Reprinted from INFORMATION AND COMPUTATION
All Rights Reserved by Academic Press, New York and London

Vol. 86, No. 2, June 1990
Printed in Belgium

Kolmogorov Complexity and Degrees of Tally Sets*

ERIC ALLENDER[†]

Department of Computer Science, Rutgers University, New Brunswick, New Jersey 08903

AND

OSAMU WATANABE[‡]

Department of Computer Science, Tokyo Institute of Technology, Tokyo 152, Japan

We show that either

$$E_m^p(\text{TALLY}) = E_{btt}^p(\text{TALLY})$$

or

$$E_m^p(\text{TALLY}) \subset E_{1-tt}^p(\text{TALLY}) \subset E_{2-tt}^p(\text{TALLY}) \subset E_{3-tt}^p(\text{TALLY}) \cdots,$$

where $E_r^p(\text{TALLY})$ denotes the class of sets which are equivalent to a tally set under $\leqslant_r^p$ reductions. Furthermore, the question of whether or not $E_m^p(\text{TALLY}) = E_{btt}^p(\text{TALLY})$ is equivalent to the question of whether or not NE predicates can be solved in deterministic exponential time. The proofs use the techniques of generalized Kolmogorov complexity. As corollaries to some of the main results, we obtain new results about the Kolmogorov complexity of sets in P.

1. INTRODUCTION

A recent paper by Tang and Book (1988) initiated a study of the classes of sets which are equivalent to tally sets (i.e., subsets of 0*) and sparse sets,

* A preliminary version of this work was presented at the "3rd IEEE Structure in Complexity Theory Conference, 1988."

† Supported in part by a Faculty Development Award from the Electronics Education Foundation of the American Electronics Association.

‡ This work was done while the author was visiting the Dept. of Mathematics, University of California, Santa Barbara, and was supported in part by the National Science Foundation under Grant CCR-8611980.

0890-5401/90 $3.00

under varying notions of reducibility. A number of interesting results are proved in Tang and Book (1988), and many additional questions are posed and left open. This paper investigates some of these questions, shows that they are equivalent to each other, and shows that they are also closely related to other important open questions in complexity theory.

To motivate this study, and to provide some historical context, let us first consider P/poly, the class of sets which can be recognized by circuits of polynomial size. P/poly can be characterized as the class of sets which are reducible to tally sets via $\leqslant_T^p$ or $\leqslant_{tt}^p$ reductions (see, e.g., Schöning, 1985; notions of polynomial-time reducibility such as $\leqslant_T^p$ and $\leqslant_{tt}^p$ are discussed in Ladner, Lynch, and Selman, 1975). These observations motivated a study of the classes of sets which can be reduced to sparse or tally sets under various notions of reducibility; this research is presented in (Book and Ko, 1988; Ko, 1988). Among other results, Book and Ko (1988) show that, for all k, the class of sets $\leqslant_{k+1-tt}^p$-reducible to sparse sets properly includes the class of sets $\leqslant_{k-tt}^p$-reducible to sparse sets, but that, in contrast, every set which is $\leqslant_{btt}^p$-reducible to a tally set is already $\leqslant_m^p$-reducible to a tally set.

The class of sets with *self-producible* circuits, a subclass of P/poly consisting of sets whose circuits, in a sense, have complexity no greater than the sets themselves, was studied by Ko (1985) and Hartmanis and Hemachandra (1988). In Balcázar and Book (1986) this class was shown to be equal to the class of sets which are equivalent, under $\leqslant_T^p$ reductions, to a tally set. This result of Balcázar and Book (1986) should be compared to the result of Allender and Rubinstein (1988) that $\mathbf{K}[\mathbf{log}, \mathbf{poly}]$ is equal to the class of sets which are p-isomorphic to a tally set ($\mathbf{K}[\mathbf{log}, \mathbf{poly}]$, the class of sets containing only strings of "small" generalized Kolmogorov complexity, was defined in Balcázar and Book, 1986, to be $\{L : \exists k\ L \subseteq K[k \log n, n^k]\}$. For definitions concerning p-isomorphisms, see Berman and Hartmanis, 1977.)

These results of Balcázar and Book (1986) and Allender and Rubinstein (1988) motivated Tang and Book to study sets which are *interreducible* to sparse and tally sets under different notions of reducibility. The following definitions are from Tang and Book (1988).

DEFINITION. Let $\leqslant_r^p$ denote a class of reductions, such as $\leqslant_m^p$, $\leqslant_{tt}^p$, etc. Let E_r^p(TALLY) denote the class of all sets L such that, for some set $T \subseteq 0^*$, $T \leqslant_r^p L$ and $L \leqslant_r^p T$. The class E_r^p(SPARSE) is defined similarly to be the class of sets which are interreducible under $\leqslant_r^p$ reductions to some sparse set.

Using this terminology, the two results mentioned above may be restated:

1. L has self-producible circuits iff $L \in E_T^p(\text{TALLY})$ (Balcázar and Book, 1986).

2. $\mathbf{K}[\log, \text{poly}] = E_{\text{iso}}^p(\text{TALLY})$ (Allender and Rubinstein, 1988).

In Tang and Book (1988), it was shown that $E_m^p(\text{SPARSE}) \subset E_{1-tt}^p(\text{SPARSE})$ and for all k, $E_{k-tt}^p(\text{SPARSE}) \subset E_{k+1-tt}^p(\text{SPARSE})$. However, the following questions were left open:

1. Is $E_m^p(\text{TALLY}) = E_{1-tt}^p(\text{TALLY})$?
2. Is $E_m^p(\text{TALLY}) = E_{btt}^p(\text{TALLY})$?
3. Is there some k such that $E_{k-tt}^p(\text{TALLY}) = E_{k+1-tt}^p(\text{TALLY})$?

Since the experience of Book and Ko (1988), Ko (1988), and Tang and Book (1988) had led us to expect that questions of this sort could usually be resolved using current (i.e., relativizable) techniques, we initially tried to answer these questions directly. We were somewhat surprised to discover that relativizable techniques will not suffice to answer these questions, and we were even more surprised that these questions, which may seem to be rather esoteric, are in fact equivalent formulations of basic open questions in complexity theory.

In this paper, it is shown that the open questions listed above are all equivalent. In fact, all of these questions are equivalent to the following statement: every NE predicate is solvable in exponential time.

NE predicates will be defined in Section 3. Intuitively, the question of whether or not every NE predicate is solvable in exponential time is the "witness-finding" version of the $E = NE$ question. Although the $E = NE$ question has received a great deal of attention in the research literature, the corresponding "witness-finding" question seems to have been ignored until now. Perhaps this is because the "set-recognition" question of whether or not $P = NP$ is, in fact, equivalent to the "witness-finding" question of whether or not every NP predicate is solvable in polynomial time.

This discussion is closely related to a conjecture in (Sewelson, 1983) that $E = NE \Rightarrow E = E^{NP}$. If Sewelson's conjecture is true, then the $E = NE$ question is equivalent to the question of whether or not NE predicates are E-solvable, and thus the questions about $\leqslant_{btt}^p$ degrees of tally sets which are discussed in this paper are all equivalent to $E = NE$. These connections are discussed in Sections 3 and 6.

2. Preliminaries

It is expected that the reader will be familiar with basic concepts from complexity theory, such as Turing machines, circuits, and complexity classes such as P, NP, etc. For background and definitions, see, e.g.

(Hopcroft and Ullman, 1979; Schöning, 1985). We will use E and NE to refer to DTIME($2^{O(n)}$) and NTIME($2^{O(n)}$), respectively. E^{NP} denotes the class of languages accepted by deterministic exponential-time oracle Turing machines with an oracle from NP.

For any string x, the length of x is denoted by $|x|$. For any set S, $|S|$ denotes the cardinality of S. All languages considered in this paper are subsets of $\{0,1\}^*$. For any language A, $\bar{A}$ denotes $\{0,1\}^* - A$. We will use a one–one pairing function computable in polynomial time mapping $\{0,1\}^* \times \{0,1\}^*$ onto $\{0,1\}^*$, and for inputs x and y in $\{0,1\}^*$, we will denote the output of the pairing function by $\langle x, y\rangle$. We will also need that the projection functions $\langle x, y\rangle \mapsto x$ and $\langle x, y\rangle \mapsto y$ are computable in polynomial time. We will also assume a standard mapping from $\{0,1\}^*$ onto the positive integers; namely the string x will denote the integer whose binary representation is $1x$. Thus, for example, $|x| = \lfloor \log x \rfloor$, and given strings x and y, we may say $x \leqslant y$ (which corresponds to the lexicographic ordering on $\{0,1\}^*$).

We say that $A \leqslant_m^p B$ if there is a function f computable in polynomial time, such that for all x, $x \in A \Leftrightarrow f(x) \in B$. We say that $A \leqslant_{k-tt}^p B$ if there is a function f computable in polynomial time, such that, for all x, $f(x)$ is of the form $\langle f(x)[1], f(x)[2], \ldots, f(x)[k], \alpha(x)\rangle$ where $\alpha(x)$ is a string of 2^k bits specifying a function from $\{0,1\}^k$ to $\{0,1\}$, and $x \in A \Leftrightarrow \alpha(x)(y_1, \ldots, y_k) = 1$, where $y_i = 1 \Leftrightarrow f(x)[i] \in B$. (Intuitively, $A \leqslant_{k-tt}^p B$ if there is polynomial-time routine which can accept A, given that on each input it is allowed to formulate k questions to ask oracle B.) More formal definitions are given in Ladner, Lynch, and Selman (1975).

We use the notation $f\colon A \leqslant_r^p B$ to say that f is a reduction of type r from A to B.

A $\leqslant_{k-tt}^p$ reduction f is *honest* if there is some polynomial p such that, for $1 \leqslant i \leqslant k$, $|x| \leqslant p(|f(x)[i]|)$. We call such a reduction a $\leqslant_{k-tt}^{p,h}$ reduction. Similarly, honest many–one reductions are $\leqslant_m^{p,h}$ reductions. Honest reductions have been considered before in, e.g., Homer (1987) and Joseph and Young (1985).

Generalized Kolmogorov complexity provides a framework for talking about the complexity of individual strings. The definitions we use were introduced in Hartmanis (1983): Given any Turing machine M_v, we define $K_v[s(n), t(n)]$ to be the set of all strings x such that, for some string y of length at most $s(|x|)$, M_v prints out x on input y in at most $t(|x|)$ steps. (Note that complexity is measured in terms of $|x|$, rather than in terms of $|y|$.) As was shown in Hartmanis (1983), there is a machine M_u (a *universal* Turing machine) such that, for all v there exists a constant c such that $K_v[s(n), t(n)] \subseteq K_u[s(n)+c, ct(n)\log t(n)+c]$. Dropping the subscript, we will choose some particular universal Turing machine M_u and let $K[s(n), t(n)]$ denote $K_u[s(n), t(n)]$.

Let L be any subset of Σ^*. The *ranking function* for L, denoted r_L, is defined as follows: $r_L(x) = |\{y < x : y \in L\}|$. Ranking functions were introduced in Goldberg and Sipser (1985).

A set S is said to be *P-printable* if there is an algorithm which, on input n, will run in time polynomial in n, and will print out the elements of S which have length at most n. P-printable sets were defined in Hartmanis and Yesha (1984) and were studied further in Allender and Rubinstein (1988), in connection with Kolmogorov complexity. It was shown in Allender and Rubinstein (1988) that a set is a subset of $K[t \log n, n^t]$ for some t iff it is P-isomorphic to a tally set, and that a set is P-printable iff it is P-isomorphic to a tally set in P. One of the lemmas in this paper makes use of a more specific fact about $K[t \log n, n^t]$; we prove that fact here.

THEOREM 1. *For all* $t \geq 2$, $K[t \log n, n^t]$ *is P-isomorphic to* 0^*.

Proof. Note first that there is some Turing machine M_v such that $0^* = K_v[\log n, 2n]$. Thus $0^m \in K[2 \log n, n^2]$ for all large m, and hence, for all $t \geq 2$, the number of strings in $K[t \log n, n^t]$ which lexicographically precede 1^m cannot be much less than m. (It is at least $m-k$ for some constant k which does not depend on m.) Since $K[t \log n, n^t]$ is clearly a P-printable set, it thus suffices to show that, if S is any P-printable set that is not *overly* sparse, then S is P-isomorphic to 0^*. We will show that if S is a P-printable set such that, for some c, $(r_S(1^n))^c + c > n$, then S is P-isomorphic to 0^*.

Let S be any such set. Let $T = \overline{S}$ and let $L = \Sigma^* - 0^*$. As was pointed out in Goldberg and Sipser (1985), r_S, r_T, and r_L are all computable in polynomial time, and they all have inverses computable in time polynomial in the length of their output. Let f be defined as follows: $f(x) = 0^{r_S(x)}$ if $x \in S$, and $f(x) = r_L^{-1}(r_T(x))$ if $x \notin S$. Some straightforward calculations verify that f is a P-isomorphism mapping S onto 0^*. ∎

3. NE PREDICATES

The notion of NP-completeness has been extremely useful in characterizing the complexity of many optimization problems. One key to this success is the fact that the *recognition* problem for NP-complete sets is equivalent to the problem of *constructing* a solution to an instance of the corresponding optimization problem.

Somewhat surprisingly, it is not known if the same situation holds for nondeterministic exponential time. The following definitions and results help to make this precise.

Let M be a nondeterministic Turing machine such that every configuration of M has at most two possible successor states. An *accepting computation* for a string x on M (or a *witness* for x on M) is a binary string encoding the sequence of nondeterministic moves of M on input x along some computation path leading to an accepting configuration. That is, if y is a string encoding a sequence of moves of M, then the ith bit of y will be 1 (0) iff the ith move in the sequence which involves a nondeterministic choice is resolved in favor of the higher (lower) numbered state.

DEFINITION. An *NE predicate* is a binary predicate R such that, for some nondeterministic Turing machine M which runs in time $2^{O(n)}$, $R(x, y) \Leftrightarrow y$ is an accepting computation for x on M. An *NE* predicate R is *E-solvable* if there is some function f computable in time 2^{cn} for some c such that, for all x $\exists y\, R(x, y) \Leftrightarrow R(x, f(x))$. (Note that the running time of the *NE* machine M is exponential in $|x|$, but may be linear in $|y|$ (this will be the case if M makes many moves which involve nondeterministic choice.) Thus it seems quite possible that there exists an *NE* predicate R such that $R(x, y)$ is decidable in time linear in $|\langle x, y\rangle|$, and yet R is not *E*-solvable (i.e., not solvable in time exponential in $|x|$).)

Thus R is *E*-solvable if there is an exponential-time routine which, for all x, can find a witness for x if one exists. One can in a similar way consider "*NP* predicates" and "*P*-solvability"; it is well known that $P = NP$ iff every *NP* predicate is *P*-solvable.

It is natural to wonder if every *NE* predicate is *E*-solvable iff $E = NE$. This question is closely related to the conjecture of (Sewelson, 1983) that $E = NE \Rightarrow E = E^{NP}$, as the following proposition shows:

PROPOSITION 2. $E = E^{NP} \Rightarrow$ *every NE predicate is E-solvable* $\Rightarrow E = NE$.

Proof. It is immediate from the preceding discussion that if every *NE* predicate is *E*-solvable, then $E = NE$. Thus it suffices to show that $E = E^{NP}$ implies every *NE* predicate is *E*-solvable. Let R be an *NE* predicate defined by a nondeterministic Turing machine running in time 2^{cn}. Let $L = \{\langle 0^i, w\rangle$: for some x of length at most i^c, $R(i, wx)\}$. L is in *NP*. An exponential-time machine with an oracle for L can recognize the set $L' = \{\langle i, j\rangle$: the jth bit of the lexicographically least witness for i is $1\}$. By assumption, L' is in E. It now follows easily that R is *E*-solvable. (Related observations were made by Sewelson herself; see Sewelson, 1983, p. 43.) ∎

Thus if Sewelson's conjecture is true, then all of these conditions are equivalent, and the results about $\leqslant^p_{btt}$ degrees of tally sets which are proved in this paper turn out to all be equivalent to $E = NE$.

There is also a close relationship between the E-solvability of NE predicates and the question of whether or not every infinite set in P has an infinite P-printable subset. This latter question has been considered recently in (Allender, 1989; Allender and Rubinstein, 1988). In order to formulate this relationship, it is necessary to introduce certain notions of immunity.

An infinite set S is said to be *immune* to a class of sets $\mathscr{C}$ if S has no infinite subset in $\mathscr{C}$. (Immunity has been widely studied in complexity theory and recusive function theory, e.g., Rogers, 1967; Balcázar and Schöning, 1985.) Now we wish to extend the notion of immunity to cover predicates as well.

DEFINITION. An NE predicate R is said to be *E-immune* if (1) the set $\{x : \exists y R(x, y)\}$ is infinite, and (2) for all f computable in exponential time, the set $\{x : R(x, f(x))\}$ is finite.

It is also necessary to define a function len: $2^{\Sigma^*} \to 2^N$ such that, for any language L, $\text{len}(L) = \{n$: there is a string of length n in $L\}$.

PROPOSITION 3. 1. *For every infinite set $L \in P$ there is an infinite P-printable $S \subseteq L \Leftrightarrow$ No NE predicate is E-immune.*

2. *For every infinite set $L \in P$ there is an infinite P-printable $S \subseteq L$ such that* $\text{len}(S) = \text{len}(L) \Leftrightarrow$ *every NE predicate is E-solvable.*

Proof. Part 2 is proved as part of Theorem 4. The proof of part 1 is very similar to that of part 2. ∎

Note that if L is in P, then $L \cap K[k \log n, n^k]$ is a P-printable subset of L. Since it seems unlikely that every NE predicate is E-solvable (since that would imply that $E = NE$), the preceding proposition tells us that there are probably sets in P which, for infinitely many lengths, contain only "complex" strings. More precisely, there probably exist sets L in P such that, for all k, $\text{len}(L \cap K[k \log n, n^k]) \neq \text{len}(L)$.

Note that if $P = NP$, then $E = E^{NP}$ and hence every NE predicate is E solvable, and thus every set in P has "simple" strings of every length. This is somewhat surprising, since $P = NP$ implies that sets such as $\overline{K[n/2, n^2]}$ are in P. $\overline{K[n/2, n^2]}$, in some sense, contains only complex strings; yet if $P = NP$ it also contains infinitely many "simple" strings, i.e., strings in $K[t \log n, n^t]$ for some t.

In the proofs of the main results of this paper, we actually will make use of different statements which are equivalent to the statement that all NE predicates are E-solvable. The following result deals with all of the equivalent restatements of which we make use in this paper.

THEOREM 4. *The following are equivalent:*

1. *Every NE predicate is E-solvable.*

2. *Every honest function* $f: \Sigma^* \to 0^*$ *computable in polynomial time is weakly invertible (i.e., there is a function* g *computable in polynomial time such that* $f(g(x)) = x$ *for all* $x \in \text{image}(f)$*).*

3. *For all honest polynomial-time computable* f *such that for some* t, $\text{image}(f) \subseteq K[t \log n, n^t]$, *there exists an* r *such that for all* $x \in \text{image}(f)$, $f^{-1}(x) \cap K[r \log n, n^r] \neq \emptyset$.

4. *For all length-increasing* $f: \Sigma^* \to 0^*$ *computable in polynomial time, there exists a* t *such for all* $x \in \text{image}(f)$, $f^{-1}(x) \cap K[t \log n, n^t] \neq \emptyset$.

5. $\forall L \in P\ \exists S \subseteq L$ *such that* S *is* P*-printable and* $\text{len}(S) = \text{len}(L)$.

Proof. $(1 \Rightarrow 2)$ Let $f: \Sigma^* \to 0^*$ be an honest function computable in polynomial time. Since f is assumed to be honest, there is a constant b such that $\forall i$, $[\exists x\, f(x) = 0^i \Rightarrow \exists x\ (|x| < i^b \text{ and } f(x) = 0^i)]$. Let R be the NE predicate $R(i, x) \Leftrightarrow f(x) = 0^i$ and $|x| < i^b$. Assuming that every NE predicate is E-solvable, it follows that there is a function h computable in time 2^{cn} for some c such that $\exists y R(i, y) \Leftrightarrow R(i, h(i))$. Thus f can be inverted on 0^* in time n^c using the following algorithm: on input 0^i, compute $h(i)$.

$(2 \Rightarrow 3)$ Let f and t be such that f is honest and computable in polynomial time and $f: \Sigma^* \to K[t \log n, n^t]$. By Allender and Rubinstein (1988), there is some P-isomorphism g mapping the image of f onto some tally set. Since $g \circ f: \Sigma^* \to 0^*$ is honest, by assumption, there is some function h computable in polynomial time such that $g(f(h(x))) = x$ for all x in the image of $g \circ f$. That is, for all $x \in \text{image}(f)$, $h(g(x)) \in f^{-1}(x)$, and for all such x, $g(x) \in 0^*$, and thus there is some r such that, for all $x \in \text{image}(f)$, $h(g(x)) \in f^{-1}(x) \cap K[r \log n, n^r]$.

$(3 \Rightarrow 4)$ Immediate.

$(4 \Rightarrow 5)$ Let $L \in P$. Let $f(x) = 0^{2^{|x|+1}}$ if $x \in L$, and $f(x) = 0^{2^{|x|}}$ if $x \notin L$. By assumption, there exists a t such that for all $x \in \text{image}(f)$, $f^{-1}(x) \cap K[t \log n, n^t] \neq \emptyset$. Let $S = \{x \in K[t \log n, n^t] : |f(x)| \text{ is odd}\}$. Then $S \subseteq L$, S is P-printable, and $\text{len}(S) = \text{len}(L)$.

$(5 \Rightarrow 1)$ Let R be an NE predicate defined by a nondeterministic Turing machine which runs in time 2^{cn}. Let $L = \{0^i 1 y 1 0^j : |1y10^j| = i^c + 2 \text{ and } R(i, y)\}$. L is in P. Let S be a P-printable subset of L such that $\text{len}(S) = \text{len}(L)$. Then the following routine can be executed in exponential time, and on input i it will return a string y such that $R(i, y)$ if any such y exists: On input i, print the elements of S of size at most $i^c + 2 + i$. If any element in the list has length $i^c + 2 + i$, it is of the form $0^i 1 y 1 0^j$ for some y and j such that $R(i, y)$. Output y. ∎

4. Some Basic Lemmas

In this section, we present characterizations of sets which are equivalent to tally sets under honest reductions. These characterizations play a central role in the proofs of the main results. Each characterization has essentially the same flavor: namely, if a set is in the same degree as a tally set, then the set is reducible to itself, via reductions which query only strings of low Kolmogorov complexity.

LEMMA 5. 1. $A \in E_{1-tt}^{p,h}(TALLY) \Leftrightarrow \exists t\, A \leqslant_{1-tt}^{p,h} A \cap K[t \log n, n^t]$

2. $A \in E_{btt}^{p,h}(TALLY) \Leftrightarrow \exists t\, A \leqslant_{btt}^{p,h} A \cap K[t \log n, n^t]$

3. $\exists t\, A \leqslant_{k-tt}^{p,h} A \cap K[t \log n, n^t] \Rightarrow A \in E_{k-tt}^{p,h}(TALLY)$

4. $A \in E_{k-tt}^{p,h}(TALLY) \Rightarrow \exists t\, A \leqslant_{k^2-tt}^{p,h} A \cap K[t \log n, n^t]$.

Proof. We will prove 3 and 4; these clearly imply 1 and 2.

3. Suppose $A \leqslant_{k-tt}^{p,h} A \cap K[t \log n, n^t]$. Let g be a P-isomorphism between $K[t \log n, n^t]$ and 0^*. (Such an isomorphism exists, by Theorem 1.) Let T be $g(A \cap K[t \log n, n^t])$. Clearly, $A \leqslant_{k-tt}^{p,h} T$. Also, it is easy to see that $T \leqslant_{1-tt}^{p,h} A$, using the following procedure: on input x, reject if $x \notin 0^*$, and otherwise accept iff $g^{-1}(x) \in A$. Thus $A \in E_{k-tt}^{p,h}(\text{TALLY})$.

4. Let $A \in E_{k-tt}^{p,h}(\text{TALLY})$. Thus there exists some tally set T and some reductions f and g such that $f: A \leqslant_{k-tt}^{p,h} T$ and $g: T \leqslant_{k-tt}^{p,h} A$. Note that, since g is honest and runs in polynomial time, all queries made by g on inputs from 0^* are in $K[t \log n, n^t]$ for some t. Thus $g \circ f$ is a $\leqslant_{k^2-tt}^{p,h}$ reduction from A to $A \cap K[t \log n, n^t]$. ∎

Lemma 5 says nothing about $\leqslant_m^{p,h}$ reductions. We would like to say something like "$A \in E_m^{p,h}(\text{TALLY}) \Leftrightarrow \exists t\, A \leqslant_m^{p,h} A \cap K[t \log n, n^t]$." Indeed, the forward implication does hold. Unfortunately, however, the converse is false. (*Proof.* Let T be a tally set, and let $f: T \leqslant_m^{p,h} A$. Then $f(1^*)$ is an infinite subset of $\bar{A}$. Thus for any set $A \in E_m^{p,h}(\text{TALLY})$, $\bar{A}$ has an infinite P-printable subset. On the other hand. let B be any P-immune tally set, and let $A = \{x : 0^{|x|} \notin B\}$. Clearly, $A \leqslant_m^{p,h} A \cap K[2 \log n, n^2]$, but $\bar{A}$ has no infinite P-printable subset.)

However, something like Lemma 5 does hold for sets A of the form $B \times 0^*$.

LEMMA 6. *Let A be any set of the form $B \times 0^*$ for some set B. Then*

$$A \in E_m^{p,h}(TALLY) \Leftrightarrow \exists t\, A \leqslant_m^{p,h} A \cap K[t \log n, n^t].$$

Proof. The proof of the forward direction is like part 4 of Lemma 5. We prove the reverse direction.

Let A be of the form $B \times 0^*$, and let $A \leqslant_m^{p,h} A \cap K[t \log n, n^t]$. Let g be a P-isomorphism between $K[t \log n, n^t]$ and 0^*. (Such an isomorphism exists, by Theorem 1.) Let T be $g(A \cap K[t \log n, n^t])$. Clearly, $A \leqslant_m^{p,h} T$. Also, the function

$$x \mapsto \begin{cases} g^{-1}(x) & \text{if } x \in 0^* \\ \langle x, 1^{|x|} \rangle & \text{otherwise} \end{cases}$$

is a $\leqslant_m^{p,h}$ reduction from T to A. Thus $A \in E_m^{p,h}(\text{TALLY})$. ∎

It is natural to wonder if part 4 of Lemma 5 can be improved, by replacing the "k^2" with a "k." We conjecture that no such improvement is possible. In fact, we are able to prove that no such improvement is possible for a restricted class of truth-table reductions, which we shall call *parity* reductions. The proofs of our main results make frequent use of the properties of parity reductions.

DEFINITION. $A \leqslant_{k-tt}^p$ reduction g will be called a *parity* reduction if, for all $x \in K[3 \log n, n^3]$, $g(x) = \langle x, \iota \rangle$, where ι is a string of 2^k bits denoting the identity function, and for all $x \notin K[3 \log n, n^3]$, $g(x)$ is of the form $\langle g(x)[1], g(x)[2], ..., g(x)[k], \ominus \rangle$, where $|g(x)[i]| > |x|$ and $g(x)[i] \in K[3 \log n, n^3]$ for all i, $1 \leqslant i \leqslant k$, and $\ominus$ is a string of 2^k bits denoting the function that takes the value 1 iff an even number of the strings in the set $\{g(x)[i] : 1 \leqslant i \leqslant k\}$ are in the oracle set. (There is no special significance to the number 3 in this definition. It is sufficient to choose any t such that $0^*1^* \subseteq K[t \log n, n^t]$.)

Clearly, parity reductions are a very technical notion. However, these reductions have some special properties which enable certain of our proofs to go through. The following lemma should be compared to parts 3 and 4 of Lemma 5.

LEMMA 7. *If* $A \leqslant_{k^2-tt}^p A \cap K[3 \log n, n^3]$ *via a parity reduction, then* $A \in E_{k-tt}^{p,h}(TALLY)$.

Proof. The general idea of this lemma is that if A is $\leqslant_{k^2-tt}^p$ reducible to itself via a parity reduction, then A is $\leqslant_{k-tt}^{p,h}$ reducible to a set of k-tuples of strings of low generalized Kolmogorov complexity. Also, the set of k-tuples is $\leqslant_{k-tt}^{p,h}$ reducible to A. That is, a parity $\leqslant_{k^2-tt}^p$ reduction can, in some sense, be decomposed into two $\leqslant_{k-tt}^p$ reductions.

Let $g: A \leqslant_{k^2-tt}^p A \cap K[3 \log n, n^3]$, where g is a parity $\leqslant_{k^2-tt}^p$ reduction computable in time $q(n)$ for some polynomial q.

Let $S = \{\langle x_1, x_2, ..., x_k \rangle$: for all i, $x_i \in K[3 \log n, n^3]$ and $q(|x_i|) \geqslant |\langle x_1, x_2, ..., x_k \rangle|$, and the set $\{x_i : x_i \in A\}$ has an odd number of elements$\}$.

Clearly S is $\leqslant^{p,h}_{k-tt}$ reducible to A. We need to show that A is $\leqslant^{p,h}_{k-tt}$ reducible to S.

Let h be the $\leqslant^{p,h}_{k-tt}$ reduction given by

$$h(x)=\begin{cases}\langle\langle x, x, \ldots, x\rangle, \iota\rangle, \\ \quad \text{if} \quad g(x)=\langle x, \iota\rangle; \\ \langle\langle x_1, \ldots, x_k\rangle, \langle x_{k+1}, \ldots, x_{2k}\rangle, \ldots, \langle x_{(k-1)k+1}, \ldots, x_{k^2}\rangle, \ominus\rangle, \\ \quad \text{if} \quad g(x)=\langle x_1, \ldots, x_{k^2}, \ominus\rangle.\end{cases}$$

It is easy to verify that $h: A \leqslant^{p,h}_{k-tt} S$. Also, note that S consists only of strings of low Kolmogorov complexity, since any element $\langle x_1, x_2, \ldots, x_k\rangle$ of S can be constructed from the descriptions of the x_i. Since the x_i are all in $K[3 \log n, n^3]$, it is not hard to see that for some constant c, S is a subset of $K[6k \log n + c, cn^3 \log n + c]$. It now follows from Allender and Rubinstein (1988) that S is P-isomorphic to a tally set. Thus $A \in E^{p,h}_{k-tt}(\text{TALLY})$. ∎

The lemmas presented so far in this section have dealt with honest reductions. The next lemma provides a bridge between degrees of honest reductions and degrees of unrestricted reductions.

LEMMA 8. *For any class of reduction* $\leqslant^p_r \in \{\leqslant^p_m, \leqslant^p_{btt}, \leqslant^p_{1-tt}, \leqslant^p_{2-tt}, \ldots\}$ *and any set* A, $A \in E^p_r(TALLY) \Leftrightarrow A \times 0^* \in E^{p,h}_r(TALLY)$.

Proof. We prove the result in the case of many-one reductions. The proof in the case of $\leqslant^p_{k-tt}$ reductions is similar.

($\Rightarrow$) Assume $A \in E^p_m(\text{TALLY})$. That is, there is a tally set T and there are $\leqslant^p_m$ reductions $f: A \leqslant^p_m T$, and $g: T \leqslant^p_m A$. Assume without loss of generality that the pairing function is such that $\langle T, 0^*\rangle \subseteq 0^*$, so that $T \times 0^*$ is a tally set. Let $f'(\langle x, y\rangle) = \langle f(x), 0^{|\langle x, y\rangle|}\rangle$ if $y \in 0^*$, and $f'(\langle x, y\rangle) = \langle 1, 1^{|\langle x, y\rangle|}\rangle$ if $y \notin 0^*$. Let $g'(\langle x, y\rangle) = \langle g(x), 0^{|\langle x, y\rangle|}\rangle$ if $y \in 0^*$, and $g'(\langle x, y\rangle) = \langle 1, 1^{|\langle x, y\rangle|}\rangle$ if $y \notin 0^*$. It is easy to verify that $f': A \times 0^* \leqslant^{p,h}_m T \times 0^*$, and $g': T \times 0^* \leqslant^{p,h}_m A \times 0^*$.

($\Leftarrow$) Assume $A \times 0^* \in E^{p,h}_m(\text{TALLY})$. That is, there is a tally set T and there are $\leqslant^{p,h}_m$ reductions $f: A \times 0^* \leqslant^{p,h}_m T$, and $g: T \leqslant^{p,h}_m A \times 0^*$. Let $f'(x) = f(\langle x, 0\rangle)$, and define $g'(x)$ to be equal to y if $g(x) = \langle y, 0^j\rangle$ for some j, and $g'(x) = z$ for some fixed string $z \notin A$, if $g(x) \notin \Sigma^* \times 0^*$. It is easy to verify that $f': A \leqslant^p_m T$, and $g': T \leqslant^p_m A$. ∎

COROLLARY 9. *If, for all* A,

$$\exists t\, A \leqslant^{p,h}_{btt} A \cap K[t \log n, n^t] \Rightarrow \exists r\, A \leqslant^{p,h}_m A \cap K[r \log n, n^r],$$

then $E^p_{btt}(TALLY) = E^p_m(TALLY)$.

Proof. Let $A \in E^p_{btt}(\text{TALLY})$. By Lemma 8, $A \times 0^* \in E^{p,h}_{k-tt}(\text{TALLY})$ for some k. By Lemma 5, $A \times 0^* \leqslant^{p,h}_{k^2-tt} A \times 0^* \cap K[t \log n, n^t]$ for some t. By assumption $A \times 0^* \leqslant^{p,h}_{m} A \times 0^* \cap K[r \log n, n^r]$ for some r. By Lemma 6, $A \times 0^* \in E^{p,h}_m(\text{TALLY})$, and by Lemma 8, $A \in E^p_m(\text{TALLY})$. ∎

COROLLARY 10. *If there exists a set A such that $A \leqslant^{p,h}_{1-tt} A \cap K[3 \log n, n^3]$ and*

$$\forall t\ A \times 0^* \not\leqslant^{p,h}_m A \times 0^* \cap K[t \log n, n^t]$$

then $E^p_m(TALLY) \subset E^p_{1-tt}(TALLY)$.

Proof. Let A be such that $A \leqslant^{p,h}_{1-tt} A \cap K[3 \log n, n^3]$ and

$$\forall t\ A \times 0^* \not\leqslant^{p,h}_m A \times 0^* \cap K[t \log n, n^t].$$

By Lemma 5, $A \in E^p_{1-tt}(\text{TALLY})$. By Lemma 6, $A \times 0^* \notin E^{p,h}_m(\text{TALLY})$. By Lemma 8, $A \notin E^p_m(\text{TALLY})$. ∎

COROLLARY 11. *If for all l there exists a set A such that A is reducible to $A \cap K[3 \log n, n^3]$ via a parity $\leqslant^p_{l+1-tt}$ reduction and*

$$\forall t,\ A \times 0^* \not\leqslant^{p,h}_{l-tt} A \times 0^* \cap K[t \log n, n^t]$$

then for all k, $E^p_{k-tt}(TALLY) \subset E^p_{k+1-tt}(TALLY)$.

Proof. Let k be given. By assumption there is a set A such that $A \leqslant^{p,h}_{(k+1)^2-tt} A \cap K[3 \log n, n^3]$ via a parity $\leqslant^p_{(k+1)^2-tt}$ reduction, and

$$\forall t,\ A \times 0^* \not\leqslant^{p,h}_{(k+1)^2-1-tt} A \times 0^* \cap K[t \log n, n^t].$$

By Lemma 7, $A \in E^p_{k+1-tt}(\text{TALLY})$. By Lemma 5, $A \times 0^* \notin E^{p,h}_{k-tt}(\text{TALLY})$, since $k^2 < (k+1)^2 - 1$. By Lemma 8, $A \notin E^p_{k-tt}(\text{TALLY})$. ∎

The corollaries suggest how the proofs will be structured. We will show that if every *NE* predicate is *E*-solvable, then the hypothesis of Corollary 9 is satisfied, and we will show that if not all *NE* predicates are *E*-solvable, then the hypotheses of Corollaries 10 and 11 are satisfied.

5. MAIN RESULTS

THEOREM 12. *If all NE predicates are E-solvable, then $E^p_{btt}(TALLY) = E^p_m(TALLY)$.*

Proof. By Corollary 9, it suffices to show that if all *NE* predicates are *E*-solvable, then for all sets A,

$$\exists t,\ A \leqslant^{p,h}_{btt} A \cap K[t \log n, n^t] \Rightarrow \exists r,\ A \leqslant^{p,h}_m A \cap K[r \log n, n^r].$$

Assume that all NE predicates are E-solvable, and let $f: A \leqslant^{p,h}_{k-tt} A \cap K[t \log n, n^t]$.

Recall that, for all x, $f(x)$ is of the form $\langle f(x)[1], f(x)[2], \ldots, f(x)[k], \alpha(x)\rangle$, where $\alpha(x)$ is a string of length 2^k. Without loss of generality, we may assume that there is some u such that for all i, $f(x)[i] \in K[u \log n, n^u]$. (We can do this since, if $f(x)[i] \notin K[t \log n, n^t]$, the value of "$f(x)[i] \in A \cap K[t \log n, n^t]$" is 0. We can easily find a string z in $K[u \log n, n^u] - K[t \log n, n^t]$ and set $f(x)[i]$ to z. Using this replacement, the truth value of $f(x)$ remains unchanged.)

Since all of the $f(x)[i]$ are of low Kolmogorov complexity, and since $\alpha(x)$ has bounded length, it follows that for some v, the range of f is contained in $K[v \log n, n^v]$. Since it is assumed that all NE predicates are E-solvable, it is thus the case by Theorem 4 that there exists some r such that for all x there is a $y \in K[r \log n, n^r]$ such that $f(x) = f(y)$. The routine that, on input x, searches through $K[r \log n, n^r]$ until it finds such a y, and then outputs y, is a $\leqslant^{p,h}_m$ reduction from A to $A \cap K[r \log n, n^r]$. ∎

The proof of Theorem 12 is not hard; it is more difficult to prove the separation results.

Each of the separation results involves constructing a set A with certain properties. As is usual in such constructions, these sets will be built in "stages." In order to explain how we construct these sets, some discussion is necessary.

Let g be any parity $\leqslant^p_{k-tt}$ reduction, and let S be any subset of $K[3 \log n, n^3]$. Notice that there is a (unique) set A such that $A \cap K[3 \log n, n^3] = S$ and $g: A \leqslant^p_{k-tt} A \cap K[3 \log n, n^3]$. That is, for any string y, the reduction g applied to y asks questions only about elements of $K[3 \log n, n^3]$. Put y into A iff the reduction g says to accept y, using oracle S. That is, *any* subset of $K[3 \log n, n^3]$ gives rise to some set which reduces to itself via g, and membership in any such set is entirely determined by the membership of strings of low Kolmogorov complexity. Thus we will build our set A by specifying membership in A for certain strings of low Kolmogorov complexity. The next paragraph makes this more precise.

At the start of each stage s, there will be a function A_{s-1}: $K[3 \log n, n^3] \to \{0, 1, ?\}$ such that $A_{s-1}(x) = ?$ for all but finitely many x. During stage s we will build a function A_s which is a finite extension of A_{s-1} (i.e., $A_s(x) = A_{s-1}(x)$ for all x, except for finitely many x such that $A_{s-1}(x) = ?$). Let us say that a set A is *consistent* with A_s if $g: A \leqslant^p_{k-tt} A \cap K[3 \log n, n^3]$, and $A_s(x) \neq ? \Rightarrow (A_s(x) = 1 \Leftrightarrow x \in A)$. The functions A_s will be constructed so that there is at least one set A such that, for all s, A is consistent with A_s. Any set which is consistent with all of the functions A_s will be a witness for the separation result.

During the course of the construction, there will be many strings $y \notin K[3 \log n, n^3]$ such that, for some stage s, $A_s(g(y)[j]) \neq ?$ for all j, $1 \leqslant j \leqslant k$. For any such string y, it follows that membership of y in A is determined by A_s, for any set A which is consistent with A_s. Thus we shall sometimes say that A_s guarantees that such a string y is in A or is not in A.

We are now ready to prove the first separation result.

THEOREM 13. *If not all NE predicates are E-solvable, then* $E^p_m(TALLY) \subset E^p_{1-tt}(TALLY)$.

Proof. As suggested by Corollary 10, the strategy will be to show that, if not all NE predicates are E-solvable, then there exists a set A such that $A \leqslant^{p,h}_{1-tt} A \cap K[3, \log n, n^3]$ and

$$\forall t,\ A \times 0^* \not\leqslant^{p,h}_m A \times 0^* \cap K[t \log n, n^t].$$

Recall from part 4 of Theorem 4, that if not all NE predicates are E-solvable, then there is some length-increasing $f: \Sigma^* \to 0^*$ computable in polynomial time such that for all t there exist infinitely many $x \in \text{image}(f)$ such that $f^{-1}(x) \cap K[t \log n, n^t] = \varnothing$.

The $\leqslant^p_{1-tt}$ reduction from A to $A \cap K[3 \log n, n^3]$ will be given by the function g defined by

$$g(x) = \begin{cases} \langle f(x), \ominus \rangle, & \text{if } x \notin K[3 \log n, n^3]; \\ \langle x, 1 \rangle, & \text{otherwise.} \end{cases}$$

Clearly, g is a parity $\leqslant^p_{1-tt}$ reduction. (Notice that if $g(x) = \langle f(x), \ominus \rangle$, then $x \in A \Leftrightarrow f(x) \notin A$.)

Let $f_1, f_2, \ldots$ be an enumeration of $\leqslant^p_m$ reductions, and let p_i be a polynomial bounding the running time of a machine computing f_i. Let q be a polynomial such that f is computable in time bounded by q.

We will build A in stages. At stage $s = \langle i, t \rangle$ we will guarantee that f_i is not a $\leqslant^p_m$ reduction of $A \times 0^*$ to $A \times 0^* \cap K[t \log n, n^t]$.

Initially, set $A_0(x) = ?$ for all x.

At stage $s = \langle i, t \rangle$, choose r so that, for all y, if $\langle y, 0^m \rangle \in K[t \log n, n^t]$ and $m \leqslant p_i(2q(|y|))$, then $y \in K[r \log n, n^r]$. Choose x so that $A_{s-1}(f(x)) = ?$ and $f^{-1}(f(x)) \cap K[r \log n, n^r] = \varnothing$. (This is possible since A_{s-1} is only defined on finitely many strings.)

The construction of A_s now proceeds according to one of two cases, depending on the value of $f_i(\langle x, 0 \rangle)$.

Case 1. Either $f_i(\langle x, 0 \rangle) \notin K[t \log n, n^t]$ or $f_i(\langle x, 0 \rangle)$ is not of the form $\langle y, 0^m \rangle$. In this case, set $A_s(f(x)) = 0$. This guarantees that $x \in A$, and thus $\langle x, 0 \rangle \in A \times 0^*$, but $f_i(\langle x, 0 \rangle) \notin A \times 0^* \cap K[t \log n, n^t]$.

Case 2. $f_i(\langle x, 0\rangle) \in K[t \log n, n^t]$ and $f_i(\langle x, 0\rangle) = \langle y, 0^m\rangle$ for some y and some $m \leq p_i(|\langle x, 0\rangle|)$. This case has three subcases, according to whether $A_{s-1}(y) = 0$, $A_{s-1}(y) = 1$, or $A_{s-1}(y) = ?$.

Case 2a. $A_{s-1}(y) = 0$. In this case set $A_s(f(x)) = 0$. This guarantees that $x \in A$, and thus $\langle x, 0\rangle \in A \times 0^*$, but $f_i(\langle x, 0\rangle) = \langle y, 0^m\rangle \notin A \times 0^* \cap K[t \log n, n^t]$.

Case 2b. $A_{s-1}(y) = 1$. In this case set $A_s(f(x)) = 1$. This guarantees that $x \notin A$, and thus $\langle x, 0\rangle \notin A \times 0^*$, but $f_i(\langle x, 0\rangle) = \langle y, 0^m\rangle \in A \times 0^* \cap K[t \log n, n^t]$.

Case 2c. $A_{s-1}(y) = ?$. This case has two subcases, depending on whether or not $y \in K[3 \log n, n^3]$.

Case 2c.i. If $y \in K[3 \log n, n^3]$, then set $A_s(y) = 1$ and $A_s(f(x)) = 1$. (Note that this is possible even if $f(x) = y$.) This guarantees that $x \notin A$ and thus $\langle x, 0\rangle \notin A \times 0^*$, but $f_i(\langle x, 0\rangle) = \langle y, 0^m\rangle \in A \times 0^* \cap K[t \log n, n^t]$.

Case 2c.ii. If $y \notin K[3 \log n, n^3]$, then note that $f(y) \neq f(x)$. This is because if $f(y) = f(x)$, then $y \notin K[r \log n, n^r]$ (by choice of x). Also, $q(|y|) \geq |f(x)| \geq |x|$, and thus $m \leq p_i(|\langle x, 0\rangle|) \leq p_i(2|x|) \leq p_i(2q(|y|))$. Thus $\langle y, 0^m\rangle \notin K[t \log n, n^t]$ (by choice of r), contrary to assumption.

Thus we may set $A_s(f(x)) = 1$ and $A_s(f(y)) = 0$. This guarantees that $x \notin A$ and $y \in A$. Thus $\langle x, 0\rangle \notin A \times 0^*$, but $f_i(\langle x, 0\rangle) = \langle y, 0^m\rangle \in A \times 0^* \cap K[t \log n, n^t]$. ∎

THEOREM 14. *If not all NE predicates are E-solvable, then for all k,* $E^p_{k-tt}(TALLY) \subset E^p_{k+1-tt}(TALLY)$.

Proof. By Corollary 11, it suffices to show that if not all *NE* predicates are *E*-solvable, then for all k, $\exists A$ such that $A \leq^{p,h}_{k+1-tt} A \cap K[3 \log n, n^3]$ via a parity $\leq^p_{k+1-tt}$ reduction, and

$$\forall t,\ A \times 0^* \not\leq^{p,h}_{k-tt} A \times 0^* \cap K[t \log n, n^t].$$

Recall that, if not all *NE* predicates are *E*-solvable, then there is some length-increasing $f: \Sigma^* \to 0^*$ computable in polynomial time such that for all t there exist infinitely many $x \in \text{image}(f)$ such that $f^{-1}(x) \cap K[t \log n, n^t] = \varnothing$.

The $\leq^p_{k+1-tt}$ reduction from A to $A \cap K[3 \log n, n^3]$ will be given by the function g defined by

$$g(x) = \begin{cases} \langle f(x), f(x)1, f(x)11, \ldots, f(x)1^k, \ominus\rangle, & \text{if } x \notin K[3 \log n, n^3]; \\ \langle x, \iota\rangle, & \text{otherwise.} \end{cases}$$

Clearly, g is a parity $\leq^p_{k+1-tt}$ reduction.

We will build A in stages. At stage $s=\langle i, t\rangle$ we will guarantee that f_i is not a $\leqslant^p_{k-tt}$ reduction of $A\times 0^*$ to $A\times 0^*\cap K[t\log n, n^t]$, where $f_1, f_2, \ldots$ is an enumeration of $\leqslant^p_{k-tt}$ reductions. Let p_i be a polynomial bounding the running time of a machine computing f_i. Let q be a polynomial such that f is computable in time bounded by q.

Initially, set $A_0(x)=?$ for all x.

At stage $s=\langle i, t\rangle$, choose r so that, for all z, if $\langle z, 0^m\rangle\in K[t\log n, n^t]$ and $m\leqslant p_i(2q(|z|))$, then $z\in K[r\log n, n^r]$. Choose x so that $A_{s-1}(f(x))=A_{s-1}(f(x)1)=\cdots=A_{s-1}(f(x)1^k)=?$ and $f^{-1}(f(x))\cap K[r\log n, n^r]=\varnothing$.

Our goal now is to determine membership in $A\times 0^*$ for $\langle x, 0\rangle$ and for all of the strings queried by $f_i(\langle x, 0\rangle)$ in such a way that $\langle x, 0\rangle\in A\times 0^*\Leftrightarrow f_i(\langle x, 0\rangle)$ evaluates to false with oracle $A\times 0^*\cap K[t\log n, n^t]$.

Without loss of generality, we can assume that $f_i(\langle x, 0\rangle)$ is of the form

$$\langle\langle y_1, 0^{m_1}\rangle, \langle y_2, 0^{m_2}\rangle, \ldots, \langle y_k, 0^{m_k}\rangle, \alpha\rangle,$$

for some $y_1, y_2, \ldots y_k, m_1, \ldots m_k$, where α is a string of length 2^k denoting some function from $\{0,1\}^k$ to $\{0,1\}$.

Let $Z=\{z_1, z_2, \ldots, z_l\}=\{y_j: 1\leqslant j\leqslant k \text{ and } A_{s-1}(y_j)=? \text{ and } \langle y_j, 0^{m_j}\rangle\in K[t\log n, n^t]\}$. Notice that if we construct A_s in such a way that it determines whether $z\in A$ for each $z\in Z$, then we will have determined the truth value of $f_i(x)$ when the oracle is $A\times 0^*\cap K[t\log n, n^t]$, where A is any set consistent with A_s.

Notice that, for all $z\in Z$, $f(z)\neq f(x)$. This is because if $f(z)=f(x)$, then $z\notin K[r\log n, n^r]$ (by choice of x). Also, $q(|z|)\geqslant|f(x)|\geqslant|x|$. Let m be the integer such that, for some j, $f_i(\langle x, 0\rangle)[j]=\langle z, 0^m\rangle$. Then we have $m\leqslant p_i(|\langle x, 0\rangle|)\leqslant p_i(2|x|)\leqslant p_i(2q(|z|))$. Thus $\langle z, 0^m\rangle\notin K[t\log n, n^t]$ (by choice of r), which contradicts the fact that $\langle z, 0^m\rangle\in K[t\log n, n^t]$ (since $z\in Z$).

Notice also that there is some h, $0\leqslant h\leqslant k$ such that $f(x)1^h\notin\{z_1, \ldots, z_l\}$. (There are $k+1$ choices for h, but there are only $l\leqslant k$ z's.) Fix h. Also, since range$(f)\subseteq 0^*$ and since $f(x)\neq f(z)$ for all $z\in Z$, it follows that $f(x)1^h\notin\{z, f(z), f(z)1, \ldots f(z)1^k\}$ for all $z\in Z$.

We are now ready to define A_s. Let $Y=\{f(x)1^j: j\neq h\}\cup(Z\cap K[3\log n, n^3])\cup\{f(z), f(z)1, \ldots, f(z)1^k: z\in Z-K[3\log n, n^3]\}$. Set $A_s(y)=1$ for all $y\in Y$; the construction of A_s will be complete once an assignment is made for $f(x)1^h$. (It is important to note, using the observations in the preceding paragraphs, that $f(x)1^h$ is *not* in Y, and thus we are still free to set $A_s(f(x)1^h)$ to 0 or to 1.)

Notice that A_s already determines membership in A for all strings $z\in Z$. Thus there is some $b\in\{0,1\}$ such that the truth value of $f_i(x)=b$ when the oracle is $A\times 0^*\cap K[t\log n, n^t]$, where A is any set consistent with A_s.

Now set the value of $A_s(f(x)1^h)$ so that the number of elements in the set $\{j : A_s(f(x)1^j) = 1\}$ is even iff $b = 0$.

This guarantees that $\langle x, 0\rangle \in A \times 0^* \Leftrightarrow x \in A \Leftrightarrow |\{j : A_s(f(x)1^j) = 1\}|$ is even $\Leftrightarrow b = 0 \Leftrightarrow f_i(\langle x, 0\rangle)$ evaluates to false with oracle $A \times 0^* \cap K[t \log n, n^t]$. ∎

6. Conclusions and Questions

We have defined *NE* predicates and considered the question of whether or not all *NE* predicates are solvable in exponential time. This question has natural interpretations in terms of one-way functions, and in terms of the Kolmogorov complexity of sets in *P*. If $P = NP$, then every *NE* predicate is *E*-solvable, whereas if $E \neq NE$, then not all *NE* predicates are *E*-solvable. Furthermore,

Every *NE* predicate is *NE*-solvable

$$\Rightarrow E^p_m(\text{TALLY}) = E^p_{1-tt}(\text{TALLY}) = \cdots = E^p_{btt}(\text{TALLY})$$

and

Not all *NE* predicates are *E*-solvable $\Rightarrow E^p_m(\text{TALLY})$

$$\subset E^p_{1-tt}(\text{TALLY}) \subset E^p_{2-tt}(\text{TALLY}) \subset E^p_{3-tt}(\text{TALLY})$$

$$\subset E^p_{4-tt}(\text{TALLY}) \subset \cdots \subset E^p_{btt}(\text{TALLY}).$$

Since there are oracles relative to which $P = NP$ and oracles relative to which $E \neq NE$, it follows that the question of whether or not $E^p_m(\text{TALLY}) = E^p_{btt}(\text{TALLY})$ cannot be resolved by any proof technique which relativizes.

The literature in complexity theory is rich in "hierarchies;" often the hierarchies which are of greatest interest are those which are not known to be infinite, such as the polynomial hierarchy and the Boolean hierarchy. The Boolean hierarchy is especially interesting in relation to the work reported here, since it can be defined in terms of bounded truth-table reductions (Cai et al., 1988). There are oracles relative to which the Boolean hierarchy is infinite, and oracles relative to which it collapses at any given level (Cai et al., 1988). Similar results are known for the polynomial hierarchy (Ko, 1989). The hierarchy considered here, on the other hand, is either infinite, or else it collapses to its lowest level. Such "all or nothing" results are rare in complexity theory.

We believe that the proof technique used in this paper is also interesting. It seems that it would be difficult to devise proofs for these results without

having the machinery of generalized Kolmogorov complexity available to guide the process of lemma formulation.

We mention in closing some other interesting open problems from (Tang and Book, 1988) regarding the classes of sets which are equivalent to sparse and tally sets:

1. Is $E_T^p(\text{SPARSE}) = P/\text{poly}$?
2. Is $E_{tt}^p(\text{SPARSE}) = E_T^p(\text{SPARSE})$?
3. Is $E_T^p(\text{TALLY}) \subseteq E_{tt}^p(\text{SPARSE})$?

The most interesting of these is, without doubt, the question of whether or not $E_T^p(\text{SPARSE}) = P/\text{poly}$. That is, assuming only that L is reducible to a sparse set, can one conclude that L is, in fact, *equivalent* to some sparse set? Does every set with small circuits have circuits which are of "low relative complexity" in this sense?

Finally, recall that, if the conjecture of (Sewelson, 1983) that $E = NE \Rightarrow E = E^{NP}$ is true, then the conditions considered in this paper are equivalent to $E = NE$. A number of other questions concerning conditions related to the $E = NE$ problem seem quite interesting. For instance, it is relevant to ask if

$$E = NE \Rightarrow \text{Every infinite set in } P \text{ has an infinite } P\text{-printable subset.}$$

In light of Proposition 3, this is equivalent to asking if $N = NE$ implies that no infinite NE predicate is E-immune. There are reports of recent progress on questions of this sort and on relativizations related to Sewelson's conjecture (Impagliazzo and Tardos, 1989).

ACKNOWLEDGMENTS

We thank the referees for their careful work and for their many helpful comments. We also thank Ron Book for making us aware of this problem area.

RECEIVED May 26, 1988; FINAL MANUSCRIPT RECEIVED May 15, 1989

REFERENCES

ALLENDER, E. (1989), Some consequences of the existence of pseudorandom generators, *J. Comput. System Sci.* **39**, 101–124; Extended abstract in "Proceedings, 19th Annual ACM Symposium on Theory of Computing, 1987," pp. 151–159.

ALLENDER, E., AND RUBINSTEIN, R. (1988), P-printable sets, *SIAM J. Comput.* **17**, 1193–1202.

BALCÁZAR, J., AND BOOK, R. (1986), Sets with small generalized Kolmogorov complexity, *Acta Inform.* **23**, 679–688.

BALCÁZAR, J., AND SCHÖNING, U. (1985), Bi-immune sets for complexity classes, *Math. Systems Theory* **18**, 1–10.

BERMAN, L., AND HARTMANIS, J. (1977), On isomorphisms and density of *NP* and other complete sets, *SIAM J. Comput.* **6**, 305–323.

BOOK, R., AND KO, K. (1988) On sets truth-table reducible to sparse sets, *SIAM J. Comput.* **17**, 903–919.

CAI, J., GUNDERMANN, K., HARTMANIS, J., HEMACHANDRA, L., SEWELSON, V., WAGNER, K., WECHSUNG, G. (1988), The Boolean hierarchy *I*: Structural properties, *SIAM J. Comput.* **17**, 1232–1253.

GOLDBERG, A., AND SIPSER, M. (1985), Compression and ranking, *in* "Proceedings, 17th Annual ACM Symposium on Theory of Computing," pp. 440–448.

HARTMANIS, J. (1983), Generalized Kolmogorov complexity and the structure of feasible computations, *in* "Proceedings, 24th IEEE Symposium on Foundations of Computer Science," pp. 439–445.

HARTMANIS, J., AND HEMACHANDRA, L. (1986), On sparse oracles separating feasible complexity classes, *in* "Proceedings, 3rd Annual Symposium on Theoretical Aspects of Computer Science," Lecture Notes in Computer Science, Vol. 210, pp. 321–333, Springer-Verlag, New York/Berlin.

HARTMANIS, J., AND YESHA, Y. (1984), Computation times of *NP* sets of different densities, *Theoret. Comput. Sci.* **34**, 17–32.

HOMER, S. (1987), Minimal degrees for polynomial reducibilities, *J. Assoc. Comput. Mach.* **34**, 480–491.

HOPCROFT. J., AND ULLMAN, J. (1979), "Introduction to Automata Theory, Languages, and Computation," Addison–Wesley, Reading, MA.

IMPAGLIAZZO, R., AND TARDOS, G. (1989), Decision versus search problems in super-polynomial time, *in* "Proceedings, 30th IEEE Symposium on Foundations of Computer Sciences," pp. 222–227.

JOSEPH, D., AND YOUNG, P. (1985), Some remarks on witness functions for non-polynomial and non-complete sets in *NP*, *Theoret. Comput. Sci.* **39**, 225–237.

KO, K. (1985), Continuous optimization problems and a polynomial hierarchy of real functions, *J. Complexity* **1**, 210–231.

KO, K. (1988), Distinguishing bounded reducibilities by sparse sets, *in* "Proceedings, Structure in Complexity Theory, 3rd Annual Conference, IEEE," pp. 181–191.

KO, K. (1989), Relativized polynomial time hierarchies extending exactly k levels, *SIAM J. Comput.* **18**, 392–408.

LADNER, R., LYNCH, N., AND SELMAN, A. (1975), A comparison of polynomial-time reducibilities, Theoretical Computer Science 1, 103–123.

ROGERS, H. (1967), "Theory of Recursive Functions and Effective Computability," MIT Press, Cambridge, MA.

SCHÖNING, U. (1985), "Complexity and Structure," Lecture Notes in Computer Science, Vol. 211, New York/Berlin.

SEWELSON, V. (1983), "A Study of the Structure of *NP*," Ph. D. thesis, Cornell University.

TANG, S., and Book, R. (1988), Separating polynomial-time Turing and truth-table degrees of tally sets, *in* "Proceedings, 15th International Colloquium on Automata, Languages, and Programming," Lecture Notes in Computer Science, Vol. 317, pp. 591–599, Springer-Verlag, New York/Berlin, *J. Comput. System Sci.*, to appear.

Printed by Catherine Press, Ltd., Tempelhof 41, B-8000 Brugge, Belgium