SIAM J. COMPUT.
Vol. 4, No. 1, March 1975

A DECISION PROCEDURE FOR THE FIRST ORDER THEORY OF REAL ADDITION WITH ORDER*

JEANNE FERRANTE† AND CHARLES RACKOFF‡

Abstract. Consider the first order theory of the real numbers with the function $+$ (plus) and the predicate $<$ (less than). Let S be the set of true sentences of this theory. We first present an elimination of quantifiers decision procedure for S, and then analyze it to show that it takes at most time $2^{2^{cn}}$, where c is a constant, to decide sentences of length n.

We next show that a given sentence does not change in truth value when each of the quantifiers is limited to range over an appropriately chosen finite set of rationals. This fact leads to a new decision procedure for S which uses at most space 2^{cn}. We also remark that our methods lead to a decision procedure for Presburger arithmetic which operates within space $2^{2^{cn}}$. These upper bounds should be compared with the results of Fischer and Rabin [2] that for some constant c, real addition requires time 2^{cn} and Presburger arithmetic requires time $2^{2^{cn}}$.

Key words. real addition, decision procedures, quantifier-bounding, elimination of quantifiers, Presburger arithmetic.

1. Introduction. In this paper we present an efficient decision procedure for the first order theory of the real numbers with the function $+$ (plus) and the predicate $<$ (less than). Of course, the decidability of the theory in question is a consequence of Tarski's theorem that the real numbers under $+, \cdot$ (times), and $<$ is decidable [5]; however, Tarski's procedure is far from efficient for the restricted theory we are interested in. We propose to exhibit a procedure which is nearly optimal in its computational efficiency. Fischer and Rabin [2] show that there is a constant $c > 0$ such that any nondeterministic Turing machine which decides real addition (even without order) requires, for almost every n, time 2^{cn} to decide some sentences of length n. We will present a deterministic procedure for the theory of addition on the ordered set of real numbers which uses at most space 2^{dn} and time $2^{2^{gn}}$ (where d and g are constants) to decide sentences of length n. Thus there appears to be a gap of approximately one exponential between upper and lower time bounds. But since the upper bound is deterministic and the lower bound is nondeterministic, this gap should be viewed in the light of a long-standing, unproved conjecture of automata theory which states that nondeterministic time t is equal in power to deterministic time 2^t.

The procedure we give replaces unbounded quantifiers by quantifiers ranging over a finite set of rationals; truth of a sentence about the real numbers will thus be determined by checking finitely many instances of a matrix. In order to prove the correctness of our procedure, we first present an elimination of quantifiers procedure with the important feature that it does not require the sentence to be put in disjunctive normal form at each quantifier elimination.

In § 2 we define the language under consideration. In § 3 we give our elimination of quantifiers procedure. Our method utilizes an idea used by Cooper [1] in deciding

* Received by the editors May 15, 1973, and in revised form April 24, 1974. This research was supported by the National Science Foundation under Grant GJ-34671.

† Project MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts. Now at Tufts University, Medford, Massachusetts 02155.

‡ Project MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

integral addition. In § 4 we show—via an analysis of § 3—that each quantifier in a formula can be replaced by a suitably bounded quantifier, and then show that the desired space bound can be achieved. In § 5 we remark on further applications of our methods.

2. Notation. We now define a language $\mathscr{L}$ of the first order predicate calculus:

$\mathscr{L}$ has *variables* $x_0, x_1, x_{10}, \cdots$ (i.e., the subscripts are written in binary);

$\mathscr{L}$ has a *constant symbol* i (written in binary) for every integer i;

$\mathscr{L}$ has *rational constant symbols* composed of integer constant symbols, that is, if a and b are nonzero integers, then (a/b) is a rational constant symbol of $\mathscr{L}$;

$\mathscr{L}$ has *terms* of the form $(a_1/b_1)y_1 + (a_2/b_2)y_2 + \cdots + (a_n/b_n)y_n$ (abbreviated $\sum_{i=1}^{n} (a_i/b_i)y_i$), where (a_i/b_i) is a rational constant for $1 \leqq i \leqq n$ and where $y_1, \cdots, y_n$ represent distinct variables of $\mathscr{L}$. The constant symbol 0 will also be considered a term of $\mathscr{L}$.

An *atomic formula* of $\mathscr{L}$ is either the string TRUE, the string FALSE, or a formula of the form $t_1 = t_2$ or of the form $t_1 < t_2$, where t_1 and t_2 are terms; the formulas and sentences of $\mathscr{L}$ are built up from the atomic formulas in the usual way using the symbols $\vee$, $\exists$, $\forall$, $\sim$, (,).

Let R be the set of real numbers. We interpret the formulas of $\mathscr{L}$ as follows: if (a/b) is a rational constant symbol and x is interpreted as having the value $r \in R$, then we give $(a/b)x$ the value $(a/b) \cdot r$. We interpret $=$ as equality, $+$ as the usual operation of addition on R, and $<$ as the usual ordering on R. The atom TRUE is always taken to be true, and FALSE is always taken to be false.

Let S be the set of sentences of $\mathscr{L}$ true under this interpretation. We will exhibit a decision procedure for S (that is, an algorithmic procedure for deciding whether an arbitrary sentence of $\mathscr{L}$ is in S or not) such that if B is a sentence of length n, the algorithm determines whether or not $B \in S$ within space 2^{dn}, where d is a constant.

A remark should be made here as to why we have defined the terms of $\mathscr{L}$ as we have; if we had only allowed *integer* coefficients in our terms, then the resulting language would have been no less powerful, yet it would have been more difficult to arrive at a decision procedure. The reason is that our definition of a term reflects the fact that R is not only an ordered group (under addition), but it is also divisible and torsion-free. That R is divisible means that for every $r \in R$ and every positive integer k, there is an $s \in R$ such that $k \cdot s = r$, i.e.,

$$\underbrace{s + s + \cdots + s}_{k \text{ times}} = r.$$

That R is torsion-free means that for every $r \in R$ and every positive integer k, there exists at most one $s \in R$ such that $k \cdot s = r$. It is because R is divisible and torsion-free that it makes sense to talk about division by positive integers, and hence multiplication by rational constants.

In fact, a close examination of our decision procedure will reveal that the only fact we use about R is that it is an ordered, divisible, torsion-free Abelian group. Hence our procedure will work as well on Q, the set of rationals (under the usual addition and order).

3. Elimination of quantifiers.

DEFINITION. Let $F_1(x_1, \cdots, x_n)$ and $F_2(x_1, \cdots, x_n)$ be formulas of $\mathscr{L}$. Then F_1 and F_2 are *equivalent* if for every $r_1, \cdots, r_n \in R, F_1(r_1, \cdots, r_n)$ is true $\Leftrightarrow F_2(r_1, \cdots, r_n)$ is true.

The goal of this section is to prove the following theorem.

THEOREM 1. *For every formula $F(x_1, \cdots, x_n)$, there exists an equivalent quantifier-free formula $F'(x_1, \cdots, x_n)$. In fact, there is an effective procedure for going from F to F'.*

It is clear how Theorem 1 leads to a decision procedure for S. To decide if a sentence F is true, one need merely find an equivalent quantifier-free sentence F'; F' will be a Boolean combination of the atoms TRUE and FALSE, which we know how to decide.

The proof of Theorem 1 is by induction on the complexity of $F(x_1, \cdots, x_n)$. If F is an atomic formula, then we can take F' to be F. If F is $F_1 \vee F_2$, then we can take F' to be $F'_1 \vee F'_2$, where F'_1 and F'_2 are quantifier-free formulas equivalent, respectively, to F_1 and F_2. If F is $\sim F_1$, then we can take F' to be $\sim F'_1$. The remaining two cases, $\forall x F_1$ and $\exists x F_1$, are handled by the following lemma, since the quantifier $\forall x$ is equivalent to $\sim\exists x \sim$.

LEMMA 1. *Let $B(x, x_1, \cdots, x_n)$ be a quantifier-free formula. Then there exists an effective procedure for obtaining another quantifier-free formula, $B'(x_1, \cdots, x_n)$, such that $B'(x_1, \cdots, x_n)$ is equivalent to $\exists x B(x, x_1, \cdots, x_n)$.*

Proof. Let $B(x, x_1, \cdots, x_n)$ be a quantifier-free formula.

Step 1. "Solve for x" in each atomic formula of B to obtain a quantifier-free formula, $D(x, x_1, \cdots, x_n)$, such that every atomic formula of D either does not involve x or is of the form (i) $x < t$, (ii) $t < x$, or (iii) $x = t$, where t is a term not involving x.

Step 2. We now make the following definitions:

Given $D(x, x_1, \cdots, x_n)$, to get $D_{-\infty}(x_1, \cdots, x_n)$ $(D_{\infty}(x_1, \cdots, x_n))$, replace

$$\begin{aligned} x < t &\text{ in } D \text{ by TRUE (FALSE)},\\ t < x &\text{ in } D \text{ by FALSE (TRUE)},\\ x = t &\text{ in } D \text{ by FALSE (FALSE)}. \end{aligned}$$

Clearly, for any real numbers $r_1, \cdots, r_n$, if r is a sufficiently small real number, then $D(r, r_1, \cdots, r_n)$ and $D_{-\infty}(r_1, \cdots, r_n)$ are equivalent. A similar statement can be made for D_{∞} for r sufficiently large.

Step 3. We will now eliminate the quantifier from $\exists x D(x, x_1, \cdots, x_n)$ using a method very similar to that used by Cooper in his decision procedure for Presburger arithmetic [1]. Let U be the set of all terms t (not involving x) such that $t < x$, $x < t$, or $x = t$ is an atomic formula of D.

LEMMA 1.1. *$\exists x D(x, x_1, \cdots, x_n)$ is equivalent to the quantifier-free formula $B'(x_1, \cdots, x_n)$ defined to be*

$$D_{-\infty} \vee D_{\infty} \vee \bigvee_{t,v \in U} D((t + v)/2, x_1, \cdots, x_n).$$

Proof. Suppose we are given real numbers $r_1, \cdots, r_n$.

$(B' \to \exists xD)$: Suppose

$$D_{-\infty} \vee D_{\infty} \vee \bigvee_{t,v \in U} D((t+v)/2, r_1, \cdots, r_n)$$

is true. If one of the disjuncts $D((t+v)/2, r_1, \cdots, r_n)$ is true, so is $\exists xD(x, r_1, \cdots, r_n)$. So suppose one of the first two disjuncts is true, say $D_{-\infty}$. (The proof for D_∞ is similar.) Then since we can pick r sufficiently small so that $D(r, r_1, \cdots, r_n)$ is equivalent to $D_{-\infty}$, $\exists xD(x, r_1, \cdots, r_n)$ is true.

$(\exists xD \to B')$: Suppose $\exists xD(x, r_1, \cdots, r_n)$ is true. Let $t_1, \cdots, t_m$ be the distinct real numbers, in increasing order, obtained by substituting $r_1, \cdots, r_n$ for $x_1, \cdots, x_n$ in the terms in U. Since $\exists xD(x, r_1, \cdots, r_n)$ is true, there is some real number r such that $D(r, r_1, \cdots, r_n)$ is true. Now r must satisfy a specific order relation with respect to the numbers $t_1, \cdots, t_m$. That is, exactly one of the following must hold:

(a) $r < t_1$,

(b) $t_m < r$,

(c) $r = t_i$ for some i, $1 \leqq i \leqq m$,

(d) $t_i < r < t_{i+1}$ for some i, $1 \leqq i \leqq m - 1$.

If any other real number r' satisfies the same order relations with respect to $t_1, \cdots, t_m$ as r, then $D(r', r_1, \cdots, r_n)$ is true. So if (a) holds, $D_{-\infty}$ must be true; if (b) holds, D_∞ must be true; if (c) holds, $D((t_i + t_i)/2, r_1, \cdots, r_n)$ must be true; if (d) holds, $D((t_i + t_{i+1})/2, r_1, \cdots, r_n)$ must be true.

So Lemma 1.1, Lemma 1 and Theorem 1 are proven. The key point of the proof was in Step 3, where (following Cooper) instead of putting the formula D in disjunctive normal form as is usually done, we replaced $\exists xD(x, x_1, \cdots, x_n)$ by (essentially) a disjunct of formulas of the form $D(t, x_1, \cdots, x_n)$ for t a term in our language.

4. Bounds on the procedure. The purpose of this section is to show that the desired space bound can be attained. In order to do this, we want to compute a space bound on the elimination of quantifiers procedure given in § 3.

It should be noted that we are using as our model of computation the deterministic, one-tape Turing machine; space bounds, or the number of tape squares used by the Turing machine, are given as a function of n, the length of the sentence the machine is deciding. As is widely known, this model is not restrictive for bounds as large as exponential, since it can simulate a multitape or nondeterministic machine in space at most the square of the space required by the more powerful model [4]. Of course, we describe our procedure informally, leaving it to the reader to convince himself or herself that straightforward implementation of our procedure on a Turing machine would achieve the claimed bounds on time and space.

Notation. If F is a formula, let $l(F)$ be the length of F and let $s(F)$ be the largest absolute value of any integral constant appearing in any rational constant in F. (We assume, for ease of computing the complexity of our procedure, that $l(F) \geqq 2$ and $s(F) \geqq 2$.) By the "length" of an integer, we merely mean its length when written out in binary.

DEFINITION. Let r be a real number and let k be a positive integer. Then r is *limited by* k, written $r \preceq k$, if r is rational and if there exist integers a, b such that $r = a/b$ and $|a| \leqq k$ and $|b| \leqq k$.

Remark. Let $r_1, r_2, \cdots, r_k$ be real numbers limited by the positive integers $w_1, w_2, \cdots, w_k$ respectively. Then $r_1 + r_2 + \cdots + r_k \preceq k \cdot w_1 \cdot w_2 . \cdots . w_k$ and $r_1 \cdot r_2 . \cdots . r_k \preceq w_1 \cdot w_2 . \cdots . w_k$. Now let $B(x, x_1, \cdots, x_k)$ be a quantifier-free formula and let $B'(x_1, \cdots, x_k)$ be the formula obtained by applying the elimination of quantifiers procedure of § 3 to $\exists xB$. Let $s_0 = s(\exists xB)$ and let $l_0 = l(\exists xB)$. We compute an upper bound on $s(B')$ in terms of s_0 and an upper bound on $l(B')$ in terms of l_0.

Step 1 of the procedure, "Solve for x," first involves putting each atomic formula of B which contains x in the form $ax = t$, or $t < ax$ or $ax < t$, where t is a term not containing x. Call the resulting formula $C(x, x_1, \cdots, x_k)$. Obtaining C involves, for each variable in each atomic formula, subtracting one rational coefficient from another. Hence by the remark above, $s(C) \leqq 2(s_0)^2$. Step 1 then entails dividing through in each atomic formula of C by the coefficient of x (if it is nonzero) to obtain the formula $D(x, x_1, \cdots, x_k)$. Clearly, $s(D) \leqq (s(C))^2 \leqq 4(s_0)^4$.

No new integer constant is created by writing down D_∞ and $D_{-\infty}$.

Step 3 of the procedure involves writing $D((t + v)/2, x_1, \cdots, x_k)$ for every pair of terms t, v in D which don't contain x. Now $s(t + v) \leqq 2 \cdot (s(D))^2$, so we have

$$s((t + v)/2) \leqq 4 \cdot (s(D))^2 \leqq (s_0)^{14}. \tag{1}$$

So $s(B') \leqq (s_0)^{14}$.

To calculate $l(B')$, note that $l(D_\infty)$ and $l(D_{-\infty})$ are both $\leqq l_0$. D looks exactly like B except that the atomic formulas have been changed, so D has no more than l_0 terms. Therefore we have to write down no more than l_0^2 formulas of the form $D((t + v)/2, x_1, \cdots, x_k)$. To determine the length of each $D((t + v)/2, x_1, \cdots, x_k)$, note that in each of the at most l_0 atomic formulas, we may have to write two terms, each term containing k rational coefficients, each numerator and denominator of each coefficient bounded in size by $(s_0)^{14}$ and in length by $14 \cdot \text{length}\,(s_0)$. So the length of each formula $D((t + v)/2, x_1, \cdots, x_k) \leqq l_0 \cdot 2 \cdot k \cdot 2 \cdot (14 \cdot \text{length}\,(s_0)) \leqq 56(l_0)^3$. So $l(B') \leqq 2l_0 + l_0^2(56(l_0)^3) \leqq (l_0)^{14}$.

We now compute the amount of space it would take to eliminate quantifiers in a formula E where $l(E) = l_0$, $s(E) = s_0$, and the number of quantifiers in E is n_0. Our analysis is similar to that given by Oppen [3] for Cooper's procedure for integral addition. We first put E in prenex normal form, using the standard algorithm but always choosing variables with the shortest subscripts possible, obtaining E'. Note that E' is of length $\leqq l_0 \log (l_0)$; this is because there are at most l_0 occurrences of variables, and thus any subscript of a variable in E will be increased in length by a factor of at most $\log (l_0)$. Note that the prenex normal form procedure does not change the number of quantifiers or the size of constants, so E' has n_0 quantifiers and $s(E') = s(E)$.

Clearly, the largest formula obtained in the course of eliminating quantifiers from E' is of length at most

$$(l_0 \log l_0)^{14^{n_0}} \leqq (l_0 \log l_0)^{14^{l_0}} \leqq 2^{2^{c_0 l_0}}$$

for some constant c_0. Also, the largest integer constant (in absolute value) encountered is at most

$$(s_0)^{14^{n_0}}.$$

Notice that if E is a sentence, then the total space used in eliminating quantifiers from E need be no more than

$$2^{2^{c_0 \cdot l(E)}}$$

Since the number of steps involved in each quantifier elimination (and also in the final step of evaluating a Boolean combination of TRUE and FALSE) is only a fixed polynomial in the total space used, we see that our procedure operates within time

$$2^{2^{c_1 \cdot l(E)}}$$

for some constant c_1.

Our next goal is to derive a new decision procedure for S which will be approximately as efficient as the previous one with respect to time but more efficient with respect to space.

DEFINITION. A quantifier Qx, where Q is $\forall$ or $\exists$, is *limited by the positive integer* k (written $Qx \preceq k$) if, instead of ranging over all real numbers, it ranges over the numbers limited by k.

LEMMA 2. *There exists a constant c such that the following is true. Let $F(x, x_1, \cdots, x_k)$ be a formula containing n quantifiers; let $s_0 = s(F)$ and let $r_1, \cdots, r_k$ be any real numbers limited by the positive integers $w_1, \cdots, w_k$, respectively; let Q be either a universal or existential quantifier. Then $QxF(x, r_1, \cdots, r_k)$ is true if and only if*

$$[Qx \preceq (s_0)^{2^{c(n+k)}}(w_1 \cdot \cdots \cdot w_k)]F(x, r_1, \cdots, r_k)$$

is true. (If $k = 0$, then we take $w_1 \cdot \cdots \cdot w_k$ to equal 1).

Proof. Since $\forall x$ is equivalent to $\sim \exists x \sim$, we may assume without loss of generality that Q is existential. Let $F'(x, x_1, \cdots, x_k)$ be the quantifier-free formula equivalent to F obtained by our quantifier elimination procedure. If we solve for x in F' and take the average of any two terms that appear, (1) tells us that every rational coefficient will be limited by $(s(F'))^{14}$.

Assume now that some value of x satisfies $F'(x, r_1, \cdots, r_k)$, where $r_i \preceq w_i$ for $1 \leqq i \leqq k$. Then some value of x satisfying $F'(x, r_1, \cdots, r_k)$ is either equal to the average of two terms obtained by solving for x in $F'(x, r_1, \cdots, r_k)$ or is 1 bigger than or 1 smaller than all such averages. It is sufficient, therefore, to show that any average is limited by

$$(s_0)^{2^{c(n+k)}}(w_1 \cdot \cdots \cdot w_k).$$

But by the above paragraph, any such average is equal to $\sum_{i=1}^k a_i r_i$ for some $a_1, \cdots, a_k$ limited by $(s(F'))^{14}$. Since $a_i r_i \preceq (s(F'))^{14} \cdot w_i$, for $1 \leqq i \leqq k$, we have $\sum_{i=1}^k a_i r_i \preceq k \cdot \prod_{i=1}^k [(s(F'))^{14} w_i]$. Since $s(F') \leqq (s_0)^{14^n}$, one can easily calculate that

$$\sum_{i=1}^{k} a_i r_i \preceq (s_0)^{2^{c(n+k)}}(w_1 \cdot \cdots \cdot w_k).$$

for some constant c.

LEMMA 3. *Let c be the constant of Lemma 2, let $Q_1 x_1 Q_2 x_2 \cdots Q_n x_n F(x_1, \cdots, x_n)$ be a sentence, where F is quantifier-free and where Q_i is $\forall$ or $\exists$ for each i, $1 \leqq i$*

$\leqq n$, and let $s_0 = s(F)$. Let $w_1 = (s_0)^{2^{cn}}$ and let $w_{k+1} = (s_0)^{2^{cn}}(w_1 \cdot \cdots \cdot w_k)$ for $1 \leqq k < n$. Then $Q_1x_1 \cdots Q_nx_nF(x_1, \cdots, x_n)$ is true if and only if $(Q_1x_1 \preceq w_1) \cdot (Q_2x_2 \preceq w_2) \cdots (Q_nx_n \preceq w_n)F(x_1, \cdots, x_n)$ is true.

Proof. The proof is immediate from Lemma 2.

THEOREM 2. *There is a constant d, and a decision procedure for S, such that to decide a sentence B of length n takes at most space 2^{dn}.* (Note that the procedure must therefore take time $\leqq 2^{2^{d'n}}$, for some constant d', because of a well-known theorem of automata theory relating time and space.)

Proof. Let B be a sentence of length n, and let $s_0 = s(B)$. Put B in prenex normal form to obtain a sentence B'. Now $l(B') \leqq n \log(n)$, $s(B') = s_0$, and B' has no more than n quantifiers, so we can assume B' looks like $Q_1x_1 \cdots Q_nx_nF(x_1, \cdots, x_n)$, where F is quantifier-free and Q_i is $\forall$ or $\exists$ for $1 \leqq i \leqq n$.

Define $w_1 = (s_0)^{2^{cn}}$ and $w_{k+1} = (s_0)^{2^{cn}}(w_1 \cdot \cdots \cdot w_k)$ for $1 \leqq k \leqq n$. Then by Lemma 3, B' is equivalent to $(Q_1x_1 \preceq w_1) \cdots (Q_nx_n \preceq w_n)F(x_1, \cdots, x_n)$. It is easy to calculate that $w_k = ((s_0)^{2^{cn}})^{2^{k-1}}$ for $1 \leqq k \leqq n$, so $w_n \leqq (s_0)^{2^{(c+1)n}}$. Since $s_0 \leqq 2^n$, we have $w_n \leqq 2^{2^{c'n}}$ for some constant c'. Note that every rational constant limited by $2^{2^{c'n}}$ can be written in space proportional to $2^{c'n}$ (since integer constants are written in binary). So B' can be decided by cycling through the set of rationals associated with each quantifier appropriately, all the time testing the truth of F on different n-tuples of rational constants. We let the reader convince himself or herself that a Turing machine implementing this outlined procedure need use only 2^{dn} tape squares for some constant d.

5. Applications. The idea of deciding truth in a particular theory as outlined above can be applied to many other theories, thereby obtaining procedures of considerable computational efficiency. That is, given a particular theory, one gives an elimination of quantifiers procedure, analyzes it to see how "large" constants can grow, and then uses this analysis and the original procedure (in a manner similar to that given above) to limit quantifiers to range over finite sets instead of an infinite domain.

In particular, we consider the efficient quantifier elimination procedure given by Cooper [1] for deciding truth in the first order theory of integer addition. Define the first order language $\mathscr{L}'$ as follows:

$\mathscr{L}'$ has variables $x_0, x_1, x_{10}, \cdots$ (i.e., the subscripts are written in binary);
for each integer i, $\mathscr{L}'$ has a *constant symbol i* (written in binary);

$\mathscr{L}'$ has *terms* of the form $a_1y_1 + \cdots + a_ky_k$, where a_i is an integer constant for $1 \leqq i \leqq k$ and where $y_1, y_2, \cdots, y_k$ are distinct formal variables;

$\mathscr{L}'$ has *atomic formulas* of the form $t_1 \leqq t_2$ (read "t_1 is less than or equal to t_2") or $a|t_1$ (read "a divides t_1"), where t_1 and t_2 are terms and a is a positive integer constant, or TRUE, or FALSE.

Sentences and formulas are built up in the usual way.

Let S' be the set of sentences of $\mathscr{L}'$ which are true of Z, the set of integers, when the symbols of $\mathscr{L}'$ are interpreted in the obvious way. Cooper decides S' by elimination quantifiers, and Oppen [3] has determined bounds for this procedure.

DEFINITION. An integer n is *limited by the positive integer k*, written $n \preceq k$, if $|n| \leqq k$.

DEFINITION. If F is a formula of $\mathscr{L}'$, then $s(F)$ is the smallest integer $\geqq 2$ such that every integer constant of F is limited by $s(F)$.

THEOREM 3 (Oppen). *There exists a constant e such that the following is true. If F is a formula of $\mathscr{L}'$ with n quantifiers, then when Cooper's procedure is applied to F, every integer constant encountered is limited by*

$$(s(F))^{2^{2^{en}}}.$$

We can now state a lemma.

LEMMA 4. *There exists a constant f such that the following is true. Let $F(x, x_1, \cdots, x_k)$ be a formula of $\mathscr{L}'$ containing n quantifiers; let $s_0 = s(F)$ and let $n_1, \cdots, n_k$ be integers limited by the positive integer w. Then $\exists x F(x, n_1, \cdots, n_k)$ is true of Z if and only if*

$$[\exists x \preceq (s_0)^{2^{2^{f(n+k)}}} \cdot w]F(x, n_1, \cdots, n_k)$$

is true of Z.

Proof. Use Theorem 3, Cooper's procedure, and an analysis similar to that given for real addition.

LEMMA 5. *There exists a constant g such that the following is true. Let B be the formula $Q_1x_1 \cdots Q_nx_nF(x_1, \cdots, x_n)$, where F is quantifier-free and Q_i is $\forall$ or $\exists$ for each i, $1 \leqq i \leqq n$; let $s_0 = s(F)$. Then B is true of Z if and only if*

$$(Q_1x_1 \preceq (s_0)^{2^{2^{gn+1}}})(Q_2x_2 \preceq (s_0)^{2^{2^{gn+2}}}) \cdots (Q_nx_n \preceq (s_0)^{2^{2^{gn+n}}})F(x_1, \cdots, x_n)$$

is true of Z.

Proof. Apply the previous lemma.

We can now state the following theorem.

THEOREM 4. *There exists a constant h and a decision procedure for S' such that to decide a sentence of length n takes at most $2^{2^{hn}}$ space.*

Remark. Theorem 4 should be compared to the following result of Fischer and Rabin [2].

THEOREM (Fischer and Rabin). *There exists a constant $j > 0$ such that any nondeterministic Turing machine which decides S' requires for almost every n time $2^{2^{jn}}$ to decide some sentences of length n.*

Acknowledgments. We would like to acknowledge some helpful suggestions by Henry Baker in the early stages of our work and the assistance of Albert Meyer in the conception and preparation of this paper.

REFERENCES

[1] C. D. COOPER, *Theorem-proving in arithmetic without multiplication*, Machine Intelligence 7, Meltzer and Michie, ed., John Wiley, 1972, pp. 91–99.

[2] M. FISCHER AND M. O. RABIN, *Super exponential complexity of Presburger arithmetic*, Project MAC Tech. Mem. 43, Mass. Inst. of Tech., Cambridge, 1974.

[3] D. C. OPPEN, *Elementary bounds for Presburger arithmetic*, 5th ACM Sympos. on the Theory of Computing, 1973, pp. 34–37.

[4] W. J. SAVITCH, *Relationships between nondeterministic and deterministic tape complexities*, J. Comput. System Sci., 4 (1970), pp. 177–191.

[5] A. TARSKI, *A Decision Method for Elementary Algebra and Geometry*, 2nd ed., University of California Press, Berkeley, 1951.