THE JOURNAL OF SYMBOLIC LOGIC
Volume 39, Number 1, March 1974

TURING MACHINES AND THE SPECTRA OF FIRST-ORDER FORMULAS

NEIL D. JONES[1] AND ALAN L. SELMAN[2]

Introduction. H. Scholz [11] defined the *spectrum* of a formula φ of first-order logic with equality to be the set of all natural numbers n for which φ has a model of cardinality n. He then asked for a characterization of spectra. Only partial progress has been made. Computational aspects of this problem have been worked on by Gunter Asser [1], A. Mostowski [9], and J. H. Bennett [2]. It is known that spectra include the Grzegorczyk class ε_*^2 and are properly included in ε_*^3. However, no progress has been made toward establishing whether spectra properly include ε_*^2, or whether spectra are closed under complementation.

A possible connection with automata theory arises from the fact that ε_*^2 contains just those sets which are accepted by deterministic linear-bounded Turing machines (Ritchie [10]). Another resemblance lies in the fact that the same two problems (closure under complement, and proper inclusion of ε_*^2) have remained open for the class of context sensitive languages for several years.

In this paper we show that these similarities are not accidental—that spectra and context sensitive languages are closely related, and that their open questions are merely special cases of a family of open questions which relate to the difference (if any) between deterministic and nondeterministic time or space bounded Turing machines.

In particular we show that spectra are just those sets which are acceptable by nondeterministic Turing machines in time 2^{cx}, where c is constant and x is the length of the input. Combining this result with results of Bennett [2], Ritchie [10], Kuroda [7], and Cook [3], we obtain the "hierarchy" of classes of sets shown in Figure 1. It is of interest to note that in all of these cases the amount of unrestricted read/write memory appears to be too small to allow diagonalization within the larger classes.

A further characterization of spectra is by means of the "spectrum automaton" described in §2 below. This model is of interest because it possesses almost no computational abilities, being limited to several read-only heads which scan the vertices of a multi-dimensional cube given to it by an "oracle."

It is well known that a nondeterministic Turing machine can be regarded as a

Received June 8, 1973.
[1] This work was partially supported by NSF grant GJ-508.
[2] The authors thank the referee for his corrections and helpful suggestions during the preparation of the manuscript.

deterministic device with a read-only "choice tape"; an input is accepted iff there exists a choice tape which causes the device to deterministically accept the input, meanwhile using the work space for computation. The spectrum automaton is a logical extension of this tradeoff—an input is accepted if there is a corresponding k-dimensional "choice-space" which will be accepted by a device with no rewriteable memory at all. Surprisingly, this device is at least as powerful as the writing push-down automata of Mager [8], which are equipped with a linear-bounded tape space and an unbounded stack.

A further result is a reduction to the tautology recognition problem—spectra are closed under complementation (and equal to deterministic 2^{cx} time bounded Turing machine sets) if the set of tautologies can be recognized deterministically by a Turing machine in polynomial time.

FIGURE 1

Class of Sets	Limited Turing Machine Class	Relation to Classes Below	Closure Properties $\cup$	$\cap$	$-$
Spectra	= Nondet, time $\leq 2^{cx}$	$\supseteq$, unknown if =	Y	Y	?
Writing PDA (Det. or Nond.)	= Det, time $\leq 2^{cx}$	$\supseteq$, unknown if =	Y	Y	Y
Context Sensitive	= Nondet, space $\leq x$	$\supseteq$, unknown if =	Y	Y	?
ε_*^2	= Det, space $\leq x$	$\supseteq$, unknown if =	Y	Y	Y
Rudimentary	None	$\supseteq$, unknown if =	Y	Y	Y
Positive Rudimentary	None	—	Y	Y	?

§**1. Spectra.** It is assumed that we have at our disposal some first-order language with equality whose grammar contains an infinite list of k-ary predicate letters $P_1^k, P_2^k, \cdots, P_i^k, \cdots, i \geq 1$, for each k. $\varphi, \psi, \cdots$ shall denote formulas of this language. $\varphi(P_1^{k_1}, \cdots, P_l^{k_l})$, $l \geq 1$, is a formula whose predicate letters consist precisely of one or more occurrences of the predicate letters $P_1^{k_1}, \cdots, P_l^{k_l}$.

Let $\mathfrak{A}$ be an interpretation of the formula φ. $\mathfrak{A}$ is a structure with domain A and k-ary relations $\mathscr{P}_i^k$ corresponding to predicate letters P_i^k occurring in φ. The interpretation of $=$ is always the identity relation. By the cardinality of a structure $\mathfrak{A}$ we mean the cardinality of its domain. A structure $\mathfrak{A}$ is finite, if its domain is.

Let N denote the set of all natural numbers. For each $k > 0$, N_k denotes the set $\{0, 1, \cdots, k\}$.

If a formula of first-order logic without equality has a model of cardinality n ($n \in N$), then it has a model of cardinality $n + 1$. This is not the case for formulas of first-order logic with equality.

DEFINITION 1.1. The spectrum of a formula φ, $\mathrm{Sp}(\varphi)$, is the set of all positive integers n for which φ has a model of cardinality n.

If $n \in \mathrm{Sp}(\varphi)$, then φ has a model with domain N_{n-1} (obtained by renaming the elements of the domain of any model of φ with cardinality n). The following Lemma

1.2 states that φ can be replaced by a formula $\varphi_0 \wedge \varphi$ so that $\mathrm{Sp}(\varphi) = \mathrm{Sp}(\varphi_0 \wedge \varphi)$ and so that, for each $n \in \mathrm{Sp}(\varphi)$, N_{n-1} can be interpreted arithmetically.

LEMMA 1.2. *There is a formula* $\varphi_0(Z^1, M^1, R^2)$ *so that, for each formula* φ *not containing* Z^1, M^1, R^2, $\mathrm{Sp}(\varphi) = \mathrm{Sp}(\varphi_0 \wedge \varphi)$, *and, for* $n \in N$, $n \in \mathrm{Sp}(\varphi_0 \wedge \varphi)$ *if and only if* $\varphi_0 \wedge \varphi$ *has a model* $\langle N_{n-1}, \mathscr{Z}^1, \mathscr{M}^1, \mathscr{R}^2, \cdots \rangle$ *such that*

(a) $\mathscr{Z}^1(x)$ *iff* $x = 0$,

(b) $\mathscr{M}^1(x)$ *iff* $x = n - 1$, *and*

(c) $\mathscr{R}^2(x, y)$ *iff* $0 \leq x < y \leq n - 1$.

PROOF. Such a formula φ_0 is given in Mostowski [9].

To increase readability, we shall make use of the following informal expressions in the construction of formulas. These expressions denote corresponding formulas as indicated.

Informal expression	Corresponding formula
$x = 0$	$\varphi_0 \wedge Z^1(x)$
$x = \max$	$\varphi_0 \wedge M^1(x)$
$x < y$	$\varphi_0 \wedge R^2(x, y)$
$x + 1 = y$	$\varphi_0 \wedge R^2(x, y) \wedge \sim\exists z[R(x, z) \wedge R(z, y)]$
$x = 1$	$\varphi_0 \wedge \exists y[y = 0 \wedge y + 1 = x]$

Furthermore, if 0 or 1 occur as constant terms in a formula $\varphi(0, 1)$, an equivalent formula may be obtained as follows, where u and v are variables not in φ:

$$\varphi_0 \wedge \exists u, v[u = 0 \wedge v = 1 \wedge \varphi(u, v)].$$

This informality is allowed because in each case the intended interpretations are models of φ_0 with domains N_n that satisfy (a), (b), and (c) above.

Notation. A finite sequence $x_1, \cdots, x_k$ is abbreviated to $\boldsymbol{x}_k$; $\boldsymbol{x}_k + \boldsymbol{y}_k$ denotes the sequence $x_1 + y_1, x_2 + y_2, \cdots, x_k + y_k$. For a fixed k and $m > 0$, and, for each i, $0 < i \leq m$, the sequence $x_{ki-k+1}, x_{ki-k+2}, \cdots, x_{ki}$ is denoted by $\boldsymbol{x}^i_{km}$; consequently $\boldsymbol{x}_{km} = \boldsymbol{x}^1_{km}\boldsymbol{x}^2_{km} \cdots \boldsymbol{x}^m_{km}$. If $c \geq 0$, the sequence $c, c, \cdots, c$ (k occurrences of c) is abbreviated as $\boldsymbol{c}_k$.

For each $k \geq 1$, $m \geq 0$, $i \leq k$, and $0 \leq x_i \leq m$, define $\rho_k(m, x_1, \cdots, x_k)$ to be the unique y for which $\boldsymbol{x}_k$ is the k-digit $(m + 1)$-ary representation of y. Then define a $(2k + 1)$-ary relation $\mathscr{succ}$ on N by $\mathscr{succ}(m, \boldsymbol{x}_k, \boldsymbol{y}_k)$ iff

(a) $0 \leq x_i \leq m$, $0 \leq y_i \leq m$, $i \leq k$, and

(b) $\rho_k(m, \boldsymbol{x}_k) + 1 = \rho_k(m, \boldsymbol{y}_k)$.

In the following, superscripts on predicate letters will be deleted if no confusion of arity can arise.

LEMMA 1.3. *There is a formula* φ_1 (succ, $\cdots$) *containing a* $(2k + 1)$*-ary predicate letter* succ *such that*

(1) *for each* $n \geq 0$, *there is a model* $\langle N_n, \mathscr{P}_1^{2k+1}, \cdots \rangle$ *of* $\varphi_1 \wedge \varphi_0$ *satisfying* (a), (b), *and* (c) *of Lemma* 1.2, *and*

(2) *for each model* $\langle N_n, \mathscr{P}_1^{2k+1}, \cdots \rangle$ *of* φ_1 *satisfying* (a), (b), *and* (c) *of Lemma* 1.2, $\mathscr{P}_1^{2k+1}(m, \boldsymbol{x}_k, \boldsymbol{y}_k)$ *iff* $\mathscr{succ}(m, \boldsymbol{x}_k, \boldsymbol{y}_k)$ *and* $m = n$.

PROOF. φ_1 is defined to be

$$\begin{aligned}&\varphi_0 \wedge \forall m, \mathbf{x}_k, \mathbf{y}_k[\mathrm{succ}(m, \mathbf{x}_k, \mathbf{y}_k) \leftrightarrow m = \max\\ &\wedge [(x_1 = y_1 \wedge \cdots \wedge x_{k-1} = y_{k-1} \wedge x_k + 1 = y_k)\\ &\quad \vee (x_1 = y_1 \wedge \cdots \wedge x_{k-2} = y_{k-2} \wedge x_{k-1} + 1 = y_{k-1} \wedge (x_k = \max \wedge y_k = 0))\\ &\quad \vee \cdots \vee (x_1 = y_1 \wedge x_2 + 1 = y_2\\ &\qquad\qquad \wedge (x_3 = \max \wedge y_3 = 0) \wedge \cdots \wedge (x_k = \max \wedge y_k = 0))\\ &\qquad \vee (x_1 + 1 = y_1 \wedge (x_2 = \max \wedge y_2 = 0) \wedge \cdots \wedge (x_k = \max \wedge y_k = 0))].\end{aligned}$$

Clearly φ_1 is defined as required.

The *transitive closure* of a $2k$-ary relation $\mathscr{P}$ on N is a $2k$-ary relation $\mathscr{P}^+$ so that $\mathscr{P}^+(x_k, y_k)$ iff there is a sequence $\mathbf{x}_k = \mathbf{x}_k^1, \cdots, \mathbf{x}_k^p = \mathbf{y}_k, p > 1$, so that, for all $i < p$, $\mathscr{P}(\mathbf{x}_k^i, \mathbf{x}_k^{i+1})$.

LEMMA 1.4. *For each formula* $\varphi(P^{2k}, \cdots)$, $k \geq 1$, *there is a formula* $\varphi^+ = \varphi \wedge \Psi(P^+, \cdots)$ *so that, for all* $n \geq 0$, *if* $\langle N_n, \mathscr{P}^{2k}, \cdots \rangle$ *is a model of* $\varphi \wedge \varphi_0$ *satisfying* (a), (b), *and* (c) *of Lemma* 1.2, *then* $\langle N_n, \mathscr{P}^{2k}, \mathscr{P}^+, \cdots \rangle$ *is a model of* φ^+ *if and only if* P^+ *is the transitive closure of* $\mathscr{P}^{2k}$ *on* N_n.

PROOF. Define φ^+ to be

$$\begin{aligned}&\varphi \wedge \varphi_1 \wedge m = \max \wedge \forall \mathbf{x}_k, \mathbf{y}_k[P^+(\mathbf{x}_k, \mathbf{y}_k) \leftrightarrow \exists \mathbf{r}_k, \mathbf{s}_k T(\mathbf{x}_k, \mathbf{r}_k, \mathbf{y}_k, \mathbf{s}_k)]\\ &\qquad \wedge \forall \mathbf{x}_k, \mathbf{r}_k, \mathbf{y}_k, \mathbf{s}_k[T(\mathbf{x}_k, \mathbf{r}_k, \mathbf{y}_k, \mathbf{s}_k) \leftrightarrow [[P^{2k}(\mathbf{x}_k, \mathbf{y}_k) \wedge \mathrm{succ}(m, \mathbf{r}_k, \mathbf{s}_k)]\\ &\qquad \vee \exists \mathbf{z}_k, \mathbf{t}_k[T(\mathbf{x}_k, \mathbf{r}_k, \mathbf{z}_k, \mathbf{t}_k) \wedge P^+(\mathbf{z}_k, \mathbf{y}_k) \wedge \mathrm{succ}(m, \mathbf{t}_k, \mathbf{s}_k)]]].\end{aligned}$$

Let $\langle N_n, \mathscr{P}^{2k}, \cdots \rangle$ be a model of $\varphi \wedge \varphi_0$ that satisfies (a), (b), and (c) of Lemma 1.2.

Let $\mathscr{P}^+$ be the transitive closure of $\mathscr{P}^{2k}$ on N_n. Referring to the definition of transitive closure, $\mathscr{P}^+(\mathbf{x}_k, \mathbf{y}_k)$ iff there is a sequence $\mathbf{x}_k = \mathbf{x}_k^1, \cdots, \mathbf{x}_k^p = \mathbf{y}_k$, $p > 1$, so that, for all $i < p$, $\mathscr{P}^{2k}(\mathbf{x}_k^i, \mathbf{x}_k^{i+1})$. Without loss of generality we way assume $p < (n + 1)^k$. Define $\mathscr{T}$ on N_n so that, for each such sequence and $i < p$, $\mathscr{T}$ contains the $4k$-tuple $(\mathbf{x}_k^i, \mathbf{s}_k^i, \mathbf{x}_k^{i+1}, \mathbf{s}_k^{i+1})$, where $\mathbf{s}_k^1, \cdots, \mathbf{s}_k^p$ is the sequence $1, \cdots, p$ expressed as k-digit $(m + 1)$-ary integers. It is easy to see that the resulting interpretation $\langle N_n, \mathscr{P}^{2k}, \mathscr{P}^+, \mathscr{T}, \cdots \rangle$ is a model of φ^+.

Conversely, if $\langle N_n, \mathscr{P}^{2k}, \mathscr{P}^+, \mathscr{T}, \cdots \rangle$ is a model of φ^+ and if $\mathscr{P}^+(\mathbf{x}_k, \mathbf{y}_k)$ it is possible to obtain a sequence $\mathbf{x}_k^1, \cdots, \mathbf{x}_k^p$ as required by the following algorithm: Let $n = m + 1$, then

1. $i = 0$, $X^i = \{(\mathbf{x}_k, \mathbf{r}_k)\}$,
2. $X^{i+1} = \{(\mathbf{v}_k, \mathbf{t}_k) \mid \exists(\mathbf{u}_k, \mathbf{s}_k) \in X^i \wedge \mathscr{P}^{2k}(\mathbf{u}_k, \mathbf{v}_k) \wedge \mathit{succ}(m, \mathbf{s}_k, \mathbf{t}_k)\}$,
3. if $(\mathbf{y}_k, \mathbf{t}_k) \in X^{i+1}$ for some $\mathbf{t}_k$, or $X^{i+1} = \varnothing$, then stop,
4. otherwise increase i by one and go to step 2.

Note that if $(\mathbf{u}_k, \mathbf{s}_k) \in X^i$, then $\mathbf{s}_k$ is the k-digit $(m + 1)$-ary representation of $\mathbf{r}_k + i$, so X^i and X^j are disjoint if $i \neq j$. Thus the algorithm must halt. Furthermore, $X^{i+1} = \varnothing$ contradicts $\mathscr{T}(\mathbf{x}_k, \mathbf{r}_k, \mathbf{y}_k, \mathbf{s}_k)$. Thus $(\mathbf{y}_k, \mathbf{t}_k) \in X^{i+1}$ for some i, $\mathbf{t}_k$. From $X^0, X^1, \cdots, X^{i+1}$ a sequence $\mathbf{x}_k^1, \cdots, \mathbf{x}_k^p$ can be extracted. Hence $\mathscr{P}^+$ is the transitive closure of $\mathscr{P}^{2k}$ on N_n.

§2. Spectrum automata and their relation to spectra.

Informal description. A *scene* is a k-dimensional cubical array of symbols from a fixed alphabet. A *spectrum automaton* is a multihead nonwriting deterministic finite state automaton which scans the vertices of a scene. In a single step, the automaton reads the scene symbols scanned by each of its heads, and uses this information and its control state to decide which way (or whether) to move each of its heads, and what the next control state is to be. Further, the machine is able to sense when it is on the perimeter of the scene, and change its state accordingly. The scene is accepted if the machine eventually goes into a final state, if started on its initial state at the origin of the scene.

An integer n ($n > 0$) is *accepted* by the automaton iff there is a scene with sides of length n which is accepted. A main result of this paper is that a set of integers is a spectrum iff it is accepted by a spectrum automaton.

DEFINITION 2.1. (a) A *spectrum automaton* is a 7-tuple $M = (k, m, K, \Sigma, \delta, q_0, F)$ where $k \geq 1$ (the *dimensionality* of M), $m \geq 1$ (the *number of heads*), K is an alphabet (the *state* set), Σ is an alphabet (the set of *scene symbols*), $q_0 \in K$ (the *initial state*), $F \subseteq K$ (the set of *final states*), and δ is a function $\delta\colon (K - F) \times \Sigma^m \to K \times K \times \{-1, 0, 1\}^{km}$ (the *transition function*).

(b) An *instantaneous description* (ID for short) is a $(km + 1)$-tuple $\gamma = (q, \boldsymbol{x}_{km})$ such that $q \in K$ and $x_i \geq 0$ for $i = 1, 2, \cdots, km$.

(c) Let $n > 0$. An *n-scene* is a mapping $\Pi\colon N^k_{n-1} \to \Sigma$.

In any ID $\gamma = (q, \boldsymbol{x}_{km})$, q is the control state, $\boldsymbol{x}^1_{km}$ specifies the coordinates of the position of the first head, $\boldsymbol{x}^2_{km}$ describes the second head's position, etc. If Π is an n-scene and $0 \leq x_i < n$ ($i = 1, \cdots, k$), then the value of $\Pi(\boldsymbol{x}_k)$ is the symbol at position $(x_1, \cdots, x_k)$.

Now suppose $\delta(q, \boldsymbol{a}_m) = (r, s, \boldsymbol{d}_{km})$. The informal interpretation of this is: If in state q head i is scanning symbol a_i of Π at position $\boldsymbol{x}^i_{km}$ ($i = 1, 2, \cdots, m$), then attempt to move head 1 to position $\boldsymbol{x}^1_{km} + \boldsymbol{d}^1_{km}$, head 2 to position $\boldsymbol{x}^2_{km} + \boldsymbol{d}^2_{km}$, etc. If this is possible (i.e., the resulting head positions all lie within the n-scene Π), the next state will be r; if these displacements would move one or more heads beyond the boundary of Π, then no head motion takes place and the next state will be s. If q is a final state, the machine stops.

This informal description is now made precise.

DEFINITION 2.2. (a) Let $M = (k, m, K, \Sigma, \delta, q_0, F)$ be as in Definition 2.1, let Π be an n-scene ($n > 0$) and let $\gamma_1 = (q_1, \boldsymbol{x}_{km})$ and $\gamma_2 = (q_2, \boldsymbol{y}_{km})$ be ID's. By definition $\gamma_1 \vdash_\Pi \gamma_2$ is true iff $q_1 \in K - F$, and

(i) $\delta(q_1, \Pi(\boldsymbol{x}^1_{km}), \cdots, \Pi(\boldsymbol{x}^m_{km})) = (r, s, \boldsymbol{d}_{km})$, and *either*

(ii) $q_2 = r$ and, for each $i = 1, \cdots, km$, $x_i + d_i = y_i$ and $0 \leq y_i < n$; *or*

(iii) $q_2 = s$ and, for some j ($1 \leq j \leq km$), $x_j + d_j < 0$ or $x_j + d_j \geq n$, and $x_i = y_i$ for $i = 1, 2, \cdots, km$.

(b) $\vdash^+_\Pi$ denotes the transitive closure of $\vdash_\Pi$.

(c) An ID $\gamma = (q, \boldsymbol{x}_{km})$ is *final* iff $q \in F$.

(d) An n-scene Π is *accepted* by M iff there is a final ID γ such that $(q_0, \boldsymbol{0}_{km}) \vdash^+_\Pi \gamma$.

(e) The *set of integers accepted by M* is $S(M) = \{n \mid M \text{ accepts some } n\text{-scene } \Pi\}$.

THEOREM 2.3. *If M is a spectrum automaton, there is a formula φ_M of first-order logic with equality such that* $\mathrm{Sp}(\varphi_M) = S(M)$.

PROOF. Let $M = (k, m, K, \Sigma, \delta, q, F)$. Let $K = \{q_1, \cdots, q_r\}$ and $\Sigma = \{a_1, \cdots, a_s\}$.

Define

$$\hat{q}_i = 0, \cdots, 0, 1, 0, \cdots, 0 \quad (r \text{ terms with 1 in position } i), \quad \text{and}$$
$$\hat{a}_i = 0, \cdots, 0, 1, 0, \cdots, 0 \quad (s \text{ terms with 1 in position } i).$$

To each n-scene $\Pi: N_{n-1}^k \to \Sigma$ let correspond the $(k + s)$-ary relation:

$$\mathscr{P}\!i(\boldsymbol{x}_k, \hat{a}_i) \quad \text{iff} \quad \Pi(\boldsymbol{x}_k) = a_i.$$

A formula $\varphi_M(P, \cdots)$ is to be defined so that (1) if M accepts an n-scene Π, then $\varphi_M(P, \cdots)$ holds in a structure $\langle N_{n-1}, \mathscr{P}\!i, \cdots \rangle$ in which $\mathscr{P}\!i$ is the interpretation of P, and conversely (2) if φ_M has a model $\langle N_{n-1}, \mathscr{P}\!i, \cdots \rangle$, then $\mathscr{P}$ corresponds to an n-scene Π ($\mathscr{P} = \mathscr{P}\!i$) and Π is accepted by M.

To each ID $\gamma = (q, \boldsymbol{x}_{km})$, let correspond the $(r + km)$-tuple $\hat{\gamma} = (\hat{q}, \boldsymbol{x}_{km})$.

Suppose $q_i \in K$, $a_1, \cdots, a_m \in \Sigma$, and $\delta(q_i, \boldsymbol{a}_m) = (q_g, q_h, \boldsymbol{d}_m)$. For $j = 1, 2, \cdots, km$, let $Z^j_{i,\,\boldsymbol{a}_m}$ denote

$$\begin{array}{ll} x + 1 = y \vee (x = \max \wedge x = y), & \text{if } d_j = 1, \\ x = y, & \text{if } d_j = 0, \\ y + 1 = x \vee (x = 0 \wedge x = y), & \text{if } d_j = 1. \end{array}$$

Z will be used to describe the effects of δ upon head positions of M. The following formula will express the situation that occurs when δ directs M to move one or more heads beyond the boundaries of a scene. For these fixed values of i, $a_1, \cdots, a_m$, let $i_1, i_2, \cdots, i_u$ list all values of i such that $d_i = 1$, and let $j_1, j_2, \cdots, j_v$ list all values of j such that $d_j = -1$. Now let $Q_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km})$ denote the formula

$$(x_{i_1} = \max \vee \cdots \vee x_{i_u} = \max) \vee (x_{j_1} = 0 \vee \cdots \vee x_{j_v} = 0).$$

Now define $X_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}, \boldsymbol{y}_{km})$ to be

$$\begin{array}{l} P(\boldsymbol{x}^1_{km}, \hat{a}_1) \wedge P(\boldsymbol{x}^2_{km}, \hat{a}_2) \wedge \cdots \wedge P(\boldsymbol{x}^m_{km}, \hat{a}_m) \\ \quad \wedge\ [Q_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}) \wedge \boldsymbol{x}_{km} = \boldsymbol{y}_{km} \\ \quad\ \vee \sim Q_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}) \wedge Z^1_{i,\,\boldsymbol{a}_m}(x_1, y_1) \wedge Z^2_{i,\,\boldsymbol{a}_m}(x_2, y_2) \wedge \cdots \wedge Z^{km}_{1,\,\boldsymbol{a}_m}(x_{km}, y_{km})]. \end{array}$$

Let an n-scene Π be given, and let P be the predicate letter which corresponds to $\mathscr{P}\!i$. It is readily verified that $X_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}, \boldsymbol{y}_{km})$ will be satisfied in a model $\langle N_{n-1}, \mathscr{P}\!i, \cdots \rangle$ of φ_0 satisfying conditions (a), (b), and (c) of Lemma 1.2 iff

(i) $\Pi(\boldsymbol{x}^1_{km}) = a_1, \cdots, \Pi(\boldsymbol{x}^m_{km}) = a_m$, and

(ii) for some $p \in K$, $(q_i, \boldsymbol{x}_{km}) \vdash_\Pi (p, \boldsymbol{y}_{km})$.

Conversely, if $X_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}, \boldsymbol{y}_{km})$ is satisfied in a model $\langle N_{n-1}, \mathscr{P}, \cdots \rangle$ of φ_0 satisfying conditions (a), (b), and (c) of Lemma 1.2, then $\mathscr{P}$ corresponds to an n-scene Π satisfying conditions (i) and (ii) above.

Recalling that $\delta(q_i, \boldsymbol{a}_m) = (q_g, q_h, \boldsymbol{d}_{km})$, we now define $Y_{i,\,\boldsymbol{a}_m}(\hat{p}, \boldsymbol{x}_{km}, \hat{q}, \boldsymbol{y}_{km})$ by $X_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}, \boldsymbol{y}_{km}) \wedge \hat{p} = q_i \wedge [[Q_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}) \wedge \hat{q} = \hat{q}_h] \vee [\sim Q_{i,\,\boldsymbol{a}_m}(\boldsymbol{x}_{km}) \wedge \hat{q} = \hat{q}_g]]$.

Next, let $\varphi(P, Y, \cdots)$ be a formula defining $Y(\hat{p}, \boldsymbol{x}_{km}, \hat{q}, \boldsymbol{y}_{km})$ as the finite disjunction of $X_{i,\,\boldsymbol{a}_m}(\hat{p}, \boldsymbol{x}_{km}, \hat{q}, \boldsymbol{y}_{km})$ for every $q_i \in K - F$ and $a_1, \cdots, a_s \in \Sigma$.

It is easily seen that if $\mathscr{P}\!i$ corresponds to an n-scene Π and $\mathscr{Y}(\hat{p}, \boldsymbol{x}_{km}, \hat{q}, \boldsymbol{y}_{km})$ iff $(p, \boldsymbol{x}_{km}) \vdash_\Pi (q, \boldsymbol{y}_{km})$, then each model $\langle N_{n-1}, \mathscr{P}\!i, \mathscr{Y}, \cdots \rangle$ of φ_0 satisfying (a), (b), and (c) of Lemma 1.2 is a model of φ. Moreover, if $\langle N_{n-1}, \mathscr{P}, \mathscr{Y}, \cdots \rangle$ is a model of $\varphi \wedge \varphi_0$ satisfying conditions (a), (b), and (c) of Lemma 1.2, then there is an n-scene Π such that $\mathscr{P}$ is the relation $\mathscr{P}i$ corresponding to Π, and $\mathscr{Y}(\hat{p}, \mathbf{x}_{km}, \hat{q}, \mathbf{y}_{km})$ iff $(p, \mathbf{x}_{km}) \vdash_{\Pi} (q, \mathbf{y}_{km})$.

Let $\varphi^{+}(P, Y, Y^{+}, \cdots)$ be obtained from φ by use of Lemma 1.4. Then, by Lemma 1.4, the remarks made in the previous paragraph concerning φ, Y and $\vdash_{\Pi}$ hold, *mutatis mutandis*, for φ^{+}, Y^{+}, and $\vdash_{\Pi}^{+}$.

Finally, let φ_M be the formula

$$\varphi^{+} \wedge \exists \mathbf{y}_{km}, \hat{q}(Y^{+}(\hat{q}_1, \mathbf{0}_{km}, \hat{q}, \mathbf{y}_{km}) \wedge q \in F).$$

(The final state set F of M is finite, so "$q \in F$" is a finite disjunction.)

By Lemma 1.2 and the remarks above, if $n \in \mathrm{Sp}(\varphi_M)$, then φ_M has a model $\langle N_{n-1}, \mathscr{P}i, \mathscr{Y}, \mathscr{Y}^{+}, \cdots \rangle$ so that $\mathscr{P}i$ corresponds to an n-scene $\Pi: N_{n-1}^{k} \to \Sigma$ and $(q_1, \mathbf{0}_{km}) \vdash_{\Pi}^{+} (q, \mathbf{y}_{km})$, $q \in F$; i.e., Π, hence n, is accepted by M. Conversely, if M accepts n, then there is an n-scene Π and an accepting computation from $(q_1, \mathbf{0}_{km})$. Hence, $\langle N_{n-1}, \mathscr{P}i, \mathscr{Y}, \mathscr{Y}^{+}, \cdots \rangle$ is a model of φ_M. Thus, $n \in \mathrm{Sp}(\varphi_M)$. Thus, $S(M) = \mathrm{Sp}(\varphi_M)$.

The converse of Theorem 2.3 will be proved indirectly. However, we believe that the spectrum automaton arises as a natural model of the process (evaluation of quantifiers) by which one determines for a formula φ, whether $n \in \mathrm{Sp}(\varphi)$. Thus, we conclude this section with an example.

Let φ be $\forall x \exists y[P(y, x) \vee \sim P(x, y)]$. A finite interpretation of φ is a structure $\langle N_n, \mathscr{P} \rangle$, $\mathscr{P} \subseteq N_n^2$. Thus, we may think of a finite interpretation of φ as an $(n + 1)$-scene $\Pi: N_n^2 \to \{0, 1\}$, defined so that, for each $x, y \in N_n$, $\Pi(x, y) = 1$ if and only if $\mathscr{P}(x, y)$. Define a function $\beta: \{1, 0\}^2 \to \{1, 0\}$ so that $\beta(0, 1) = 0$ and $\beta(a_1, a_2) = 1$, otherwise. Then, for any assignment to the variables x and y, the assignment makes $[P(y, x) \vee \sim P(x, y)]$ true in $\langle N_n, \mathscr{P} \rangle$ if and only if $\beta(\Pi(y, x), \Pi(x, y)) = 1$. We construct a spectrum automaton M with the following properties.

1. M has two reading heads, corresponding to the number of occurrences of the predicate letter P in φ.
2. The dimensionality of M is 2, corresponding to the arity of the predicate letter P in φ.
3. M accepts an $(n + 1)$-scene Π iff the corresponding interpretation $\langle N_n, \mathscr{P} \rangle$ is a model of φ.
4. M acts on Π so as to systematically evaluate $P(y, x) \vee \sim P(x, y)$, for all values of x and y in N_n. Heads 1 and 2 are moved to scan the corresponding values of $\Pi(y, x)$ and $\Pi(x, y)$ for x fixed and $y = 0, \cdots, n$, evaluating $\beta(\Pi(y, x), \Pi(x, y))$ at each step.
5. The finite state of M is used to record these intermediate values of φ (i.e., $\beta(\Pi(y, x), \Pi(x, y))$) which are obtained during the scan.

Corresponding to the number of quantifiers in φ, the state set K of M contains the 13 members: $[q_1\, q_2]$, $[q_1\, q_2]'$, $[q_1]$, $[q_1]'$, f, for all $q_1, q_2 \in \{0, 1\}$. $[1\ 0]$ is the initial state, f is the final state, and $[0]'$ is a dead state.

Let a_1, a_2 range over $\{0, 1\}$. The transition function δ is given by

$$\begin{aligned}
\delta([q_1\, q_2], a_1, a_2) &= ([q_1\, t], [q_1\, t]', 1, 0, 0, 1) \quad \text{where } t = \max(q_2, \beta(a_1, a_2)),\\
\delta([q_1\, q_2]', a_1, a_2) &= ([q_1\, q_2]', [t], -1, 0, 0, -1) \quad \text{where } t = \min(q_1, q_2),\\
\delta([q_1], a_1, a_2) &= ([q_1\, 0], [q_1]', 0, 1, 1, 0),\\
\delta([1]', a_1, a_2) &= (f, f, 0, 0, 0, 0),\\
\delta([0]', a_1, a_2) &= ([0]', [0]', 0, 0, 0, 0).
\end{aligned}$$

For x fixed, in state $[q_1\, 0]$, M scans all pairs (x, y), $y = 0, 1, \cdots, n$. Heads 1 and 2 are moved to scan the corresponding values of $\Pi(y, x)$ and $\Pi(x, y)$, evaluating $\beta(\Pi(y, x), \Pi(x, y))$ at each step. If there is a $y \le n$ so that $\beta(\Pi(y, x), \Pi(x, y)) = 1$, then the state becomes $[q\, 1]$. By the time y is advanced to n, the state is $[q_1\, t]$, where $t = 1$ if, for this fixed value of x, $\exists y[P(y, x) \vee \sim P(x, y)]$ is true, and $t = 0$, otherwise.

Once y is advanced to its limit, state $[q_1\, t]'$ is entered, which resets y back to 0. When y reaches 0, the state becomes $[p]$, where $p = \min(q_1, t)$. Thus, M goes into the dead state if no y is found for which $\beta(\Pi(y, x), \Pi(x, y)) = 1$. Otherwise, in state $[p]$, x is increased by 1 and the state becomes $[1\ 0]$ again, now ready to increase y from 0 to n again. If x is at its limit (also n), then the formula φ has been evaluated and the $(n + 1)$-scene Π is accepted iff the current state is $[1]'$.

Thus we see that M accepts an $(n + 1)$-scene Π if and only if the corresponding interpretation $\langle N_n, \mathscr{P}\rangle$ is a model of φ, consequently $S(M) = \mathrm{Sp}(\varphi)$.

§3. **Turing machines related to spectra.** In the following, x is the *length* of the binary representation of a number n given to a Turing machine as input. Observe that $x \approx \log_2(n)$.

THEOREM 3.1. *If a set $S \subseteq N$ is a spectrum, then there is a nondeterministic Turing machine Z which accepts S within time 2^{cx}.*

PROOF. If S is a spectrum, then there is a formula φ with no free variables and in prenex normal form so that $\mathrm{Sp}(\varphi) = S$; i.e., φ is of the form $Q_1 y_1 Q_2 y_2 \cdots Q_l y_l \psi$, where Q_i is $\forall$ or $\exists$, $y_1, \cdots, y_l$ are the only free variables occurring in ψ, and where ψ is quantifier free.

For any $n > 0$, let A_n be the propositional formula obtained from φ by the following process:

(a) Let $B_0 = \varphi$.

(b) For $i = 1, \cdots, l$, DO: Let B_{i-1} be $Q_i y_i C$; If Q_i is $\forall$, then let B_i be $C^0 \wedge C^1 \wedge \cdots \wedge C^{n-1}$; if Q_i is $\exists$, then let B_i be $C^0 \vee C^1 \vee \cdots \vee C^{n-1}$, where C^j is the result of replacing every occurrence of the variable y_i in C by the integer j.

(c) Each atomic formula in B_l is of the form $P_i^k(a_1, \cdots, a_k)$ or of the form $b_1 = b_2$, where $b_1, b_2, a_1, \cdots, a_k \in N_{n-1}$. A_n is obtained from B_l by replacing each occurrence of the form $P_i^k(\boldsymbol{a}_k)$ by a new propositional variable $P_{i,\boldsymbol{a}_k}$ and from replacing $b_1 = b_2$ by $\perp$ (truth) or $\top$ (falsity) according to whether b_1 and b_2 are identical or not.

Three facts are apparent:

1. The number of symbols in A_n is bounded by a polynomial in n.
2. A_n is constructed from φ in time bounded by a polynomial in n.

3. $n \in \mathrm{Sp}(\varphi)$ iff A_n is truth-table satisfiable.

Now, there is a nondeterministic Turing machine which will accept A_n iff A_n is truth-table satisfiable in time p (length A_n), for some polynomial p. Thus the total time is bounded by n^c for some fixed c. That is, there is a nondeterministic Turing machine which will accept n iff $n \in \mathrm{Sp}(\varphi)$ in time $n^c = 2^{c \log n} = 2^{cx}$.

The following result gives yet more evidence of the centrality of the problem "is the set of tautologies recognizable in polynomial time?" as stated in Cook [3].

THEOREM 3.2. *If the set of tautologies is recognizable in polynomial time, then every spectrum is recognizable by a deterministic Turing machine within time* 2^{cx}.

PROOF. Referring to the proof of Theorem 3.1, simply observe that if the hypothesis of Theorem 3.2 is true, then the question of whether A_n is truth-table satisfiable can be decided deterministically within time bounded by a polynomial in the length of A_n.

§4. Spectrum automata related to Turing machines. In this section it is shown that nondeterministic 2^{cx} time bounded Turing machines can be simulated by spectrum automata, thereby completing the proof of our principal result (Theorem 4.5 below). We first establish some lemmas which will simplify the simulation.

LEMMA 4.1. *Let $S \subseteq N$. The following statements are equivalent, where Z represents a Turing machine, $c > 0$, $n \in N$, and x is the number of symbols in the binary representation of n:*

(a) *For some Z and c, Z accepts S in time 2^{cx}, where an input n is presented to Z in binary notation;*

(b) *for some Z and c, Z accepts S in time n^c where an input is presented to Z in binary notation; and,*

(c) *for some Z and c, Z accepts S in time n^c, where n is presented to Z in unary notation.*

PROOF. As previously observed, $x \approx \log n$; consequently $2^{cx} \approx 2^{c \log n} = n^c$, so (a) and (b) are equivalent. That (b) and (c) are equivalent follows from the fact that the familiar algorithms for conversion between unary and binary notations require time bounded by at most a polynomial in n.

In the following, the length of a string of symbols α will be denoted by $|\alpha|$.

Let Z be a nondeterministic Turing machine with tape alphabet Σ, state set K, initial state q_0, and final state set $F \subseteq K$. Define a configuration $\alpha \in \Sigma^* K \Sigma^*$ and the yield relation $\alpha \vdash \beta$ in the usual way. Let $\#$ be a new symbol, not in $K \cup \Sigma$.

LEMMA 4.2. *Let Z accept $S \subseteq N$ in time n^c, where n is presented to Z in unary notation. Then there is a constant k so that, for all $n > 0$, $n \in S$ if and only if there is a string $\mu = \alpha_1 \# \alpha_2 \# \cdots \alpha_p \#$ such that $\alpha_i \in \Sigma^* K \Sigma^*$, $1 \leq i \leq p$, and*

(a) $\alpha_1 = q_0 1^n$,

(b) $\alpha_p \in \Sigma^* F \Sigma^*$,

(c) $\alpha_i \vdash \alpha_{i+1}$, *for* $i = 1, \cdots, p - 1$, *and*

(d) $|\mu| \leq n^k$.

PROOF. Clearly $n \in S$ iff there is such a sequence μ which satisfies (a), (b), and (c). Thus, it suffices to show that in this case (d) is also satisfied. Since Z is n^c time bounded, we may assume that $p < n^c$. Further, $|\alpha_1| = n + 1$, and generally $|\alpha_i| \leq n + i$, $1 \leq i \leq p$. Thus

$$\begin{aligned} |\mu| = p + |\alpha_1| + \cdots + |\alpha_p| &\leq p + (n + 1) + (n + 2) + \cdots + (n + p) \\ &\leq p + p(n + p) = p(p + n + 1) \\ &\leq n^c(n^c + n + 1) \leq n^k, \end{aligned}$$

for sufficiently large k, and all n.

To show that there is a spectrum automaton which accepts S, where S satisfies the hypothesis of the previous lemma, it suffices to construct a spectrum automaton which accepts n iff there is a sequence μ satisfying (a) through (d). Given μ, notice that conditions (b) and (c) can be checked by a two-head one-way finite automaton applied to μ.

The next lemma shows that if an n-scene for a spectrum automaton M is "unraveled" into a string of symbols $\pi = a_0, \cdots, a_p$, then M can scan, with a single head, $a_0, \cdots, a_p$ in order. Recall that $\rho_k(m, \boldsymbol{x}_k)$ is the number whose k-digit $(m + 1)$-ary representation is $x_1, \cdots, x_k$.

LEMMA 4.3. *Let* $\Pi: N_{n-1}^k \to \Gamma$ *be a k-dimensional n-scene, and the string* $\pi = a_0a_1, \cdots, a_{n^k-1} \in \Gamma^*$ *be defined by*

$$a_i = \Pi(\boldsymbol{x}_k) \quad \textit{iff} \quad i = \rho_k(n - 1, \boldsymbol{x}_k) \qquad (x_1, \cdots, x_k \in N_{n-1}).$$

There is a one-head spectrum automaton M which is capable of scanning Π *so that it encounters* $a_0, a_1, \cdots, a_{n^k-1}$ *in that order.*

PROOF. Clearly π is well-defined. If the head of M is at coordinates $(x_1, \cdots, x_k)$ it is scanning a_i, where $i = \rho_k(n - 1, \boldsymbol{x}_k)$; in order to move to a_{i+1} it must in effect add 1 to the k-digit number $x_1 \cdots x_k$. This can be done as follows:

1. Initially the head is at position $(\boldsymbol{0}_k)$.
2. Let the head be currently scanning position $(x_1, \cdots, x_k)$; let $j = k$.
3. Attempt to advance x_j by 1; if possible, then stop.
4. If impossible (i.e., $x_j = n - 1$) then reset x_j to 0, decrease j by 1, and go to step 2.
5. If $j = 0$ then all of π has been scanned.

Clearly this can be effected in the finite control of M (since k is fixed *a priori*).

THEOREM 4.4. *If S is accepted by a nondeterministic Turing machine in time* 2^{cx}, *then S is accepted by a spectrum automaton M.*

PROOF. By Lemma 4.1, we may assume that S is accepted by a Turing machine Z in time n^c, where n is presented in unary notation. Then, let k, Σ, K, and $\#$ be as in Lemma 4.2. We define a spectrum automaton $M = (k, 2, K', \Gamma, \delta, q_0, F')$, where the scene alphabet $\Gamma = \Sigma \cup K \cup \{\#\}$, and where K', F', and δ are implicit in the following:

1. Check whether π is of the form $\alpha_1 \# \alpha_2 \# \cdots \alpha_p \#\#^m$, where $m \geq 0$, $\alpha_i \in \Sigma^*K\Sigma^*$ $(1 \leq i \leq p)$, $\alpha_1 \in q_0\Sigma^*$, and $\alpha_p \in \Sigma^*F\Sigma^*$; if not, reject.
2. Check whether $\alpha_1 = q_0 1^n$; if not reject.
3. Check whether $\alpha_i \vdash \alpha_{i+1}$ for $i = 1, \cdots, p - 1$; if so, then go into a final state.

Step 1 can be done with one head, since the collection of all such π is regular. Step 2 can be done by two heads which move synchronously, one scanning one dimension of Π, and the other scanning α_1 in π (beginning at the second symbol).

If head 1 reaches the end of its dimension at the same time that head 2 first encounters the symbol #, then α_1 is of the required form. As remarked before, step 3 may be accomplished by a two-head, one-way finite automaton. By Lemma 4.3, M can simulate such a device. Consequently M can be constructed to behave as required.

To show that $S = S(M)$, suppose first that $n \in S$; that is, n is accepted by Z. Then there is a string $\mu = \alpha_1 \# \cdots \alpha_p \# \in \Gamma^*$ which satisfies the conditions of Lemma 4.2. Define $\pi = \mu\#^m$, where m is chosen so that $|\pi| = n^k$. Let $\pi = a_0, a_1, \cdots, a_{n^k-1}$ ($a_i \in \Gamma, 0 \leq i < n^k$), and define the n-scene Π by

$$\Pi(\boldsymbol{x}_k) = a_i \quad \text{where } i = \rho_k(n-1, \boldsymbol{x}_k), \text{ for each } x_1, \cdots, x_k \in N_{n-1}.$$

It should now be clear that π will pass the checks of steps 1, 2, 3 above, so that M will accept n. Hence $S \subseteq S(M)$.

Conversely, suppose M accepts n. Then, there is an n-scene $\Pi: N_{n-1}^k \to \Gamma$ which M accepts. Since M applies steps 1, 2, 3 above, the string π corresponding to Π, as in Lemma 4.3 above, must be of the form $\pi = \alpha_1\#\alpha_2\#\cdots\alpha_p\#\#^m$ where $m \geq 0$, $\alpha_1 = q_0 1^n$, $\alpha_p \in \Sigma^* F \Sigma^*$, and $\alpha_1 \vdash \alpha_{i+1}$ for $i = 1, 2, \cdots, p-1$. If we let $\mu = \alpha_1\#\cdots\alpha_p\#$, then conditions (a) through (d) of Lemma 4.2 are satisfied, so Z accepts n. Thus $S(M) \subseteq S$, so $S(M) = S$.

The following theorem summarizes our results.

THEOREM 4.5. *The following statements are equivalent:*

S is the spectrum of a formula of first-order logic with equality;

S is accepted by a spectrum automaton;

S is accepted by a nondeterministic Turing machine in time 2^{cx}, where c is a constant and x is the length of the input.

COROLLARY 4.6. *The class of spectra includes both the context sensitive languages and their complements. The class of spectra includes ε_*^2.*

The first statement follows from Cook [3]. The second result was stated in Bennett [2].

COROLLARY 4.7. *If the set of tautologies is recognizable deterministically in polynomial time, then nondeterministic and deterministic 2^{cx} time bounded Turing machines accept the same sets and the class of spectra is closed under complementation.*

PROOF. By Theorems 3.2 and 4.5.

Remarks.

1. It follows from Theorem 4.5 that there exist spectra that are hard to compute. More precisely, using the diagonalization technique of Hartmanis and Stearns [6], there exist sets that can be recognized in time, say, 2^{2x}, but that cannot be recognized by any 2^x time bounded Turing machine.

2. Let V_{fin} be the set of all formulas of first-order logic with equality valid in all finite structures. We indicate a proof of the result due to B.A. Trachtenbrot [12] that V_{fin} is not recursively enumerable. The emptiness problem for ε_*^2 (and for context-sensitive languages) is known to be unsolvable. Thus, by Corollary 4.6, the emptiness problem for spectra (is Sp(φ) empty?) is unsolvable. A formula φ belongs to V_{fin} if and only if Sp($\sim\varphi$) is empty. Thus, V_{fin} is not recursive. It is clear that the

complement of V_{fin} is recursively enumerable, hence V_{fin} is not recursively enumerable.

3. We make some observations about spectrum automata. First, a spectrum automaton with m heads and dimension k is equivalent to one with one head and dimension mk. Likewise, a spectrum automaton with m heads and dimension k is equivalent to one with mk heads and dimension one.

If the alphabet Σ of a spectrum automaton consists of exactly one letter, then each head can be thought of as a counter capable of holding an integer value up to n (the length of a scene). Thus, it is easy to see that a set is in ε_*^2 iff it is accepted by some one-letter one-dimensional spectrum automaton. Similarly, if nondeterministic spectrum automata are defined in the obvious way, a set is context-sensitive iff it is accepted by some nondeterministic one-letter one-dimensional spectrum automaton. Thus, the question of whether every nondeterministic one-letter spectrum automaton is equivalent to some deterministic one-letter spectrum automaton is equivalent to the LBA problem.

Theorem 4.5, and the proof of Theorem 2.3 applied to nondeterministic spectrum automata show that the class of nondeterministic spectrum automata is no more powerful than the class of spectrum automata.

The hierarchy question (do spectra properly include ε_*^2?) is the question of whether spectrum automata with arbitrary alphabets are more powerful than one-letter spectrum automata.

REFERENCES

[1] G. ASSER, *Das Repräsentenproblem im Prädikatenkalkul der ersten Stufe mit Identität*, ***Zeitschrift für mathematische Logik und Grundlagen der Mathematik***, vol. 1 (1955), pp. 252–263.

[2] J. BENNETT, *On spectra*, Doctoral Dissertation, Princeton University, Princeton, N.J., 1962.

[3] S. A. COOK, *Characterization of push-down machines in terms of time-bounded computers*, ***Journal of the Association for Computing Machinery***, vol. 18 (1971), pp. 4–18.

[4] ———, *The complexity of theorem-proving procedures*, ***Third ACM Symposium on Theory of Computing***, 1971, pp. 151–158.

[5] A. GRZEGORCZYK, *Some classes of recursive functions*, ***Rozprawy Mathematyczne***, vol. 44 (1953), pp. 1–45.

[6] J. HARTMANIS and R. E. STEARNS, *On the computational complexity of algorithms*, ***Transactions of the American Mathematical Society***, vol. 117 (1965), pp. 285–306.

[7] S. Y. KURODA, *Classes of languages and linear-bounded automata*, ***Information and Control***, vol. 7 (1964), pp. 207–223.

[8] G. MAGER, *Writing pushdown acceptors*, ***Journal of Computing and Systems Sciences***, vol. 3 (1969), pp. 276–319.

[9] A. MOSTOWSKI, *Concerning a problem of H. Scholz*, ***Zeitschrift für mathematische Logik und Grundlagen der Mathematik***, vol. 2 (1956), pp. 210–214.

[10] R. W. RITCHIE, *Classes of predictably computable functions*, ***Transactions of the American Mathematical Society***, vol. 106 (1963), pp. 139–173.

[11] H. SCHOLZ, *Problems*, this JOURNAL, vol. 17 (1952), p. 160.

[12] B. A. TRACTENBROT, *Impossibility of an algorithm for the decision problem in finite classes*, ***Doklady Akademii Nauk SSSR***, vol. 70 (1950), pp. 569–572.

PENNSYLVANIA STATE UNIVERSITY
UNIVERSITY PARK, PENNSYLVANIA 16802

FLORIDA STATE UNIVERSITY
TALLAHASSEE, FLORIDA 32306