Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition--Preliminary Report

George E. Collins*
University of Wisconsin

1. Introduction. Tarski in 1948, [18] published a quantifier elimination method for the elementary theory of real closed fields (which he had discovered in 1930). As noted by Tarski, any quantifier elimination method for this theory also provides a decision method, which enables one to decide whether any sentence of the theory is true or false. Since many important and difficult mathematical problems can be expressed in this theory, any computationally feasible quantifier elimination algorithm would be of utmost significance.

However, it became apparent that Tarski's method required too much computation to be practical except for quite trivial problems. Seidenberg in 1954, [17], described another method which he thought would be more efficient. A third method was published by Cohen in 1969, [3].

In the following I will describe in detail a new method which I discovered in February 1973 and presented in a seminar at Stanford University. A brief abstract [7] of these results was presented at a Carnegie-Mellon symposium in May, 1973. This new method is much more efficient than the previous methods, and therefore offers renewed hope of practical applicability.

In fact, it can be shown that, for a prenex input formula ϕ, the maximum computing time of this new method is dominated, in the sense of [5], by $(mn)^{k^r} d^k$, where r is the number of free and bound variables in ϕ, m is the number of polynomials occurring in ϕ, n is the maximum degree of any such polynomial in any variable, d is the maximum length of any integer coefficient of any such polynomial, and k is some constant. Thus, for fixed r, the computing time is dominated by a polynomial function $P_r(m,n,d)$. In contrast, it can be shown that the maximum computing times of the methods of Tarski and Seidenberg are exponential in both m and n for every fixed r, including even $r = 1$, and this is likely the case for Cohen's method also. (In fact, Cohen's method is presumably not intended to be efficient.)

Fischer and Rabin have recently shown, [9], that every decision method (deterministic or nondeterministic) for the first order theory of the real numbers, a fortiori for the elementary theory of real closed field, has a maximum computing which dominates 2^{c^N} where N is the length of the input formula and c is some positive constant. Since $m,n,d,r \le N$, the method we describe here has a computing time dominated by $N^{k^N} \le 2^{2^{kN}}$.

In a letter from Leonard Monk received April 26, 1974 I have been informed that he and R. Solovay have found a decision method (but not a quantifier elimination) with a time bound of $2^{2^{hN}}$.

In Section 2 we will make some definitions and state, without proof, several theorems which provide the mathematical foundations of the quantifier elimination algorithm. Section 2 also specifies the subalgorithms which are required for performing various algebraic operations, including especially calculations with real algebraic numbers. Section 3 describes in detail the main algorithms--decomposition, definition and evaluation--which constitute the quantifier elimination algorithm. Section 4 provides a brief and incomplete indication of the derivation of computing time bounds for the algorithm. Section 5 contains concluding remarks. This paper is a preliminary report; a more complete exposition will appear elsewhere, [8].

2. Preliminaries. In this section we make some needed definitions, state the basic theorems which provide a foundation for the quantifier elimination algorithm to be presented in Section 3, and define and discuss the main subalgorithms which will be required.

By an integral polynomial in r variables we shall mean any element of the ring $I[x_1,\ldots,x_r]$, where I is the ring of the rational integers. As observed by Tarski, any atomic formula of elementary algebra can be expressed in one of the two forms $A = 0$, $A > 0$, where A is an integral polynomial. Also, any quantifier-free formula can be easily expressed in disjunctive normal form as a disjunction of conjunctions of atomic formulas of these two types. However, for the quantifier elimination algorithm to be presented in this paper, there is no reason to be so restrictive, and we define a standard atomic formula as a formula of one of the six forms $A = 0$, $A > 0$, $A < 0$, $A \neq 0$, $A \geq 0$ and $A \leq 0$. A standard formula is any formula which can be constructed from standard atomic formulas using propositional connectives

* The investigation reported in this paper was partially supported by NSF Grant GJ-30125X.

and quantifiers. A *standard prenex formula* is a standard formula of the form

$$(Q_kx_k)(Q_{k+1}x_{k+1})\cdots(Q_rx_r)\phi(x_1,\ldots,x_r), \qquad (1)$$

where $\phi(x_1,\ldots,x_r)$ is a quantifier-free standard formula, $1 \le k \le r$, and each (Q_ix_i) is either an existential quantifier $(\exists x_i)$ or a universal quantifier $(\forall x_i)$.

The variables x_i range over the ordered field R of all real numbers, or over any other real closed field. For additional background information on elementary algebra, the reader is referred to Tarski's excellent monograph, [18], and van der Waerden, [19], has an excellent chapter on real closed fields.

The quantifier elimination algorithm to be described in the next section accepts as input any standard prenex formula of the form (1), with $1 \le k \le r$, and produces as output an equivalent standard quantifier-free formula $\psi(x_1,\ldots,x_{k-1})$.

$\mathcal{R}$ will denote an arbitrary commutative ring with identity. Unless otherwise specified, we will always regard a polynomial $A(x_1,\ldots,x_r) \in \mathcal{R}[x_1,\ldots,x_r]$ as an element of $\mathcal{R}[x_1,\ldots,x_{r-1}][x_r]$; that is, A is regarded as a polynomial in its *main variable*, x_r, with coefficients in the polynomial ring $\mathcal{R}[x_1,\ldots,x_{r-1}]$. Thus, for example, the *leading coefficient* of A, denoted by ldcf(A), is an element of $\mathcal{R}[x_1,\ldots,x_{r-1}]$. Similarly, deg(A) denotes the degree of A in x_r. If $A(x_1,\ldots,x_r) = \Sigma_{i=0}^{n}A_i(x_1,\ldots,x_{r-1})\cdot x_r^i$ and $\deg(A) = n$, then $\mathrm{ldcf}(A) = A_n$ and $\mathrm{ldt}(A) = A_n(x_1,\ldots,x_{r-1})\cdot x_r^n$, the *leading term* of A. Following Tarski, red(A), the *reductum* of A is the difference A-ldt(A). By convention, $\deg(0) = \mathrm{ldcf}(0) = 0$, and hence also $\mathrm{ldt}(0) = \mathrm{red}(0) = 0$. A' will denote the derivative of A.

R^k will denote the k=fold Cartesian product $R \times \cdots \times R$, $k \ge 1$. If f and g are real-valued functions defined on a set $S \subseteq R^k$, we write *$f > 0$ on S* in case $f(x) > 0$ for all $x \in S$, *$f = 0$ on S* in case $f(x) = 0$ for all $x \in S$; $f < 0$ on S, $f \ne 0$ on S, $f < g$ on S and other such relations are similarly defined. We say that *f is invariant on S* in case $f > 0$ on S, $f = 0$ on S, or $f < 0$ on S. These definitions are also applied to real polynomials, which may be regarded as real-valued functions.

Let $A(x_1,\ldots,x_r)$ be a real polynomial, $r \ge 2$, S a subset of R^{r-1}. We will say that $f_1,f_2,\ldots,f_k$ *delineate the real roots of A on S* in case the following conditions are all satisfied:

(1) $f_1,\ldots,f_k$ are continuous real-valued functions defined on S.

(2) $f_1 < f_2 < \cdots < f_k$ on S.

(3) There is a positive integer e_i such that $f_i(a_1,\ldots,a_{r-1})$ is a root of $A(a_1,\ldots,a_{r-1},x)$ of multiplicity e_i for $(a_1,\ldots,a_{r-1}) \in S$ and $1 \le i \le k$.

(4) If $(a_1,\ldots,a_{r-1}) \in S$, $b \in R$ and $A(a_1,\ldots,a_{r-1},b) = 0$ then, for some i, $1 \le i \le k$ and $b = f_i(a_1,\ldots,a_{r-1})$.

e_i will be called the multiplicity of f_i.

The following basic theorem gives sufficient conditions for the delineability of the real roots of a real polynomial A on a set S.

Theorem 1. Let $A(x_1,\ldots,x_r)$ be a real polynomial of degree $n > 0$, with $r \ge 2$. Let S be a connected subset of R^{r-1}. Assume $\mathrm{ldcf}(A) \ne 0$ on S and that the number of distinct real and complex roots of A is invariant on S; i.e., for some $m \ge 0$, $A(a_1,\ldots,a_{r-1},x)$ has exactly m distinct real and complex roots for all $(a_1,\ldots,a_{r-1}) \in S$. Then there are functions $f_1,\ldots,f_k$ which delineate the real roots of A on S.

We say that the polynomials $A;B \in \mathcal{R}[x]$ are *similar*, and write $A \approx B$, in case there exist *non-zero* $a,b \in \mathcal{R}$ such that $aA = bB$.

We define $\mathrm{red}^k(A)$, the $k^{\underline{th}}$ reductum of the polynomial A, for $k \ge 0$, by induction on k as follows: $\mathrm{red}^0(A) = A$ and $\mathrm{red}^{k+1}(A) = \mathrm{red}(\mathrm{red}^k(A))$ for $k \ge 0$. We say that B is *a reductum* of A in case $B = \mathrm{red}^k(A)$ for some $k \ge 0$.

We repeat some definitions from [4]. Let A and B be polynomials over $\mathcal{R}$ with $\deg(A) = m$ and $\deg(B) = n$. The *Sylvester matrix* of A and B is the $m + n$ by $m + n$ matrix M whose successive rows contain the coefficients of the polynomials $x^{n-1}A(x),\ldots,A(x),x^{n-1}B(x),\ldots,xB(x),B(x)$, with the coefficients of x^i occurring in column $m + n - i$. We allow either $m = 0$ or $n = 0$. As is well known, res(A,B), the resultant of A and B, is det(M), the determinant of M. (We adopt the convention $\det(N) = 0$ in case N is a zero by zero determinant.) For $0 \le i \le j \le \min(m,n)$ let $M_{j,i}$ be the matrix obtained from M by deleting the last j rows of A coefficients, the last j rows of B coefficients, and all of the last $2j + 1$ columns except column $m + n - i - j$. The *$j^{\underline{th}}$ subresultant* of A and B is the polynomial $\mathcal{S}_j(A,B) = \Sigma_{i=0}^{j}\det(M_{j,i})\cdot x^i$, a polynomial of degree j or less. We define also the *$j^{\underline{th}}$ principal subresultant coefficient* of A and B by *$\mathrm{psc}_j(A,B) = \det(M_{j,j})$*. Thus $\mathrm{psc}_j(A,B)$ is the coefficient of x^j in $\mathcal{S}_j(A,B)$. We note, for subsequent application, that if $\deg(A) = m > 0$ then $\mathrm{psc}_{m-1}(A,A') = m \cdot \mathrm{ldcf}(A)$.

Theorem 2. Let A and B be non-zero polynomials over a unique factorization domain, and let $k = \deg(\gcd(A,B))$. Then $\mathrm{psc}_j(A,B) = 0$ for $0 \le j < k$ and $\mathrm{psc}_k(A,B) \ne 0$.

Proof. By the fundamental theorem of polynomial remainder sequences, [2], for $0 \le j < k$, $\mathcal{S}_j(A,B) = 0$ so that $\mathrm{psc}_j(A,B) = 0$, and $\mathcal{S}_k(A,B)$ is similar to $\gcd(A,B)$ so that $\deg(\mathcal{S}_k(A,B)) = k$ and $\mathrm{psc}_k(A,B) \ne 0$. □

Using reducta and principal subresultant coefficients, we now obtain a more useful sufficient condition for the delineability of the real roots of a polynomial.

Theorem 3. Let $A(x_1,\ldots,x_r)$ be a real polynomial, $r \ge 2$, S a connected subset of R^{r-1} . Let $\mathcal{B}$ be the set of all reducta of A and $\mathcal{P} = \{\mathrm{psc}_k(B,B'):B \in \mathcal{B}\ \&\ 0 \le k \le \deg(B')\}$. Assume that every element of $\mathcal{P}$ is invariant on S . Then the real roots of A are delineable on S .

Let $\mathcal{A}$ be a set of real polynomials in r variables, $r \ge 2$. Let $\mathcal{B} = \mathcal{A} \cup \{A_1A_2:$ $A_1,A_2 \in \mathcal{A}\ \&\ A_1 \ne A_2\}$. Let $\mathcal{C} = \{\mathrm{red}^k(B):$ $B \in \mathcal{B}\ \&\ k \ge 0\}$. Let $\mathcal{P} = \{\mathrm{psc}_k(C,C'):$ $C \in \mathcal{C}\ \&\ 0 \le k \le \deg(C')\}$. Then $\mathcal{P}$ will be called the projection of $\mathcal{A}$. The following theorem uses projection to obtain a sufficient condition for the delineability of the real roots of a product of polynomials.

Theorem 4. Let $\mathcal{A} = \{A_1,\ldots,A_n\}$ be a set of n real polynomials in r variables, $r \ge 2$, and $A = \Pi_{i=1}^n A_i$. Let S be a connected subset of R^{r-1} . Let $\mathcal{P}$ be the projection of $\mathcal{A}$. Assume that every element of $\mathcal{P}$ is invariant on S . Then the real roots of A are delineable on S .

Let us write $\mathrm{der}(A)$ for A' , the derivative of A . We define $\mathrm{der}^0(A) = A$ and, inductively, $\mathrm{der}^{k+1}(A) = \mathrm{der}(\mathrm{der}^k(A))$ for $k \ge 0$.

Let $\mathcal{A}$ be a set of real polynomials in r variables, $r \ge 2$. Let $\mathcal{P}$ be the projection of $\mathcal{A}$, $\mathcal{D} = \{\mathrm{der}^k(A): A \in \mathcal{A}\ \&\ k \ge 0\}$, $\mathcal{R} = \{\mathrm{red}^k(D): D \in \mathcal{D}\ \&\ k \ge 0\}$, $\mathcal{P}' = \{\mathrm{psc}_k(R,R'):$ $R \in \mathcal{R}\ \&\ 0 \le k \le \deg(R')\}$. Then $\mathcal{P} \cup \mathcal{P}'$ will be called the augmented projection of $\mathcal{A}$. It is immediate from Theorem 3 that if the augmented projection of $\mathcal{A}$ is invariant on a connected subset S of R^{r-1} , then the real roots of each derivative of each element of $\mathcal{A}$ are delineable on S .

We now complete this section with discussion and specification of the more important subalgorithms which will be needed for the quantifier elimination algorithm.

The quantifier elimination algorithm of the next section will require computation of the projection or augmented projection of $\mathcal{A}$ just in case $\mathcal{A}$ is finite and $\mathcal{R} = I[x_1,\ldots,x_{r-1}]$, $r \ge 2$. Thus we assume the availability of an algorithm with the following specifications.

B = PROJ(A)

Input: $A = (A_1,\ldots,A_m)$ is a list of distinct integral polynomials in r variables, $r \ge 2$.

Output: $B = (B_1,\ldots,B_n)$ is a list of distinct integral polynomials in $r - 1$ variables, such that $\{B_1,\ldots,B_n\}$ is the projection of $\{A_1,\ldots,A_m\}$.

Another like algorithm, APROJ, is assumed for computing the augmented projection.

Now let $\mathcal{U}$ be a unique factorization domain, abbreviated u.f.d. If $a,b \in \mathcal{U}$ we say that a and b are associates, and write $a \sim b$, in case $a = ub$ for some unit u . An ample set for $\mathcal{U}$ (see [10]) is a set $\mathcal{A} \subseteq \mathcal{R}$ which contains exactly one element from each equivalence class of associates. Relative to $\mathcal{A}$ we can define a function gcd on $\mathcal{U} \times \mathcal{U}$ into $\mathcal{U}$ such that $\gcd(a,b) \in \mathcal{A}$ and $\gcd(a,b)$ is a greatest common divisor of a and b for all $a,b \in \mathcal{U}$. We will assume, moreover, that $\mathcal{A}$ is multiplicative, i.e. closed under multiplication, from which $1 \in \mathcal{A}$. Whenever $\mathcal{U}$ is a field we will have $\mathcal{A} = \{0,1\}$. For $\mathcal{U} = I$, we set $\mathcal{A} = \{0,1,2,\ldots\}$. $\mathcal{U}[x]$ is also a u.f.d. and if $\mathcal{A}$ is an ample set for $\mathcal{U}$ we take $\{A:\mathrm{ldcf}(A) \in \mathcal{A}\}$ as ample set for $\mathcal{U}[x]$ (see [14]) .

If $A(x) = \Sigma_{i=0}^n a_i x^i$ is a non-zero polynomial over $\mathcal{U}$, we set $\mathrm{cont}(A) = \gcd(a_n,a_{n-1},\ldots,a_0)$, the content of A , and we set $\mathrm{cont}(0) = 0$. If $A \ne 0$ we define $\mathrm{pp}(A)$, the primitive part of A , to be the ample associate of $A/\mathrm{cont}(A)$, and we set $\mathrm{pp}(0) = 0$. The polynomial A is primitive in case $\mathrm{cont}(A) = 1$. Clearly $\mathrm{pp}(A)$ is primitive and $A \sim \mathrm{cont}(A)\cdot\mathrm{pp}(A)$ for all $A \ne 0$.

Let $\mathcal{A}$ be a set of primitive polynomials of positive degree over $\mathcal{U}$. A basis for $\mathcal{A}$ is a set $\mathcal{B}$ of ample primitive squarefree polynomials of positive degree over $\mathcal{R}$ satisfying the following three conditions:

(a) If $B_1,B_2 \in \mathcal{B}$ and $B_1 \ne B_2$, then $\gcd(B_1,B_2) = 1$.

(b) If $B \in \mathcal{B}$, then $B|A$ for some $A \in \mathcal{A}$.

(c) If $A \in \mathcal{A}$, there exist $B_1,\ldots B_n \in \mathcal{B}$ and positive integers $e_1,\ldots,e_n$ such that $A \sim \Pi_{i=1}^n B_i^{e_i}$ (with $n = 0$ if $A \sim 1$) .

If $\mathcal{A}$ is an arbitrary set of polynomials over $\mathcal{U}$, then a basis for $\mathcal{A}$ is a set $\mathcal{B} = \mathcal{B}_1 \cup \mathcal{B}_2$ where $\mathcal{B}_1 = \{\mathrm{cont}(A):A \in \mathcal{A}\ \&\ A \ne 0\}$ and $\mathcal{B}_2$ is a basis for $\{\mathrm{pp}(A): A \in \mathcal{A}\ \&\ \deg(A) > 0\}$.

If $\mathcal{A}$ is a set of primitive polynomials of positive degree then the set $\mathcal{P}$ of ample irreducible divisors of elements of $\mathcal{A}$ is clearly a basis for A. If $\mathcal{B}_1$ and $\mathcal{B}_2$ are bases for A, we say that $\mathcal{B}_1$ is a refinement of $\mathcal{B}_2$ in case every element of $\mathcal{B}_1$ is a divisor of some element of $\mathcal{B}_2$. $\mathcal{P}$ is the finest basis of A in the sense that it is a refinement of every other basis.

Every set A also has a coarsest basis, $\mathcal{C}$, in the sense that every basis for A is a refinement of $\mathcal{C}$, as we will now see. Let $\mathcal{P}$ be the set of all ample irreducible divisors of positive degree of elements of A . For $P \varepsilon \mathcal{P}$, let $\sigma(P)$ be the set of all positive integers i such that, for some $A \varepsilon A$, i is the order of P in A . Let $e(P)$ be the greatest common divisor of the elements of $\sigma(P)$. For P,Q in $\mathcal{P}$, define $P \equiv Q$ in case, for every $A \varepsilon A$, the orders of P and Q in A are identical. Let $\mathcal{C}$ be the set of all products $\{\Pi_{Q\equiv P}Q\}^{\sigma(P)}$ with $P \varepsilon \mathcal{P}$. Then it can be shown that $\mathcal{C}$ is a coarsest basis for A .

If A is finite, its coarsest basis can be computed by g.c.d. calculation. Set $\mathcal{C} = A$. If A and B are distinct elements of $\mathcal{C}$, set $C = \gcd(A,B)$, $\bar{A} = A/C$, $\bar{B} = B/C$. If $C \neq 1$, replace A and B in $\mathcal{C}$ by the non-units from among $C,\bar{A}$ and $\bar{B}$. Eventually the elements of A will be pairwise relatively prime and $\mathcal{C}$ will be a coarsest basis for A .

A squarefree basis for A is a basis each of whose elements is squarefree. If A is any primitive element of $\mathcal{U}[x]$ of positive degree, there exist ample, squarefree, relatively prime polynomials $A_1,\ldots,A_k$ and integers $e_1 < \cdots < e_k$ such that $A \sim \Pi_{i=1}^{k} A_i^{e_i}$. $(A_1,\ldots,A_k)$ and $(e_1,\ldots,e_k)$ constitute the squarefree factorization of A . Musser, [14] and [15], discusses algorithms for squarefree factorization, which require, if $\mathcal{U}$ has characteristic zero, only differentiation, division and greatest common divisor calculations. We assume the availability of an algorithm for squarefree factorization in $\mathcal{U}[x]$ for the cases $\mathcal{U} = I[x_1,\ldots,x_{r-1}]$, $r \geq 1$, and $\mathcal{U} = Q(\alpha)$, where $Q(\alpha)$ is the real algebraic number field resulting from adjoining the real algebraic number α to the field Q of the rational numbers. For the case $\mathcal{U} = Q(\alpha)$ we assume the following specifications.

SQFREE(α,A,A,e)

Inputs: α is a real algebraic number. A is a primitive element of $Q(\alpha)[x]$ of positive degree.

Outputs: $\mathcal{A} = (A_1,\ldots,A_k)$ and $e = (e_1,\ldots,e_k)$ constitute the squarefree factorization of A .

A similar algorithm for the case $\mathcal{U} = I[x_1,\ldots,x_{r-1}]$ is needed in order to compute a coarsest squarefree basis for integral polynomials, as follows.

If $A \sim \Pi_{i=1}^{k} A_i^{e_i}$ is the squarefree factorization of A , then $\{A_1,\ldots,A_k\}$ is clearly a coarsest squarefree basis for $\{A\}$. Let $\bar{\mathcal{A}} = \{A_1,\ldots,A_m\}$ be a squarefree basis for $\mathcal{A}$, $\bar{\mathcal{B}} = \{B_1,\ldots,B_n\}$ a squarefree basis for $\mathcal{B}$. Consider the following algorithm proposed by R. Loos:

```
(1) For $j = 1,\ldots,n$ set $\bar{B}_j \leftarrow B_j$ .
(2) For $i = 1,\ldots,m$ do [$\bar{A}_i \leftarrow A_i$ ; for
      $j = 1,\ldots,n$ do ($C_{i,j} \leftarrow \gcd(\bar{A}_i,\bar{B}_j)$;
      $\bar{A}_i \leftarrow \bar{A}_i/C_{i,j}$;$\bar{B}_j \leftarrow \bar{B}_j/C_{i,j}$)].
(3) Exit.
```

Upon termination, the distinct nonunits among the $\bar{A}_i$, the $\bar{B}_j$ and the $C_{i,j}$ constitute a squarefree basis $\bar{\mathcal{C}}$ for $\mathcal{C} = \mathcal{A} \cup \mathcal{B}$. Moreover, if $\bar{\mathcal{A}}$ and $\bar{\mathcal{B}}$ are coarsest squarefree bases, then so is $\bar{\mathcal{C}}$. Thus by squarefree factorization and application of Loos' algorithm we can successively obtain coarsest squarefree bases for $\{A_1\}$, $\{A_1,A_2\},\ldots\{A_1,A_2,\ldots,A_m\}$. Thus we assume the availability of the following basis algorithm.

$\mathcal{B}$ = BASIS($\mathcal{A}$)

Input: $\mathcal{A} = (A_1,\ldots,A_m)$ is a list of distinct integral polynomials in r variables, $r \geq 2$.

Output: $\mathcal{B} = (B_1,\ldots,B_n)$ is a list of distinct integral polynomials in $r - 1$ variables such that $\{B_1,\ldots,B_n\}$ is a coarsest squarefree basis for $\{A_1,\ldots,A_m\}$.

A similar algorithm, ABASIS, with an additional input α , a real algebraic number, will be assumed for computing the coarsest squarefree basis when A is a finite list of univariate polynomials over $Q(\alpha)$.

A recent Ph.D. thesis by Rubald, [16] provides algorithms for the arithmetic operations in the field $Q(\alpha)$ and in the polynomial domain $Q(\alpha)[x]$. Rubald also provides an efficient modular homomorphisms algorithm for g.c.d. calculation in $Q(\alpha)[x]$. An important feature of Rubald's work is that the minimal polynomial of α is not required. Instead, α is represented by any pair (A,I) such that A is a primitive squarefree integral polynomial of positive degree with $A(\alpha) = 0$, and $I = (r,s)$ is an open interval with rational number endpoints such that α is the unique zero of A in I . This feature is important because as yet (see [6]) no algorithm with polynomial-dominated maximum computing time is known for factoring a primitive univariate integral polynomial into its irreducible factors. A non-zero element β of $Q(\alpha)$ is then represented by any polynomial $B(x) \varepsilon Q[x]$ such that $\deg(B) < \deg(A)$ and $B(\alpha) = \beta$ Although this representation fails to be unique whenever A is reducible, no difficulties arise.

The following algorithm will be needed.

n = NROOTS(α,B,I)

Inputs: α is a real algebraic number. B is a polynomial over $Q(\alpha)$. $I = (r,s)$ is an open interval, where r and s are rational numbers or $r = -\infty$ or $s = +\infty$.

Output: n is the number of zeros of B in I (multiplicities not counted).

This algorithm is easily obtained using Sturm's theorem, since Rubald's work provides an efficient algorithm for determining the sign of any element of $Q(\alpha)$, and because his algorithm for g.c.d. calculation in $Q(\alpha)[x]$ can be extended to the computation of Sturm sequences.

The following algorithm for isolating the real zeros of a finite set of polynomials over $Q(\alpha)$ will also be used.

ISOL(α,A,I,ν)

Inputs: α is a real algebraic number. $A = (A_1,\ldots,A_m)$ is a list of non-zero squarefree and pairwise relatively prime polynomials over $Q(\alpha)$.

Outputs: $I = (I_1,\ldots,I_n)$ is a list of open intervals with rational endpoints with $I_1 < I_2 < \cdots < I_n$ such that each I_j contains exactly one real zero of $A = \Pi_{i=1}^{m} A_i$, and every real zero of A belongs to some I_j. $\nu = (\nu_1,\ldots,\nu_n)$ is such that the zero of A in I_j is a zero of A_{ν_j}.

The algorithm ISOL can be easily obtained by application of Sturm's theorem and repeated interval bisection. Heindel, [11], presents an algorithm of this type for the case of a single univariate integral polynomial. If the real zeros of each A_i are separately isolated, then the resulting intervals can be refined until they no longer overlap, while retaining the identity of the polynomials from which they came.

In the quantifier elimination algorithm, occasion will arise to reduce a multiple real algebraic extension of the rationals, $Q(\alpha_1,\ldots,\alpha_m)$, to a simple extension $Q(\alpha)$. This can be accomplished by iterating an algorithm of Loos and Collins, [12], based on resultant theory, with the following specifications.

SIMPLE$(\alpha,\beta,\gamma,A,B)$

Inputs: α and β are real algebraic numbers.

Outputs: γ is a real algebraic number. A and B are polynomials which represent α and β respectively as elements of $Q(\gamma)$.

Finally, one additional subalgorithm, also provided in [12], is the following.

NORMAL$(\alpha,A,I,\bar{A},\bar{I})$

Inputs: α is a real algebraic number. A is a non-zero polynomial over $Q(\alpha)$. $I = (I_1,\ldots,I_m)$ is a list of rational isolating intervals, $I_1 < I_2 < \cdots < I_m$, for the real zeros of A.

Outputs: $\bar{A}$ is a non-zero squarefree primitive integral polynomial such that every real zero of A is a real zero of $\bar{A}$. $\bar{I} = (\bar{I}_1,\ldots,\bar{I}_m)$ is a list of rational intervals with $\bar{I}_j \subseteq I_j$ such that if α_j is the zero of A in I_j then α_j is the unique zero of $\bar{A}$ in $\bar{I}_j$, $1 \le j \le r$.

3. The Main Algorithms. We define, by induction on r, a *cylindrical algebraic decomposition* of R^r, abbreviated c.a.d. For $r = 1$, a c.a.d. of R is a sequence $(S_1,S_2,\ldots,S_{2\nu+1})$, where either $\nu = 0$ and $S_1 = R$, or $\nu > 0$ and there exist ν real algebraic numbers $\alpha_1 < \alpha_2 < \cdots < \alpha_\nu$ such that $S_{2i} = \{\alpha_i\}$ for $1 \le i \le \nu$, S_{2i+1} is the open interval (α_i,α_{i+1}) for $1 \le i < \nu$, $S_1 = (-\infty,\alpha_1)$ and $S_{2\nu+1} = (\alpha_\nu,\infty)$ Now let $r > 1$, and let $(S_1,\ldots,S_\mu)$ be any c.a.d. of R^{r-1}. For $1 \le i \le \mu$, let $f_{i,1} < f_{i,2} < \cdots < f_{i,\nu_i}$ be continuous real-valued algebraic functions on S_i. If $\nu_i = 0$, set $S_{i,1} = S_i \times R$. If $\nu_i > 0$ set $S_{i,2j} = f_{i,j}$, that is, $S_{i,2j} = \{(a,b):a \in S_i \;\&\; b = f_{i,j}(a)\}$ for $1 \le j \le \nu_i$, set $S_{i,2j+1} = \{(a,b): a \in S_i \;\&\; f_{i,j}(a) < b < f_{i,j+1}(a)\}$ for $1 \le j < \nu_i$, set $S_{i,1} = \{(a,b): a \in S_i \;\&\; b < f_{i,1}(a)\}$, and set $S_{i,2\nu_i+1} = \{(a,b): a \in S_i \;\&\; f_{i,\nu_i}(a) < b\}$. A c.a.d of R^r is any sequence $(S_{1,1},\ldots,S_{1,2\nu_1+1},\ldots,S_{\mu,1},\ldots,S_{\mu,2\nu_\mu+1})$ which can be obtained by this construction from a c.a.d. of R^{r-1} and functions $f_{i,j}$ as just described.

It is important to observe that the cylinder $S_i \times R$ is the disjoint union $\bigcup_{j=1}^{2\nu_i+1} S_{i,j}$ for $1 \le i \le \mu$. If $S = (S_1,\ldots,S_\mu)$ is any c.a.d. of R^r, the S_i will be called the *cells* of S. Clearly every cell of a c.a.d. is a connected set. If $\mathcal{A}$ is a set of real polynomials in r variables, the c.a.d. S of R^r is *$\mathcal{A}$-invariant* in case each A in $\mathcal{A}$ is invariant on each cell of S.

A *sample* of the c.a.d. $S = (S_1,\ldots,S_\mu)$ is a tuple $\beta = (\beta_1,\ldots,\beta_\mu)$ such that $\beta_i \in S_i$ for $1 \le i \le \mu$. The sample β is *algebraic* in case each β_i is an algebraic point, i.e. each coordinate of β_i is an algebraic number. A *cylindrical sample* is defined by induction on r. For $r = 1$, any sample is cylindrical. For $r > 1$, let $S = (S_{1,1},\ldots,S_{1,2\nu_1+1},\ldots,S_{\mu,1},\ldots,S_{\mu,2\nu_\mu+1})$ be a c.a.d. of R^r constructed from a c.a.d. $S^* = (S_1,\ldots,S_\mu)$ of R^{r-1}, and let $\beta^* = (\beta_1,\ldots,\beta_\mu)$ be a sample of S^*. The sample $\beta = (\beta_{1,1},\ldots,\beta_{1,2\nu_1+1},\ldots,\beta_{\mu,1},\ldots,\beta_{\mu,2\nu_\mu+1})$ of S is cylindrical if the first $r - 1$ coordinates of $\beta_{i,j}$ are, respectively, the coordinates of β_i, for all i and j, and β^* is cylindrical. Cylindrical algebraic sample will be abbreviated c.a.s.

Since a c.a.d. of R^r can be constructed from a unique c.a.d. of R^{r-1}, any c.a.d. S of R^r determines, for $1 \leq k < r$, a c.a.d. S^* of R^k, which will be called the c.a.d. of R^k induced by S. Similarly any c.a.s. β of S induces a unique c.a.s. β^* of S^*.

If S is an arbitrary subset of R^r, the standard formula $\phi(x_1,\ldots,x_r)$ containing just $x_1,\ldots,x_r$ as free variables, defines S in case $S = \{(a_1,\ldots,a_r)\colon a_1,\ldots,a_r \in R \;\&\; \phi(a_1,\ldots,a_r)\}$. A standard definition of the c.a.d. $S = (S_1,\ldots,S_\mu)$ is a sequence $(\phi_1,\ldots,\phi_\mu)$ such that, for $1 \leq i \leq \mu$, ϕ_i is a standard quantifier-free formula which defines S_i.

We are now prepared to describe a decomposition algorithm, DECOMP. The inputs to DECOMP are a finite set A of integral polynomials in r variables, $r \geq 1$, and an integer k, $0 \leq k \leq r$. The outputs of DECOMP are a c.a.s. β of some A-invariant c.a.d. S of R^r and, if $k \geq 1$, a standard definition ψ of the c.a.d. S^* of R^k induced by S.

Before proceeding to describe DECOMP we first explain its intended use in the quantifier elimination algorithm, ELIM, which will be described subsequently. ELIM has two distinct stages. Given as input a standard prenex formula ϕ, namely $(Q_{k+1}x_{k+1})\cdots(Q_r x_r)\hat{\phi}(x_1,\ldots,x_r)$, ELIM applies DECOMP to the set $\hat{A}$ of all non-zero polynomials occurring in $\hat{\phi}$, and the integer k. The outputs β and ψ of DECOMP, together with the formula ϕ, are then input to an "evaluation" algorithm, EVAL, which produces a standard quantifier-free formula $\phi^*(x_1,\ldots,x_k)$ which is equivalent to ϕ. Thus, ELIM does little more than to successively invoke DECOMP and EVAL.

DECOMP uses a subalgorithm, DEFINE, for construction of the standard definition. The inputs to DEFINE are an integral polynomial $A(x_1,\ldots,x_r)$, $r \geq 2$, such that for some connected set $S \subseteq R^{r-1}$ the real roots of A and of each derivative of A are delineable on S, and an algebraic point $\beta \in S$. The output of DEFINE is a sequence $(\phi_1,\ldots,\phi_{2m+1})$ of standard quantifier-free formulas ϕ_i such that if ϕ is any formula which defines S, then the conjunction $\phi \wedge \phi_i$ defines the $i^{\underline{th}}$ cell of the cylinder $S \times R$ determined by the m real roots of A on S, as in the definition of a c.a.d. The description of DEFINE will be given following that of DECOMP.

DECOMP(A,k,β,ψ)

Inputs: $A = (A_1,\ldots,A_m)$ is a list of distinct integral polynomials in r variables, $r \geq 1$. k is an integer such that $0 \leq k \leq r$.

Outputs: β is a c.a.s. for some $\mathcal{A}$-invariant c.a.d. S of R^r. ψ is a standard definition of the c.a.d. S^* of R^k induced by S if $k > 0$, and ψ is the null list if $k = 0$.

Algorithm Description

(1) If $r > 1$, go to (4). Apply BASIS to A, obtaining a coarsest squarefree basis $\mathcal{B} = (B_1,\ldots,B_h)$ for A. Apply ISOL to $\mathcal{B}$, obtaining outputs $I = (I_1,\ldots,I_n)$ and $\nu = (\nu_1,\ldots,\nu_n)$. (Each I_j contains a unique zero, say α_j, of B_{ν_j}, and $\alpha_1 < \alpha_2 < \cdots < \alpha_n$ are all the real zeros of elements of A. Thus the α_j determine an A-invariant c.a.d. S of R, and (B_{ν_j},I_j) represents α_j.)

(2) For $j = 1,\ldots,n$, where $I_j = (r_j,s_j)$, set $\beta_{2j-1} \leftarrow r_j$ and $\beta_{2j} \leftarrow \alpha_j$. If $n = 0$, set $\beta_{2n+1} \leftarrow 0$ and if $n > 0$, set $\beta_{2n+1} \leftarrow s_n$. Set $\beta \leftarrow (\beta_1,\ldots,\beta_{2n+1})$. ($\beta$ is now a c.a.s. of S.)

(3) If $k = 0$, set $\psi \leftarrow (\;)$ and exit. If $n = 0$, set $\psi_1 \leftarrow$ "$0=0$", $\psi \leftarrow (\psi_1)$ and exit. For $i = 1,\ldots,h$ do [$\sigma_{i,n} \leftarrow \mathrm{sign}(\mathrm{ldcf}(B_i))$; for $j = n-1,\ldots,0$ set $\sigma_{i,j} \leftarrow (-\sigma_{i,j+1}$ if $i = \nu_{j+1}$, $\sigma_{i,j+1}$ otherwise)]. (Now $\sigma_{i,j}$ is the sign of B_i in S_{2j+1}, where $S = (S_1,\ldots,S_{2n+1})$.) For $j = 1,\ldots,n$, where $r_j = a_j/b_j$ and $s_j = c_j/d_j$ with $b_j > 0$ and $d_j > 0$, set $\psi_{2j} \leftarrow$ "$B_{\nu_j} = 0 \;\&\; b_j x_1 - a_j > 0 \;\&\; d_j x_1 - c_j < 0$". (Now ψ_{2j} defines $S_j = \{\alpha_j\}$.) For $j = 1,\ldots,n$, set $\psi_{2j+1} \leftarrow$ "$\sigma_{\nu_j,2j+1}B_{\nu_j} > 0 \;\&\; \sigma_{\nu_{j+1},2j+1}B_{\nu_{j+1}} > 0 \;\&\; b_j x_1 - a_j > 0 \;\&\; d_{j+1}x_1 - c_{j+1} < 0$". (If $\nu_j = \nu_{j+1}$ then the first two conjuncts are identical, so one can be omitted.) Set $\psi_1 \leftarrow$ "$\sigma_{\nu_1,1}B_{\nu_1} > 0 \;\&\; d_1 x_1 - c_1 < 0$" and $\psi_{2n+1} \leftarrow$ "$\sigma_{\nu_n,2n+1}B_{\nu_n} > 0 \;\&\; b_n x_1 - a_n > 0$". Set $\psi \leftarrow (\psi_1,\ldots,\psi_{2n+1})$. ($\psi$ is now a standard definition of S.) Exit.

(4) Apply BASIS to A, obtaining $\mathcal{B}$, a coarsest squarefree basis for A. (This action is inessential; we could set $\mathcal{B} \leftarrow A$. But the algorithm is likely more efficient if the coarsest squarefree basis is used, and it may be still more efficient, on the average, if the finest basis is computed here.) If $k < r$, apply PROJ to $\mathcal{B}$, obtaining the projection, $\mathcal{P}$, of $\mathcal{B}$. If $k = r$, apply APROJ to $\mathcal{B}$, obtaining the augmented projection, $\mathcal{P}$, of $\mathcal{B}$.

(5) If $k = r$, set $k' \leftarrow k - 1$; otherwise, set $k' \leftarrow k$. Apply DECOMP (recursively) to $\mathcal{P}$ and k', obtaining as outputs β' and ψ'. (For some $\mathcal{P}$-invariant c.a.d. S' of R^{r-1}, β' is a c.a.s. of S' and ψ' is a standard definition of the

c.a.d. S^* of $R^{k'}$ induced by S' , except that $\psi' = (\ )$ if $k' = 0$. Since $\mathcal{P}$ contains the projection of $\mathcal{B}$ and S' is $\mathcal{P}$-invariant the real zeros of the elements of $\mathcal{B}$ are delineable on each cell of S' by Theorem 4 . Hence S' , together with the real algebraic functions defined by elements of $\mathcal{B}$ on the cells of S' , determines a c.a.d. S of R^r . S is $\mathcal{B}$-invariant and therefore also $\mathcal{A}$-invariant since $\mathcal{B}$ is a basis for $\mathcal{A}$. Also, S^* is induced by S .)

(6) (This step extends the c.a.s. β' of S' to a c.a.s. β of S . Let $\beta' = (\beta'_1,\ldots,\beta'_\ell)$ and $\beta'_j = (\beta'_{j,1},\ldots,\beta'_{j,r-1})$. We assume, inductively, that there is associated with each algebraic point β'_j an algebraic number α'_j such that $Q(\beta'_{j,1},\ldots,\beta'_{j,r-1}) = Q(\alpha'_j)$ and polynomials $B'_{j,k}$ which represent the $\beta'_{j,k}$. The basis for this induction is trivial since the polynomial x represents $\beta'_{j,1} = \alpha'_j$ as an element of $Q(\alpha'_j)$ if α'_j is irrational, and if α'_j is rational it represents itself as an element of $Q = Q(\alpha'_j)$.) Let $\mathcal{B} = (B_1,\ldots,B_n)$. For $j = 1,\ldots,\ell$ do [For $i = 1,\ldots,h$ set $B^*_{j,i}(x) \leftarrow B_i(\beta'_{j,1},\ldots,\beta'_{j,r-1},x)$. ($B^*_{j,i}$ is a polynomial over $Q(\alpha'_j)$.) Apply ABASIS to α'_j and $(B^*_{j,1},\ldots,B^*_{j,h})$, obtaining $\hat{\mathcal{B}}_j = (\hat{B}_{j,1},\ldots,\hat{B}_{j,m_j})$, a coarsest squarefree basis. Apply ISOL to α'_j and $\hat{\mathcal{B}}_j$, obtaining outputs $I_j = (I_{j,1},\ldots,I_{j,n_j})$ and $\nu_j = (\nu_{j,1},\ldots,\nu_{j,n_j})$. ($\hat{B}_{j,\nu_{j,k}}$ has a unique real zero $\gamma_{j,k}$ in $I_{j,k}$, and $\gamma_{j,1} < \cdots < \gamma_{j,n_j}$ are all the real zeros of elements of $\hat{\beta}_i$.) Apply NORMAL to α'_j , $\hat{\mathcal{B}}_j$ and I_j , obtaining outputs $\bar{\mathcal{B}}_j = (\bar{B}_{j,1},\ldots,\bar{B}_{j,n_j})$ and $\bar{I}_j = (I_{j,1},\ldots,I_{j,n_j})$. (Now $\gamma_{j,k}$ is represented by $(\bar{B}_{j,k},I_{j,k})$.) If $n_j = 0$, set $\delta_{j,1} \leftarrow 0$. If $n_j > 0$, for $k = 1,\ldots,n_j$, where $I_{j,k} = (r_{j,k},s_{j,k})$, set $\delta_{j,2k-1} \leftarrow r_{j,k}$ and $\delta_{j,2k} \leftarrow \gamma_{j,k}$; also set $\delta_{j,2n_j+1} \leftarrow s_{j,n_j}$. For $k = 1,\ldots,2n_j + 1$, set $\beta_{j,k} \leftarrow (\beta'_{j,1},\ldots,\beta'_{j,r-1},\delta_{j,k})$. For $k = 1,\ldots,2n_j + 1$ apply SIMPLE to α'_j and $\delta_{j,k}$, obtaining outputs $\alpha_{j,k}, A_{j,k}$ and $B_{j,k}$. (Now $Q(\beta'_{j,1},\ldots,\beta'_{j,r-1},\delta_{j,k}) = Q(\alpha'_j,\delta_{j,k}) = Q(\alpha_{j,k})$, $A_{j,k}$ represents α'_j in $Q(\alpha_{j,k})$, and $B_{j,k}$ represents $\delta_{j,k}$ in $Q(\alpha_{j,k})$.) For $h = 1,\ldots, r-1$ and $k = 1,\ldots,2n_j+1$, where $\alpha_{j,k}$ is represented by $(C_{j,k},I'_{j,k})$, set $D_{j,h,k}(x) \leftarrow B'_{j,h}(A_{j,k}(x))$ modulo $C_{j,k}(x)$. ($\alpha'_j = A_{j,k}(\alpha_{j,k})$, $\beta'_{j,h} = B'_{j,h}(\alpha'_j)$ and $C_{j,k}(\alpha_{j,k}) = 0$, so $D_{j,h,k}$ represents $\beta'_{j,h}$ in $Q(\alpha_{j,k})$.)] Set $\beta \leftarrow (\beta_{1,1},\ldots,\beta_{1,2n_1+1},\ldots,\beta_{\ell,1},\ldots,\beta_{\ell,2n_\ell+1})$. (Now β is a c.a.s. of S .)

(7) If $k < r$, set $\psi \leftarrow \psi'$ and exit. (If $k < r$, then $k' = k$ so ψ' is a standard definition of the c.a.d. S^* of R^k induced by S' , and hence induced also by S . Otherwise, $k = r$, $k' = r - 1$ and we next proceed to extend the standard definition ψ' of S' to a standard definition ψ of S . Since $k = r$, $\mathcal{P}$ is the augmented projection of $\mathcal{B}$ and, by the remark following Theorem 4, the real roots of every derivative of every element of $\mathcal{B}$ are delineable on every cell of S' because S' is $\mathcal{P}$-invariant.) For $i = 1,\ldots,\ell$ do [For $i = 1,\ldots,h$ apply DEFINE to B_i and β'_j , obtaining as output a sequence $\chi_{i,j} = (\chi_{i,j,1},\ldots,\chi_{i,j,2n_{i,j}+1})$. ($\chi_{i,j,k}$ is a standard quantifier-free formula such that ψ'_j & $\chi_{i,j,k}$ defines the kth cell of the cylinder $S'_j \times R$ as determined by the real zeros of B_i on S'_j . We next proceed to use the $\chi_{i,j,k}$ to define the cells of the cylinder $S'_j \times R$ as determined by the real zeros of $B = \Pi_{i=1}^h B_i$, that is, the cells of the jth cylinder of S , using the results of step (6) . Observe that B has n_j real zeros on S_j and that the kth real zero is a zero of $\hat{B}_{j,\nu_{j,k}}$.) For $k = 1,\ldots,n_j$ set $\lambda_{j,k} \leftarrow$ least t such that $\hat{B}_{j,\nu_{j,k}} | B^*_{j,t}$ and apply NROOTS to $\alpha'_j, B^*_{j,\lambda_{j,k}}$ and $(-\infty,s_{j,k})$ obtaining $\mu_{j,k}$ as output. (Now the kth zero of B on S'_j is the $\mu_{j,k}$th zero of $B_{\nu_{j,k}}$.) For $k = 1,\ldots,n_j$ set $\psi_{j,2k} \leftarrow \psi'_j$ & $\chi_{\nu_{j,k},j,2\mu_{j,k}}$. For $k = 1,\ldots,n_j-1$ set $\psi_{j,2k+1} \leftarrow \chi_{\nu_{j,k},j,2\mu_{j,k}+1}$ & $\chi_{\nu_{j,k+1},j,2\mu_{j,k+1}-1}$. (If $\nu_{j,k} = \nu_{j,k+1}$ then $\mu_{j,k+1} = \mu_{j,k} + 1$ so the last two conjuncts coincide and one may be omitted.) If $n_j > 0$ set $\psi_{j,1} \leftarrow \psi'_j$ & $\chi_{\nu_{j,1},j,1}$ and $\psi_{j,2n_j+1} \leftarrow \psi'_j$ & $\chi_{\nu_{j,n_j},j,2\mu_{j,n_j}+1}$. If $n_j = 0$, set $\psi_{j,1} \leftarrow \psi'_j$.] Set $\psi \leftarrow (\psi_{1,1},\ldots,\psi_{1,2n_1+1},\ldots,\psi_{\ell,1},\ldots,\psi_{\ell,2n_\ell+1})$. (Now ψ is a standard definition of S .) Exit.

Next we describe the algorithm DEFINE.

$$\phi = \text{DEFINE}(B,\beta)$$

<u>Inputs</u>: B is an integral polynomial in r variables, $r \geq 2$, such that for some connected set $S \subseteq R^{r-1}$ the real roots of B and of each derivative of B are delineable on S . β is an algebraic point of S .

<u>Output</u>: $\phi = (\phi_1,\ldots,\phi_{2m+1})$ is a sequence of standard quantifier-free formulas ϕ_i such that if

ψ defines S then ψ & ϕ_i defines the ith cell of the cylinder $S \times R$ as determined by the m real roots of B on S.

Algorithm Description

(1) (We let $\beta = (\beta_1,\ldots,\beta_{r-1})$. As in DECOMP, we may assume that we are given an algebraic number α such that $Q(\beta_1,\ldots,\beta_{r-1}) = Q(\alpha)$, and polynomials B_i which represent β_i as elements of $Q(\alpha)$. Set $B^*(x) = B(\beta_1,\ldots,\beta_{r-1},x)$. Apply SQFREE to α and B^*, obtaining the list $\mathcal{B}^* = (B_1^*,\ldots,B_h^*)$ of squarefree factors of B^* and the list $(e_1,\ldots,e_h)$ of corresponding exponents. Apply ISOL to α and $\mathcal{B}^*$, obtaining as outputs the lists $(I_1,\ldots,I_m)$ and $(\nu_1,\ldots,\nu_m)$. (I_j isolates the jth real zero, γ_j, of the elements of $\mathcal{B}^*$, and γ_j is a zero of $B_{\nu_j}^*$.) If $m = 0$, set $\phi_1 \leftarrow$ "$0 = 0$", $\phi \leftarrow (\phi_1)$, and exit. For $i = 1,\ldots,m$ set $\mu_i \leftarrow e_{\nu_i}$. (Now γ_i is a zero of $B_{\nu_i}^*$ of multiplicity μ_i.) Set $\sigma_m \leftarrow \text{sign}(\text{ldcf}(B^*))$. For $j = m-1,\ldots,0$ set $\sigma_j \leftarrow (-1)^{\mu_{j+1}}\sigma_{j+1}$. (Now σ_j is the sign of B in the $(2j+1)$th cell of the B-invariant decomposition of the cylinder $S \times R$.) If $m = 1$ and μ_1 is odd, set $\phi_1 \leftarrow$ "$\sigma_0 B > 0$", $\phi_2 \leftarrow$ "$B = 0$", $\phi_3 \leftarrow$ "$\sigma_1 B > 0$", $\phi \leftarrow (\phi_1,\phi_2,\phi_3)$, and exit.

(2) Set $B^{*\prime} \leftarrow \text{der}(B^*)$, $G \leftarrow \gcd(B^*,B^{*\prime})$ and $H \leftarrow B^{*\prime}/G$. (Now $H(\delta) = 0$ if and only if $B^{*\prime}(\delta) = 0$ and $B^*(\delta) \neq 0$.) For $j = 1,\ldots,m$, applying NROOTS, refine I_j so that $\gamma_j \varepsilon I_j$ but I_j contains no zero of H. (Now I_j contains no zeros of $B^{*\prime}$ other than γ_j. Let $I_j = (r_j,s_j)$.) For $j = 1,\ldots,m-1$, set $n_j \leftarrow \text{NROOTS}(\alpha,H,(s_j,r_{j+1}))$. Set $n_0 \leftarrow \text{NROOTS}(\alpha,H,(-\infty,r_1))$ and $n_m \leftarrow \text{NROOTS}(\alpha,H,(s_m,\infty))$. Set $\lambda_1 \leftarrow n_0$. For $j = 1,\ldots,m$ set $\lambda_{2j} \leftarrow \{\lambda_{2j-1}$ if $\mu_j = 1$; $\lambda_{2j-1} + 1$ if $\mu_j > 1\}$, and $\lambda_{2j+1} \leftarrow \lambda_{2j} + n_j$. (Now λ_{2j-1} is the number of zeros of $B^{*\prime}$ less than γ_j, λ_{2j} is the number less than or equal to γ_j, and λ_{2m+1} is the number of all the zeros.)

(3) Set $B' \leftarrow \text{der}(B)$. Apply DEFINE to B' and β, obtaining $(\phi_1',\ldots,\phi_\ell')$ as output. (Thus DEFINE is a recursive algorithm; its termination is assured because $\deg(B') < \deg(B)$.)

(4) (This step computes ϕ_{2i} for $1 \leq i \leq m$.) For $i = 1,\ldots,m$ if $\mu_i > 1$ set $\phi_{2i} \leftarrow \phi'_{2\lambda_{2i}}$. (If $\mu_i > 1$ then the ith real zero of B is the λ_{2i}-th real zero of B'.) For $i = 1,\ldots,m$ if $\mu_i = 1$ set $\phi_{2i} \leftarrow B = 0$ & $\phi'_{2\lambda_{2i-1}+1}$. (There are λ_{2i-1} zeros of B less than the ith zero of B, so the ith zero of B is in the $\lambda_{2\lambda_{i-1}+1}$-th cell of the B' decomposition. By Rolle's theorem, any two real zeros of B are separated by a zero of B' so there is only one zero of B in this cell.)

(5) (This step defines ϕ_{2i+1} for $1 \leq i < m$. There are four cases.) For $i = 1,\ldots,m-1$ if $\mu_i > 1$ and $\mu_{i+1} > 1$ set $\phi_{2i+1} \leftarrow \bigvee_{2\lambda_{2i}+1 \leq j \leq 2\lambda_{2i+2}-1}\phi_j'$. (In this case the ith zero of B is the λ_{2i}th zero of B' and the $(i+1)$th zero of B is the λ_{2i+2}th zero of B'.) For $i = 1,\ldots,m-1$ if $\mu_i = 1$ and $\mu_{i+1} > 1$ set $\phi_{2i+1} \leftarrow \{\sigma_i B > 0 \;\&\; \phi'_{2\lambda_{2i}+1}\} \vee \{\bigvee_{2\lambda_{2i}+2 \leq j \leq 2\lambda_{2i+2}-1}\phi_j'\}$. (There are λ_{2i} zeros of B' less than the ith zero of B. By Rolle's theorem the ith zero of B is the only zero of B in the $(2\lambda_{2i}+1)$th cell of the B' decomposition. Since $\mu_i = 1$, B changes sign from σ_{i-1} to σ_i at this zero.) For $i = 1,\ldots,m-1$ if $\mu_i > 1$ and $\mu_{i+1} = 1$ set $\phi_{2i+1} \leftarrow \{\sigma_i B > 0 \;\&\; \phi'_{2\lambda_{2i+2}+1}\} \vee \{\bigvee_{2\lambda_{2i}+1 \leq j \leq 2\lambda_{2i+2}}\phi_j'\}$. (This case is similar to the preceding case.) For $i = 1,\ldots,m-1$ if $\mu_i = 1$ and $\mu_{i+1} = 1$ set $\phi_{2i+1} \leftarrow \{\sigma_i B > 0 \;\&\; \phi'_{2\lambda_{2i}+1}\} \vee \{\sigma_i B > 0 \;\&\; \phi'_{2\lambda_{2i+2}+1}\} \vee \{\bigvee_{2\lambda_{2i}+2 \leq j \leq 2\lambda_{2i+2}}\phi_j'\}$. (The final disjunction may be empty.)

(6) (This step defines ϕ_1 and ϕ_{2m+1}.) If $\mu_1 > 1$ set $\phi_1 \leftarrow \bigvee_{1 \leq j \leq 2\lambda_2-1}\phi_j'$. If $\mu_1 = 1$ set $\phi_1 \leftarrow \{\sigma_0 B > 0 \;\&\; \phi'_{2\lambda_2+1}\} \vee \{\bigvee_{1 \leq j \leq 2\lambda_2}\phi_j'\}$. If $\mu_m > 1$ set $\phi_{2m+1} \leftarrow \bigvee_{2\lambda_{2m}+1 \leq j \leq 2\lambda_{2m+1}+1}\phi_j'$. If $\mu_m = 1$ set $\phi_{2m+1} \leftarrow \{\sigma_m B > 0 \;\&\; \phi'_{2\lambda_{2m}+1}\} \vee \{\bigvee_{2\lambda_{2m}+2 \leq j \leq 2\lambda_{2m+1}+1}\phi_j'\}$. Set $\phi \leftarrow (\phi_1,\ldots,\phi_{2m+1})$ and exit.

Let ϕ be any formula in r free variables and let $S \subseteq R^r$. ϕ is invariant on S in case either $\phi(a_1,\ldots,a_r)$ is true for all $(a_1,\ldots,a_r) \varepsilon S$ or $\phi(a_1,\ldots,a_r)$ is false for all $(a_1,\ldots,a_r) \varepsilon S$. If S is a c.a.d. of R^r, we say that S is ϕ-invariant in case ϕ is invariant on each cell of S. If ϕ is a standard quantifier-free formula in r variables, A is the set of all non-zero polynomials which occur in ϕ, and S is an A-invariant c.a.d. of R^r, then clearly S is also ϕ-invariant.

If ϕ is a sentence, we will denote by $v(\phi)$ the truth value of ϕ, with "true" represented by 1, "false" by 0. Accordingly, if $(v_1,\dots,v_n)$ is a vector of zeros and ones, then we define $\wedge_{i=1}^n v_i = 1$ if each $v_i = 1$ and $\wedge_{i=1}^n v_i = 0$ otherwise. Similarly, we define $\vee_{i=1}^n v_i = 0$ each $v_i = 0$ and $\vee_{i=1}^n v_i = 1$ otherwise. If ϕ is a formula in r free variables and $a = (a_1,\dots,a_r) \in R^r$, we set $v(\phi,a) = v(\phi(a_1,\dots,a_r))$. If ϕ is invariant on S, we set $v(\phi,S) = v(\phi,a)$ for any $a \in S$.

The following theorem is fundamental in the use of a c.a.d. for quantifier elimination. However, the proof, being straightforward, is omitted.

Theorem 5. Let $\phi(x_1,\dots,x_r)$ be a formula in r free variables and let ϕ^* be $(\forall x_r)\phi$ or $(\exists x_r)\phi$. If $r > 1$, let S be a ϕ-invariant c.a.d. of R^r, S^* the c.a.d. of R^{r-1} induced by S. Then S^* is ϕ^*-invariant. If $S^* = (S_1,\dots,S_m)$ and $S = (S_{1,1},\dots,S_{1,n},\dots,S_{m,1},\dots,S_{m,n_m})$ where $(S_{i,1},\dots,S_{i,n_i})$ is the $i^{\underline{th}}$ cylinder of S, then $v((\forall x_r)\phi,S_i) = \wedge_{j=1}^{n_i} v(\phi,S_{ij})$ and $v((\exists x_r)\phi,S_i) = \vee_{j=1}^{n_i} v(\phi,S_{i,j})$. If $r = 1$ and $S = (S_1,\dots,S_n)$ is a c.a.d. of R, then $v((\forall x_1)\phi) = \wedge_{i=1}^n v(\phi,S_i)$ and $v((\exists x_1)\phi) = \vee_{i=1}^n v(\phi,S_i)$.

Let $a,b, \in R^r$ with $a = (a_1,\dots,a_r)$ and $b = (b_1,\dots,b_r)$. We define $a \sim_k b$ in case $a_i = b_i$ for $1 \le i \le k$. Note that $a \sim_r b$ if and only if $a = b$, while $a \sim_0 b$ for all $a,b \in R_r$. We define $a < b$ in case $a \sim_k b$ and $a_{k+1} < b_{k+1}$ for some k, $0 \le k < r$. The relation $a < b$ is a linear order on R^r, which we recognize as the lexicographical order on R^r induced by the usual order on R. We note that if $(\beta_1,\dots,\beta_m)$ is a cylindrical sample of a c.a.d. S, then $\beta_1 < \beta_2 < \cdots < \beta_m$.

The cylindrical structure of a c.a.d. S is obtainable from any c.a.s. β of S. We define a grouping function g. Let $\beta = (\beta_1,\dots,\beta_m)$ be any sequence of elements of R^r. Then for $0 \le k \le r$, $g(k,\beta) = ((\beta_1,\dots,\beta_{n_1}), (\beta_{n_1+1},\dots,\beta_{n_2}),\dots,(\beta_{n_{\ell-1}+1},\dots,\beta_{n_\ell}))$ where $1 \le n_1 < n_2 < \dots < n_{\ell-1} < n_\ell = m$, $\beta_j \sim_k \beta_{j+1}$ for $n_i \le j < n_{i+1}$, and $\beta_{n_i} \not\sim_k \beta_{n_i+1}$. Note that $g(0,\beta) = ((\beta_1,\dots,\beta_m))$ and $g(r,\beta) = ((\beta_1),\dots,(\beta_m))$. Also, if S is a c.a.d. of R^r, $S^* = (S_1^*,\dots,S_m^*)$ is the c.a.d. of R^k induced by S, and β is a c.a.s. of S, then $g(k,\beta) = (\beta_1^*,\dots,\beta_m^*)$ where β_i^* is the list of those points in β which belong to $S_i^* \times R^{r-k}$.

We define now an evaluation function e. Let $\phi(x_1,\dots,x_r)$ be a standard quantifier-free formula, S a ϕ-invariant c.a.d. of R^r, β a c.a.s. of S, and let $\phi^*(x_1,\dots,x_k)$ be $(Q_{k+1}x_{k+1}) \dots (Q_r x_r)\phi(x_1,\dots,x_r)$, $0 \le k \le r$. Let $S^* = (S_1^*,\dots,S_m^*)$ be the c.a.d. of R^k induced by S, $\beta^* = (\beta_1^*,\dots,\beta_m^*) = g(k,\beta)$. Then we define $e(\phi^*,\beta_i^*)$ by induction on $r - k$, as follows. If $k = r$, then ϕ^* is ϕ, $\beta_i^* = (\beta_i)$, and we define $e(\phi^*,\beta_i^*) = v(\phi,\beta_i)$. If $k < r$, let $g(k+1,\beta_i^*) = (\hat\beta_1,\dots,\hat\beta_n) = \hat\beta$. Then each $\hat\beta_j$ is in the sequence $g(k+1,\beta)$. Let $\hat\phi(x_1,\dots,x_{k+1})$ be $(Q_{k+2}x_{k+2})\dots(Q_r x_r)\phi(x_1,\dots,x_r)$. Then we define

$$e(\phi^*,\beta_i^*) = \wedge_{j=1}^n e(\hat\phi,\hat\beta_j), \text{ if } Q_{k+1} = \forall,$$

$$e(\phi^*,\beta_i^*) = \vee_{j=1}^n e(\hat\phi,\hat\beta_j), \text{ if } Q_{k+1} = \exists.$$

Theorem 6. Let $\phi(x_1,\dots,x_r)$ be a standard quantifier-free formula, S a ϕ-invariant c.a.d. of R^r, β a cylindrical algebraic sample of S. Let $\phi^*(x_1,\dots,x_k)$ be $(Q_{k+1}x_{k+1})\dots(Q_r x_r)\phi(x_1,\dots,x_r)$, $0 \le k \le r$. If $k > 0$, let $S^* = (S_1^*,\dots,S_m^*)$ be the c.a.d. of R^k induced by S and let $g(k,\beta) = \beta^* = (\beta_1^*,\dots,\beta_m^*)$. Then $e(\phi^*,\beta_i^*) = v(\phi^*,S_i^*)$ for $1 \le i \le m$. If $k = 0$, then $e(\phi^*,\beta) = v(\phi^*)$.

Proof. By an induction on $r - k$, paralleling the definition of e and using Theorem 5.

By Theorem 6, if $k = 0$, then $e(\phi^*,\beta)$ is the truth value of ϕ^*. If $k > 0$, let $\psi = (\psi_1,\dots,\psi_m)$ be a standard definition of the c.a.d. S^*, as produced by DECOMP and let ψ^* be the formula $\vee_{e(\phi^*,\beta_i^*) = 1}\psi_i$. Then ψ^* is a standard quantifier-free formula equivalent to ϕ^*.

The function e can be computed by an algorithm based directly on the definition of e. $e(\phi^*,\beta_i^*)$ is ultimately just some Boolean function of the truth values of ϕ at the sample points β_j in the list β_i^*, that is, of the $v(\phi,\beta_j)$. It is important to note, however, that usually not all $v(\phi,\beta_j)$ need be computed. For example, if $Q_{k+1} = \forall$ then the computation of $e(\phi^*,\beta_i^*)$ can be terminated as soon as any j is found for which $e(\hat{\phi},\hat{\beta}_j) = 0$. Similarly, the computation of $v(\phi,\beta)$, β an algebraic point, is Boolean-reducible to the case in which ϕ is a standard atomic formula. This case itself amounts to determining the sign of $A(\beta_1,\dots,\beta_r)$ where A is an integral polynomial and $\beta = (\beta_1,\dots,\beta_r)$ is a real algebraic point. With β we are given an algebraic number α such that $Q(\beta_1,\dots,\beta_r) = Q(\alpha)$ and rational polynomials B_i such that $\beta_i = B_i(\alpha)$. We then obtain sign $(A(\beta_1,\dots,\beta_r))$ = sign $(A(B_1(\alpha),\dots,B_r(\alpha)))$ = sign $(C(\alpha))$ using an algorithm of [16].

In terms of the functions g and e , the evaluation algorithm can now be described as follows.

$\psi^* = \mathrm{EVAL}(\phi^*,\beta,\psi)$

Inputs: ϕ^* is a standard prenex formula $(Q_{k+1}x_{k+1})\dots(Q_r x_r)\phi(x_1,\dots,x_r)$ where $0 \le k \le r$ and ϕ is quantifier-free. β is a c.a.s. of some ϕ-invariant c.a.d. S of R^r . ψ is a standard definition of the c.a.d. S^* of R^k induced by S if $k > 0$, the null list if $k = 0$.

Output: $\psi^* = \psi^*(x_1,\dots,x_k)$ is a standard quantifier-free formula equivalent to ϕ^* .

Algorithm Description

(1) If $k > 0$ go to (2). Set $v = e(\phi^*,\beta)$. If $v = 0$ set $\psi^* \leftarrow$ "1 = 0". If $v = 1$, set $\psi^* \leftarrow$ "0 = 0" . Exit.

(2) Set $\beta^* \leftarrow g(k,\beta)$. Let $\beta^* = (\beta_1^*,\dots,\beta_m^*)$ and $\psi = (\psi_1,\dots,\psi_m)$. Set $\psi^* \leftarrow$ "1 = 0" . For $i = 1,\dots,m$ if $e(\phi^*,\beta_i^*) = 1$ set $\psi^* \leftarrow \psi_i \vee \psi^*$. Exit.

Finally we have the following quantifier elimination algorithm.

$\psi^* = \mathrm{ELIM}(\phi^*)$

Input: ϕ^* is a standard prenex formula $(Q_{k+1}x_{k+1})\dots(Q_r x_r)\phi(x_1,\dots,x_r)$ where $0 \le k \le r$ and ϕ is quantifier-free.

Output: ψ^* is a standard quantifier-free formula equivalent to ϕ^* .

Algorithm Description

(1) Determine k . Extract from ϕ the list $A = (A_1,\dots,A_m)$ of distinct non-zero polynomials occurring in ϕ.

(2) Apply DECOMP to A and k , obtaining β and ψ as outputs.

(3) Set $\psi^* \leftarrow \mathrm{EVAL}(\phi^*,\beta,\psi)$ and exit.

4. Analysis of the Algorithms. In the Introduction, we asserted that the maximum computing time of ELIM is dominated by $(mn)^{k^r} d^k$, where r is the number of variables in ϕ , m is the number of polynomials in ϕ, n is a bound on their degrees in each separate variable, and d is a bound on the lengths of their integer coefficients. Proof of this assertion is deferred to the complete paper [8]. Here I shall merely devote a few paragraphs to a few of the main aspects and considerations of this proof.

The norm of the integral polynomial A , denoted by $|A|_1$, is the sum of the absolute values of the integer coefficients of A . This "norm" is a semi-norm in the sense that it satisfies important properties $|A+B|_1 \le |A|_1 + |B|_1$ and $|A\cdot B|_1 \le |A|_1\cdot|B_1|$.

Let c be a bound for the norms of the polynomials occurring in ϕ and let A be the set of these polynomials. Then we can set $d = L(c)$, the length of the integer c .

The algorithm DECOMP generates a sequence of sets of integral polynomials, $A_1,\dots,A_r$, where $A_1 = A$ is an input. For $i \ge 1$, A_{i+1} is the projection or augmented projection of a basis for A_i . However, as remarked earlier, the basis calculation of step (4) of DECOMP is inessential, and we will simplify the analysis by assuming it is not performed. (However, our stated computing time bound would continue to hold if the coarsest squarefree basis were used there.)

It is not difficult to see that A_2 has at most $4m^2n^3$ elements, and that each element has degrees not exceeding $8n^2$ and norm not exceeding c^{8n} . Proceeding inductively, A_i has at most $(2mn)^{3^{i+1}}$ elements, with degrees no more than $(2n)^{3^{i-1}}$, and with norms no more than $c^{(2n)^{3^i}}$. Altogether the A_i contain at most $(2mn)^{3^{r+2}}$ polynomials, which have degrees at most $(2n)^{3^{r-1}}$, and coefficient lengths dominated by $d\cdot(2n)^{3^r}$.

One may also estimate the number of cells in the c.a.d. computed by DECOMP, and hence the number of algebraic points computed. In the induced c.a.d. of R there are at most $3\cdot(2mn)^{3^{r+1}}\cdot(2n)^{3^{r-1}} \le (2mn)^{3^{r+1}+3^r}$ cells. In the induced c.a.d. of R^2 there are hence at most $3\cdot(2mn)^{3^{r+1}+3^r}\cdot(2n)^{3^{r-2}} \le (2mn)^{3^{r+1}+3^r+3^{r-1}}$ cells. Inductively we find that there are at most $(2mn)^{3^{r+2}}$ cells in the c.a.d. of R^r .

One must also estimate the degrees and coefficient lengths of the various polynomials which represent algebraic numbers, the lengths of the endpoints of rational isolating intervals, the lengths of defining formulas, etc. The end result of such analysis may be summarized by the conclusion that the amount of space consumed by all data generated is dominated by $(mn)^{k_1^r} d$ for some positive integer constant k_1 .

The conclusion regarding maximum computing time is now almost immediate. For each of the subalgorithms specified in Section 3 there is a realization for which the maximum computing time is dominated by some polynomial function of the "size" of its input. This assertion is supported in part by analyses in [1],[5],[6],[11],[12],[14] and [16]. Further justification will be set forth in [8]. Thus if k_2 is a positive integer for which each subalgorithm has maximum computing time dominated by s^{k_2}, where s is the size of its input, then the maximum computing time of ELIM is dominated by $\left\{(mn)^{k_1^r} d\right\}^{k_2} \leq (mn)^{k^r} d^k$, where $k = k_1 k_2$.

5. Concluding Remarks. While the computing time bound $(mn)^{k^r} d^k$ is a tremendous improvement over previous methods, it is not one which ensures practical applicability, especially since k is likely quite large. However, it is likely that this crude bound greatly over-estimates actual computing times for many problems. For example, the number of real roots of a polynomial is generally much smaller than its degree. Furthermore, a number of practical improvements of the algorithm are possible, which will be treated in a subsequent paper. It seems likely that this algorithm will be capable of solving some significant problems where the number of variables is not too large, say $r < 10$, and implementation of the algorithm in SAC-1, [6], is in progress.

The algorithm DECOMP has applications other than quantifier elimination. It can be used, for example, for the (real) solution of a system of polynomial equations (and inequalities). If we apply DECOMP to the list of polynomials occurring in the system (after it is put in standard form) with $k = 0$, we obtain a c.a.s. $\beta = (\beta_1, \ldots, \beta_n)$. We then decide which β_i's satisfy the system. If the system has only a finite number of solutions (which is often known), then all solutions will be included among these β_i's, and if the system has a solution then at least one solution will be found. More generally, if DECOMP is applied with $k = r$ then a standard definition of the solution set is obtained, which can be used to compute any desired number of solutions in an obvious manner. Complex solutions can be obtained by replacing each complex variable in the given system with a pair of real variables and expressing each polynomial in terms of its real and imaginary parts.

References

1. Brown, W. S., On Eculid's Algorithm and the Computation of Polynomial Greatest Common Divisors, J. ACM, vol. 18, no. 4 (Oct. 1971), pp. 478-504.

2. Brown, W. S., and Traub, J. F., On Euclid's Algorithm and the Theory of Subresultants, J. ACM, vol. 18, no. 4 (Oct. 1971), pp. 505-514.

3. Cohen, P. J., Decision Procedures for Real and p-adic Fields, Comm. Pure and Applied Math., vol. XXII, no. 2 (March 1969), pp. 131-151.

4. Collins, G. E., Subresultants and Reduced Polynomial Remainder Sequences, J. ACM, vol. 14, no. 1 (Jan. 1967), pp. 128-142.

5. Collins, G. E., The Calculation of Multivariate Polynomial Resultants, J. ACM, vol. 18, no. 4 (Oct. 1971), pp. 515-532.

6. Collins, G. E., Computer Algebra of Polynomials and Rational Functions, Am. Math. Monthly, vol. 80, no. 7 (Aug.-Sept. 1973), pp. 725-755.

7. Collins, G. E., Efficient Quantifier Elimination for Elementary Algebra (abstract), Symposium on Complexity of Sequential and Parallel Numerical Algorithms, Carnegie-Mellon University, May 1973.

8. Collins, G. E., Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition, to appear.

9. Fischer, M. J., and Rabin, M. O., Super-Exponential Complexity of Presburger Arithmetic, M.I.T. MAC Tech. Memo. 43, Feb. 1974.

10. Goldhaber, J. K., and Ehrlich, G., Algebra, MacMillan Co., 1970 .

11. Heindel, L. E., Integer Arithmetic Algorithms for Polynomial Real Zero Determination, J. ACM, vol. 18, no. 4 (Oct. 1971), pp. 533-548.

12. Loos, R. G. K., and Collins, G. E., Resultant Algorithms for Exact Arithmetic on Algebraic Numbers, to appear in SIAM J. on Comp.

13. Marden, M., The Geometry of the Zeros of a Polynomial in a Complex Variable, Am. Math. Soc., Providence, 1949.

14. Musser, D. R., Algorithms for Polynomial Factorization (Ph.D. Thesis), Univ. of Wisconsin Computer Sciences Dept. Tech. Report No. 134, Sept. 1971.

15. Musser, D. R., Multivariate Polynomial Factorization, to appear in J. ACM.

16. Rubald, C. M., Algorithms for Polynomials over a Real Algebraic Numer Field (Ph.D. Thesis), Computer Sciences Dept. Tech. Report No. 206, Jan. 1974.

17. Seidenberg, A., A New Decision Method for Elementary Algebra, Annals of Math., vol. 60, no. 2 (Sept. 1954), pp. 365-374.

18. Tarski, A., A Decision Method for Elementary Algebra and Geometry, second ed., rev., Univ. of California Press, Berkeley, 1951.

19. van der Waerden, B. L., Modern Algebra, vol. I, F. Ungar Co., New York, 1953.