COMPUTABLE ALGEBRA*

C. F. Miller III
University of Melbourne

Computable algebra is concerned with algorithmic problems about algebraic systems. Algorithmic problems about group theory are related to topology; for example, the word problem and the isomorphism problem. I will talk about some alogarithmic problems arising from commutative algebra. More specifically, about rings $\mathbb{Z}$, $\mathbb{Q}$, k (a field), polynomial rings k[t] and modules. These are the basic ingredients of algebraic number theory and algebraic geometry.

Mathematicians have long been interested in algorithmic problems. Kronecker proved that the field $\mathbb{Q}$ of rational numbers has a splitting algorithm (see van der Waerden's book "Modern Algebra", Vol. 1). We will deal only with *recursive fields*, i.e. fields k which are given as a recursive set of elements such that the field operations +, · are computable (recursive). There is, roughly speaking, a program which recognizes the elements of k (for example, recognizing the prime numbers). We say that k has a splitting algorithm if there is an effective procedure to determine, for an arbitrary $f \in k[t]$, whether or not f is irreducible, i.e. whether $f = gh$ nontrivially, where $g, h \in k[t]$.

It is a fact that if k is a recursive field, then the set

$$\{ f \in k[t] \mid f = gh \text{ nontrivially} \}$$

is *recursively enumerable* (r.e.) (in other words, it can be, in principle, listed by some machine). Now k has a splitting algorithm $\Leftrightarrow$ $\{ f \in k[t] \mid f \text{ is irreducible} \}$ is recursive $\Leftrightarrow$ $\{ f \in k[t] \mid f \text{ is irreducible} \}$ is r.e. However, recursive theory tells us that there exist r.e. sets which are not recursive. Thus not every recursive field has a splitting algorithm.

Example. Let S be an r.e., non-recursive set of prime numbers. Form $k = \mathbb{Q}$.(roots of $t^2 - p$ for all $p \in S$). This is a recursive field but the polynomial $t^2 - q$ for q a prime is reducible over k $\Leftrightarrow$ $q \in S$. This is undecidable.

Theorem (Rabin). If k is a recursive field, then k has a recursive algebraic closure (K, φ), i.e. K is a recursive algebraically closed field and $\varphi : k \to K$ is a recursive injection and K is algebraically closed over φ (K).

We say that a recursive field k has the following type of algorithm if the corresponding set is recursive :

* *Invited lecture organised by the Department of Mathematics, National University of Singapore, and Singapore Mathematical Society on 14 October 1983.*

Algorithm	Recursive property
Separable splitting algorithm	$\{ f \in k[t] \mid f$ is separable & irreducible $\}$
Root algorithm	$\{ f \in k[t] \mid f(\alpha) = 0$ for some $\alpha \in k \}$
p – th root algorithm	$\{ \alpha \in k \mid \beta^p = \alpha$ for some $\beta \in k \}$

The following theorem is due to Rabin, Fröhlich and Shepherdson, van der Waerden, Rick Smith.

Theorem. Let k be a recursive field and (K, φ) a recursive algebraic closure of k. Then the following are equivalent.

(1) k has a splitting algorithm.
(2) The function $K \to \mathbb{Z}$ defined by $\theta \to [\ \varphi(k)(\theta) : \varphi(k)\ ]$ is recursive.
(3) k has a root algorithm.
(4) $\varphi(k)$ is a recursive subset of K.
(5) k has a separable splitting algorithm and a p-th root algorithm where p = char k.
(6) Every finite separable extension of k has a splitting algorithm.

Theorem (Metakides and Nerode, 1979). k has a separable splitting algorithm $\Leftrightarrow$ k has a recursively unique algebraic closure (i.e. any two algebraic closures are recursively isomorphic by a recursive k - isomorphism).

Theorem (Steinitz). Every field has a transcendence base.

A field of characteristic O would look like:

$$\begin{tikzcd}
k \arrow[d, dash, "\text{algebraic}"'] \\
F = \mathbb{Q}(t_1, t_2, \ldots) \arrow[d, dash, "\text{purely transcendental}"'] \\
\mathbb{Q}
\end{tikzcd}$$

Theorem (Metakides and Nerode). Let k be a recursive field. Then there exists a recursive algebraically closed field K over k of transcendence degree $\aleph_0$ such that every r.e. algebraically independent set in K over k is finite.

Let R be a commutative ring with 1. Assume R is (right) Noetherian, i.e. the right ideals are finitely generated, or equivalently, R satisfies the ascending chain condition on right ideals. We also assume R is a recursive ring (i.e. computable ring); for example, $\mathbb{Z}$, $\mathbb{Q}$, k (recursive field).

The Hilbert basis theorem states that if R is a commutatvie Noetherian ring, then so is R[t]. In particular, k[t] and $k[t_1, \ldots\ldots, t_n]$ are Noetherian.

Let M be a finitely generated R-module with generators $a_1, a_2 \ldots a_n$ say. A *word* in the generators of M is an R-linear combination of the a_i :

$$w = a_1 r_1 + a_2 r_2 + \ldots a_n r_n ,$$

where $r_1, \ldots, r_n \in R$. Let F be the free module on $a_1, \ldots, a_n$. Then there is a map $\varphi : F \to M$ from F onto M, where $K = \ker \varphi$ is a finitely generated submodule of F. Suppose K is generated by $w_1, \ldots w_q$, (words in F). We can express M as a presentation

$$\begin{aligned} M &= \langle a_1, \ldots, a_n \mid w_1(a_1, \ldots, a_n) = 0, \ldots, w_q(a_1, \ldots, a_n) = 0 \rangle \\ &= F / K. \end{aligned}$$

We say that M is submodule computable if there exist recursive procedures which when applied to a finite set $\{ v_1, \ldots, v_p \}$ of words in M yield

(1) a finite presentation of the submodule $L = \mathrm{mod}_R (v_1, \ldots, v_n)$ generated by $\{ v_1, \ldots, v_q \}$, and

(2) an algorithm to decide membership in L, i.e. whether $y \in L$ for arbitrary $y \in M$.

Note that $y \in L \Leftrightarrow y \underset{M/L}{\equiv} 0$, so this is deciding which words are 0 in some other module.

Example of submodule computable rings are $\mathbb{Z}$, $\mathbb{Q}$, k (recursive fields).

R is submodule computable if every finitely presented R-module is submodule computable.

Lemma. Let R be a computable, right-Noetherian ring. Then R is submodule computable $\Leftrightarrow$ finitely generated free R-modules F are submodule computable in their standard presentation $F = \langle z_1, \ldots, z_n \mid \phi \rangle$.

For the free module $F = (a_1 \ldots a_n \mid \phi)$, submodule computability translates as follows;

finite presentation of $L = \mathrm{mod}_R (v_1, \ldots, v_n)$ $\longleftrightarrow$ Find a set of generators for the solution space of the homogeneous system

$$\left(r_{ij} \right) \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix},$$

Decide whether $u \in L$ $\longleftrightarrow$ Determine if the inhomogeneous system

$$\left(r_{ij} \right) \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix}$$

has a solution,

where $v_i = a_1 r_{1i} + a_2 r_{2i} + \ldots + a_n r_{ni}$, $i = 1, \ldots, n$, and $u = a_1 u_1 + \ldots + a_n u_n$

Theorem (Effective Hilbert Basis Theorem). Let R be a submodule computable ring. Then the polynomial ring R[t] is submodule computable.

Lemma. If R is submodule computable, and I a 2-sided ideal in R, then R/I is submodule computable.

Corollary. If R is a commutative submodule computable ring and A is a finitely generated commutative R-algebra, then A is submodule computable.

Generally, if R is submodule computable, we can recognize homomorphisms between R-modules and check whether they are monic, epic or isomorphic. If A is a submodule computable algebra over R, then we can give a finite presentation for the subalgebras of A generated by a finite set.

There are different views on the concept of an"effective" algorithm. The "constructionist" view is, however, more demanding. As examples of some types of algorithms, we mention the following.

Theorem. If A is a finitely presented commutative ring, then there exists an algorithm to decide if any finitely presented A-module M is finitely generated as an abelian group.

Corollary. There exists an algorithm to determine whether a finitely presented group is polycyclic.

Theorem. If A is a countable commutative ring, then GL (n, A) can be embedded in a finitely presented group.

Theorem. If R is a submodule computable ring and M, N are finitely presented modules over R, then there is an algorithm to compute $\mathrm{Ext}^n_R(M,N)$, $\mathrm{Tor}^R_n(M,N)$ and to recognize split-exact sequences

$$O \to N \to L \to M \to O.$$