröbner bases method on new problems. Many of the properties listed in the theorem were already proven in (Buchberger 1965, 1970). Actually the problems that can be solved with the (Residue Class Ring) properties were the starting point for Gröbner bases theory in (Buchberger 1965). The property (Elimination Ideals) is due to (Trinks 1978). The property (Inverse Mappings) is a recent contribution by (Van den Essen 1986) that solves a decision problem that has been open since 1939. (Algebraic Relations) and (Syzygies) seem to have been known already to (Spear 1977). However, it is hard to trace were the proofs appeared for the first time. More references are given in (Buchberger 1985). Most of the proofs of the properties below are immediate consequences of the definition of Gröbner bases, the property (Reduction Closure = Congruence), and some well known algebraic lemmas in polynomial ideal theory. The proofs of the properties (Syzygies) and (Inverse Mappings) are more involved. The existence of the algorithm GB based on the above Main Theorem is the crux for the algorithmic character of the properties.

In the following, let $K[x_1,\ldots,x_n]$ be arbitrary but fixed. F and G are used as typed variables for finite subsets of $K[x_1,\ldots,x_n]$. If not otherwise stated, $\succ$ is arbitrary. When we say "y is a new indeterminate" we mean that y is different from $x_1,\ldots,x_n$. By "F is solvable" we mean that there exists an n-tuple $(a_1,\ldots,a_n)$ of elements a_i in the algebraic closure $\overline{K}$ such that $f(a_1,\ldots,a_n)=0$ for all $f\in F$. Similarly, the expression "F has finitely many solutions" and similar expressions always refer to solutions over the algebraic closure of K.

Theorem 2.5.1 (General Properties of Gröbner Bases)

(Ideal Equality, Uniqueness of Reduced Gröbner Bases)

For all F, G: $\mathrm{Ideal}(F)=\mathrm{Ideal}(G)$ *iff* $\mathrm{GB}(F)=\mathrm{GB}(G)$.

(Idempotency of GB)

For all reduced Gröbner bases G: $\mathrm{GB}(G)=G$.

(Ideal Membership)

For all F, f: $f\in\mathrm{Ideal}(F)$ *iff* $\mathrm{NF}(\mathrm{GB}(F),f)=0$.

(Canonical Simplification)

For all F, f, g: $f\equiv_F g$ *iff* $\mathrm{NF}(\mathrm{GB}(F),f)=\mathrm{NF}(\mathrm{GB}(F),g)$.

(Radical Membership)

For all F, f:
$f\in\mathrm{Radical}(F)$ *iff* $1\in\mathrm{GB}(F\cup\{y.f-1\})$, *(where y is a new indeterminate).*

(Computation in Residue Class Rings)

For all F:

The residue class ring $K[x_1,\ldots,x_n]/\mathrm{Ideal}(F)$ is isomorphic to the algebraic structure whose carrier set is $\{f \mid \underline{f}_F\}$ and whose addition and multiplication operations, $\oplus$ and $\otimes$, are defined as follows:

$$f\oplus g:=\mathrm{NF}(\mathrm{GB}(F),f+g),$$
$$f\otimes g:=\mathrm{NF}(\mathrm{GB}(F),f.g).$$

(Note that the carrier set is a decidable set and $\oplus$ *and* $\otimes$ *are computable!).*

(Residue Class Ring, Vector Space Basis)

For all F:

The set $\{[u]_F \mid u \notin \mathrm{MLP}(\mathrm{GB}(F))\}$ *is a linearly independent basis for* $K[x_1, \ldots, x_n]/\mathrm{Ideal}(F)$ *considered as a vector space over K.*

(Residue Class Ring, Structure Constants)

For all F, u, v:
if $u, v \notin \mathrm{MLP}(\mathrm{GB}(F))$,
then $[u]_F.[v]_F = \sum_{w \notin \mathrm{MLP}(\mathrm{GB}(F))} a_w.[w]_F$,
where, for all $w, a_w := \mathrm{C}(\mathrm{NF}(\mathrm{GB}(F), u.v), w)$.

(The $a_w \in K$*, appearing in these representations of products of the basis elements as linear combinations of the basis elements are the "structure constants" of* $K[x_1, \ldots, x_n]/\mathrm{Ideal}(F)$ *considered as an associative algebra.)*

(Leading Power Products)

For all F: $\mathrm{MLP}(\mathrm{Ideal}(F)) = \mathrm{MLP}(\mathrm{GB}(F))$.

(Principal Ideal)

For all F:
$\mathrm{Ideal}(F)$ *is principal (i. e. has a one-element ideal basis)*
iff $\mathrm{GB}(F)$ *has exactly one element.*

(Trivial Ideal)

For all F: $\mathrm{Ideal}(F) = K[x_1, \ldots, x_n]$ *iff* $\mathrm{GB}(F) = \{1\}$.

(Solvability of Polynomial Equations)

For all F: F is solvable iff $1 \notin \mathrm{GB}(F)$.

(Finite Solvability of Polynomial Equations)

For all F:
F has only finitely many solutions iff
for all $1 \leq i \leq n$ *there exists an* $f \in \mathrm{GB}(F)$ *such that*
$\mathrm{LP}(f)$ *is a power of* x_i.

(Number of Solutions of Polynomial Equations)

For all F with finitely many solutions:

the number of solutions of F (with multiplicities and solutions at infinity) $=$
$=$ cardinality of $\{u \mid u \notin \mathrm{MLP}(\mathrm{GB}(F))\}$.

(Minimal Polynomial)

For all F and all finite sets U of power products:

There exists an $f \in \mathrm{Ideal}(F)$ in which only power products from U occur iff $\{\mathrm{NF}(\mathrm{GB}(F, u)) \mid u \in U\}$ is linearly dependent over K.

(By applying this property successively to the powers $1, x_i, x_i^2, x_i^3, \ldots$ one can algorithmically find, for example, the univariate polynomial in x_i of minimal degree in $\mathrm{Ideal}(F)$ if it exists. On this algorithm a general method for solving arbitrary system of polynomial equations can be based, see (Buchberger 1970), which works for arbitrary $\succ$ whereas the elimination method mentioned below works only for lexical orderings.)

(Syzygies)

Let F be a (reduced) Gröebner basis and define for all $f, g \in F$:

$P^{(f,g)} := \mathrm{COF}(F, \mathrm{SP}(f, g))$,
u and v such that $\mathrm{SP}(f,g) = u.f - v.g$,
$S^{(f,g)}$ is a sequence of polynomials indexed by F,
$S_f^{(f,g)} := u - P_f^{(f,g)}$,
$S_g^{(f,g)} := -v - P_g^{(f,g)}$,
$S_h^{(f,g)} := -P_h^{(f,g)}$, for all $h \in F - \{f, g\}$.

Then,

$\{S^{(f,g)} \mid f, g \in F\}$ is a set of generators for the $K[x_1, \ldots, x_n]$-module of all sequences H of polynomials (indexed by F) that are solutions ("syzygies") of the linear diophantine equation

$\sum_{h \in F} H_h . h = 0.$

(This solution method for linear diophantine equations over $K[x_1, \ldots, x_n]$ whose coefficients form a Gröbner basis F can be easily extended to the case of arbitrary F and to systems of linear diophantine equations, see (Buchberger 1985), (Winkler 1986)).

Theorem 2.5.2 (Properties of Gröbner Bases for Particular Orderings)

(Hilbert Function)

Let $\succ$ be a total degree ordering.
Then, for all F:

The value $\mathrm{H}(d, F)$ of the Hilbert function for d and F, i. e. the number of modulo $\mathrm{Ideal}(F)$ linearly independent polynomials in $K[x_1, \ldots, x_n]$of degree $\leq d$, is equal to

$$\binom{d+n}{n} \text{ - cardinality of } \{u \text{ of degree } \leq d \mid u \notin \mathrm{MLP}(\mathrm{GB}(F))\}.$$

(Elimination Ideals, Solution of Polynomial Equations)

Let $\succ$ be the lexical ordering defined by $x_1 \prec x_2 \prec \ldots \prec x_1$.
Then, for all F, $1 \leq i \leq n$:

The set $\mathrm{GB}(F) \cap K[x_1, \ldots, x_i])$ is a (reduced) Gröbner basis for the "i-th elimination ideal" generated by F, i. e. for $\mathrm{Ideal}_{K[x_1,\ldots,x_n]}(F) \cap K[x_1, \ldots, x_i]$.

(This property leads immediately to a general solution method, by "successive substitution", for arbitrary systems of polynomial equations with finitely many solutions, which is formally described in (Buchberger 1985). We will demonstrate this method in the examples in the application section of this paper.)

(Continuation of Partial Solutions)

Let $\succ$ be a lexical ordering.
For all F:

If $F := \{f_1, \ldots, f_k\}$ is a Gröbner basis with respect to $\succ$, $f_1 \prec \cdots \prec f_k$, and $f_1, \ldots, f_l (1 \leq l \leq k)$ are exactly those polynomials in F that depend only on the indeterminates $x_1, \ldots, x_i$, then every common solution $(a_1, \ldots, a_i)$ of $\{f_1, \ldots, f_l\}$ can be continued to a common solution $(a_1, \ldots, a_n)$ of F. (For a correct statement of this property some terminology about solutions at infinity would be necessary.)

(Independent Variables Modulo an Ideal)

For all F and $1 < i_1 < \ldots < i_m < n$:

The indeterminates $x_{i_1}, \ldots, x_{i_m}$ are independent modulo $\mathrm{Ideal}(F)$ (i. e. there is no polynomial in $\mathrm{Ideal}(F)$ that depends only on $x_{i_1}, \ldots, x_{i_m}$) iff $\mathrm{GB}(F) \cap K[x_{i_1}, \ldots, x_{i_m}] = \{0\}$, where the $\succ$ used must be a lexical ordering satisfying $x_{i_1} \prec \cdots \prec x_{i_m} \prec$ all other indeterminates. (This property yields immediately an algorithm for determining the dimension of a polynomial ideal (algebraic variety).)

(Ideal Intersection)

Let $\succ$ be the lexical ordering defined by $x_1 \prec x_2 \prec \ldots \prec x_n \prec y$, y a new variable.
Then, for all F, G:

$\mathrm{GB}(\{y.f \mid f \in F\} \cup \{(y-1).g \mid g \in G\}) \cap K[x_1, \ldots, x_n]$
is a (reduced) Gröbner basis for $\mathrm{Ideal}(F) \cap \mathrm{Ideal}(G)$.

(This property yields also an algorithm for quotients of finitely generated ideals because the determination of such quotients can be reduced to the determination of intersections.)

(Algebraic Relations)

For all F:

Let $F = \{f_1, \ldots, f_m\}$, let $y_1, \ldots, y_m$ be new indeterminates and let $\succ$ be the lexical ordering defined by $y_1 \prec \ldots \prec y_m \prec x_1 \prec \ldots \prec x_n$. Then, $\mathrm{GB}(\{y_1 - f_1, \ldots, y_m - f_m\}) \cap K[y_1, \ldots, y_m]$ *is a (reduced) Gröbner basis for the "ideal of algebraic relations" over F, i. e. for the set $\{g \in K[y_1, \ldots, y_m] \mid g(f_1, \ldots, f_m) = 0\}$.*

(Inverse Mapping)

For all F:

Let $F = \{f_1, \ldots, f_n\}$, let $y_1, \ldots, y_n$ be new indeterminates and let $\succ$ be the lexical ordering defined by $y_1 \prec \ldots \prec y_n \prec x_1 \prec \ldots \prec x_n$. Then, the mapping from $\overline{K}^n$ into $\overline{K}^n$ defined by F is bijective iff $\mathrm{GB}(\{y_1 - f_1, \ldots, y_n - f_n\})$ *has the form $\{x_1 - g_1, \ldots, x_1 - g_n\}$ for certain $g_j \in K[y_1, \ldots, y_n]$.*

The properties stated in the above theorem can be read as the algorithmic solution of certain problems specified by polynomial sets F. Each of these "algorithms" requires that, for solving the problem for an arbitrary F, one first transforms F into the corresponding (reduced) Gröbner basis $\mathrm{GB}(F)$ and then performs some algorithmic actions on $\mathrm{GB}(F)$. For example, for the decision problem "$f \equiv_F g$?", (Canonical Simplification) requires that one first transforms F into $\mathrm{GB}(F)$ and then checks, by applying algorithm NF, whether or not the normal forms of f and g are identical modulo $\mathrm{GB}(F)$. Actually, most of the above properties (algorithms) are correct also if, instead of transforming F into a corresponding *reduced* Gröbner basis, one transforms F into an *arbitrary* equivalent Gröbner basis G. (We say "F is equivalent to G" iff $\mathrm{Ideal}(F) = \mathrm{Ideal}(G)$.) In practice, however, this makes very little difference because the computation of Gröbner bases is not significantly easier if one relaxes the requirement that the Gröbner basis must be reduced.

Alternatively, by (Idempotency of GB), the properties stated in the above theorem can also be read as properties of (reduced) Gröbner bases — and algorithms for solving problems for (reduced) Gröbner bases. For example, introducing the additional assumption that F is a (reduced) Gröbner basis, (Canonical Simplification) reads as follows:

For all (reduced) Gröbner bases F, and polynomials f, g:
$f \equiv_F g$ *iff* $\mathrm{NF}(F, f) = \mathrm{NF}(F, g)$.

Some of the properties stated in the above theorem are characteristic for Gröbner bases, i. e. if the property holds for a set F then F is a Gröbner basis. For example, (Leading Power Products) is a characteristic property, i. e. if $\mathrm{MLP}(\mathrm{Ideal}(F)) = \mathrm{MLP}(F)$ then F is a Gröbner basis.

Let us carry through one more exercise for reading the above properties as algorithms. For deciding whether

(Question)
for all $a_1, \ldots, a_n \in \overline{K}$,
for which $f_1(a_1, \ldots, a_n) = \cdots = f_m(a_1, \ldots, a_n) = 0$,
also $g(a_1, \ldots, a_n) = 0$,

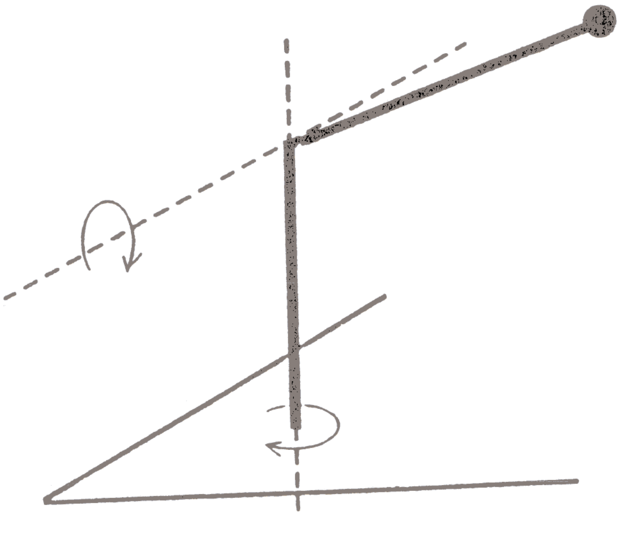
i. e. for deciding whether $g \in \mathrm{Radical}(\{f_1, \ldots, f_m\})$, because of (Radical Membership), it suffices to perform the following steps:

1. Compute the (reduced) Gröbner basis G for $\{f_1, \ldots, f_m, y {\cdot} g - 1\})$, where y is a new indeterminate.
2. The (Question) has a positive answer iff $1 \in G$.

3 Application: Inverse Robot Kinematics

The problem of inverse robot kinematics is the problem of determining, for a given robot, the distances at the prismatic joints and the angles at the revolute joints that will result in a given position and orientation of the end-effector. The mathematical description of this problem leads to a system of multivariate polynomial equations (after representing angles α by their sine and cosine and adding $\sin^2 \alpha + \cos^2 \alpha = 1$ to the set of equations), see (Paul 1981).

Let us consider, for example, the following robot having two revolute joints (two "degrees of freedom").



We introduce the following variables:

l_1, l_2	lengths of the two robot arms
px, py, pz	x-, $y-$, and z-coordinate of the position of the end-effector
ϕ, θ, ψ	Euler angles of the orientation of the end effector (Euler angles are one way of describing orientation)
δ_1, δ_2	angles describing rotation at the revolute joints

We introduce the sines and cosines of the angles occuring in the above description as separate variables:

s_1, c_1	sine and cosine of δ_1
s_2, c_2	sine and cosine of δ_2
sf, cf	sine and cosine of ϕ
st, ct	sine and cosine of θ
sp, cp	sine and cosine of ψ

The interrelation of the physical entities described by the above variables is expressed in the following system of equations:

$$\begin{aligned}
&c_1 \cdot c_2 - cf \cdot ct \cdot cp + sf \cdot sp = 0,\\
&s_1 \cdot c_2 - sf \cdot ct \cdot cp - cf \cdot sp = 0,\\
&s_2 + st \cdot cp = 0,\\
&-c_1 \cdot s_2 - cf \cdot ct \cdot sp + sf \cdot cp = 0,\\
&-s_1 \cdot s_2 + sf \cdot ct \cdot sp - cf \cdot cp = 0,\\
&c_2 - st \cdot sp = 0,\\
&s_1 - cf \cdot st = 0,\\
&-c_1 - sf \cdot st = 0,\\
&ct = 0,\\
&l_2 \cdot c_1 \cdot c_2 - px = 0,\\
&l_2 \cdot s_1 \cdot c_2 - py = 0,\\
&l_2 \cdot s_2 + l_1 - pz = 0,\\
&c_1^2 + s_1^2 - 1 = 0,\\
&c_2^2 + s_2^2 - 1 = 0,\\
&cf^2 + sf^2 - 1 = 0,\\
&ct^2 + st^2 - 1 = 0,\\
&cp^2 + sp^2 - 1 = 0.
\end{aligned}$$

Let us call those variables that describe the geometrical realization of the robot "geometrical variables" (for example, the variables l_1, l_2). Let us also call those variables that describe position and orientation of the end-effector shortly "position variables" ($px, \ldots, sf, cf, \ldots$). The other variables ($s_1, c_1, \ldots$) are the "joint variables".

In the case of more complicated robots (with six degrees of freedom), one can specify values for the geometrical variables and the position variables and, with certain restrictions, will always be able to determine appropriate values of the joint variables that yield the given position and orientation of the end-effector. In the above example robot, with only two degrees of freedom however one can only independently choose the value of two position variables, for example px and pz. The value of all the other variables, notably of the other position variables $py, sf, cf, \ldots$, and the joint variables will then be determined by the above system of algebraic equations.

The problem can be considered in three different versions of increasing generality.

(Real Time Version)

- The value of the geometrical variables are numerically given.
- The value of those position variables that can be independently chosen (e. g. px, pz) are numerically given.
- The solution of the problem consists in determining appropriate numerical values for the (remaining position variables and) the joint variables.

(Off-Line Version, Concrete Robot)

- The value of the geometrical variables are numerically given.
- The value of those position variables that can be independently chosen are left open as *parameters*.
- By a "solution of the problem", in this version, one means symbolic expressions involving the position parameters that describe, in "closed form", the dependence of the (remaining position variables and) the joint variables from the position parameters. Of course, a "symbolic closed form solution" of this kind will not always be possible. It is possible for certain classes of robots, see (Paul 1981), and it is possible in a modified sense also in the general case by using Gröbner bases.

(Off-Line Version, Robot Class)

- The value of the geometrical variables are left open as *parameters*.
- The value of those position variables that can be independently chosen are left open as *parameters*.
- By a "solution of the problem", in this version, one means symbolic expressions involving the geometrical and the position parameters that describe, in "closed form", the dependence of the (remaining position variables and) the joint variables on the geometrical *and* position parameters.. A "symbolic closed form solution" in this general sense is even more difficult. Again, it is possible for certain classes of robots and, as we shall see, it is possible in a modified sense also in the general case by using Gröbner bases.

A symbolic solution of the inverse kinematics problem in the (Off-Line Version), can be contrasted to a numerical approach:

(Symbolic Approach)

- Derivation of the symbolic expressions for the solution of the problem in the (Off-Line Version).
- Numerical specification of the parameters.
- Numerical evaluation of the symbolic expressions using the numerical values of the parameters.

(Numerical Approach)

- Numerical specification of the parameters.
- Solution of the problem in the (Real-Time Version) by numerical iteration methods.

It is clear that a symbolic solution of the problem in the (Off-Line Version) can have practical advantages over the purely numerical approach (as long as the resulting symbolic expressions describing the solutions are not too complicated) because the numerical evaluation of the symbolic solution expressions in real-time situations may be faster than a direct iterative numerical solution of the (Real Time Version) of the problem. Also, of course, the symbolic solution may give "insight" into the problem that can not be gained by a numerical solution.

For the above example, we show the solution of the problem in the (Off-Line Version, Roboter Class) by using Gröbner bases. In this version, the geometrical variables l_1, l_2 and the position variables px, pz are considered as symbolic parameters.

The solution method uses property (Elimination Ideals) of Gröbner bases. This property, read as an algorithm, tells us that we first have to compute the Gröbner bases of the set F of input polynomials. Since l_1, l_2, px, pz are to be treated as symbolic parameters, we work over the field $\mathbf{Q}(l_1, l_2, px, pz)$ as coefficient field. This is perfectly possible, because the Gröbner bases method works over arbitrary fields (whose arithmetic is algorithmic). Furthermore, we must specify an ordering on the remaining variables, for example $c_1 \prec c_2 \prec s_1 \prec s_2 \prec py \prec cf \prec ct \prec cp \prec sf \prec st \prec sp$. These variables are treated as ring variables, i.e. the Gröbner basis will be computed considering the input polynomials as polynomials in the ring $\mathbf{Q}(l_1, l_2, px, pz)[c_1, \ldots, sp]$. The resulting Gröbner basis has the following form:

$$
\begin{aligned}
& c_1^2 + \frac{px^2}{pz^2 - 2\cdot l_1\cdot pz - l_2^2 + l_1^2} = 0,\\
& c_2 + \frac{pz^2 - 2\cdot l_1\cdot pz - l_2^2 + l_1^2}{l_2} \cdot px \cdot c_1 = 0,\\
& s_1^2 - \frac{pz^2 - 2\cdot l_1\cdot pz + px^2 - l_2^2 + l_1^2}{pz^2 - 2\cdot l_1\cdot pz - l_2^2 + l_1^2} = 0,\\
& s_2 - \frac{pz - l_1}{l_2} = 0,\\
& py + \frac{pz^2 - 2\cdot l_1\cdot pz - l_2^2 + l_1^2}{px} \cdot c_1 \cdot s_1 = 0,\\
& cf^2 - \frac{pz^2 - 2\cdot l_1\cdot pz + px^2 - l_2^2 + l_1^2}{pz^2 - 2\cdot l_1\cdot pz - l_2^2 + l_1^2} = 0,\\
& ct = 0,\\
& cp + \frac{pz^3 - 3\cdot l_1\cdot pz^2 - l_2^2\cdot pz + 3\cdot l_1^2\cdot pz + l_1\cdot l_2^2 - l_1^3}{l_2\cdot pz^2 - 2\cdot l_1\cdot l_2\cdot pz + l_2\cdot px^2 - l_2^3 + l_1^2\cdot l_2} \cdot s_1 \cdot cf = 0,\\
& sf + \frac{pz^2 - 2\cdot l_1\cdot pz - l_2^2 + l_1^2}{pz^2 - 2\cdot l_1\cdot pz + px^2 - l_2^2 + l_1^2} \cdot c_1 \cdot s_1 \cdot cf = 0,\\
& st + \frac{pz^2 - 2\cdot l_1\cdot pz - l_2^2 + l_1^2}{pz^2 - 2\cdot l_1\cdot pz + px^2 - l_2^2 + l_1^2} \cdot s_1 \cdot cf = 0,\\
& sp + \frac{pz^4 - 4\cdot l_1\cdot pz^3 - 2\cdot l_2^2\cdot pz^2 + 6\cdot l_1^2\cdot pz^2 + 4\cdot l_1\cdot l_2^2\cdot pz - 4\cdot l_1^3\cdot pz + l_2^4 - 2\cdot l_1^2\cdot l_2^2 + l_1^4}{l_2\cdot px\cdot pz^2 - 2\cdot l_1\cdot l_2\cdot px\cdot pz + l_2\cdot px^3 - l_2^3\cdot px + l_1^2\cdot l_2\cdot px} \cdot c_1 \cdot s_1 \cdot cf = 0.
\end{aligned}
$$

The above Gröbner basis has a remarkable structure:

- The geometrical parameters l_1 and l_2 and the position parameters px and pz are still available as symmbolic parameters in the polynomials of the Gröbner basis. Thus, the system is still "general". The Gröbner basis is in "closed form".

- In accordance with property (Elimination Ideals), the system is "triangularized". In this example, this means that the first polynomial of the basis depends only on c_1, the second on $c_1, c2$, the third on $c_1, c_2, s_1, \ldots$. After substitution of numerical values for the parameters l_1, l_2, px, pz, we can therefore numerically determine the possible values for c_1 from the first equation then, for each of the values of c_1, determine the value of c_2 from the second equation then, for each of the values of c_1, c_2, determine the value of s_1 from the third equation etc.

- Actually, the degrees of the polynomials in this basis are quite low. This is in general not true for the first polynomial in Gröbner bases. The first polynomial, which, in case the solution set is finite, is always univariate, tends to have quite a high degree in general. The degrees of the other polynomials, however, tend to be very low (most times even linear) also in the general case because the polynomial sets describing realistic physical or geometrical situations often define prime ideals, for which linearity in the second, third ... variable can be proven theoretically. This phenomenon needs closer study, however. For numerical practice, low degrees in the second, third ... variable implies that numerical errors from the determination of the value of the first value will not drastically accumulate. In the case where the second, third ... equation is linear, the Gröbner basis has the form $\{p_1(x_1), x_2 - p_2(x_1), \ldots, x_n - p_n(x_1)\}$. In this case, the errors introduced by the numerical solution of p_1 will not accumulate at all.

- The above method of numerical backward substitution based on the Gröbner basis, by property (Elimination Ideals), is guaranteed to yield *all* (real and complex) solutions of the system.

- Again by (Elimination Ideals), *no "extraneous"* solutions of the system are produced. (Other algebraic methods, for example the resultant method, may produce extraneous solutions.)

The above Gröbner basis was produced in 62 sec on an IBM 4341 using an implementation of the Gröbner basis method by R. Gebauer and H. Kredel in the SAC-2 computer algebra system. The computation time is increasing drastically when more complicated robot types are investigated. We are far from being able to treat the most general robot of six degrees of freedom. However, so far, only very little research effort has been dedicated to this possible application of Gröbner bases. Using the special structure of the problem it may well be that more theoretical results can be derived that allow to drastically speed up the general algorithm in this particular application.