INFORMATIQUE THÉORIQUE ET APPLICATIONS

SHOUWEN TANG
RONALD V. BOOK

Reducibilities on tally and sparse sets

Informatique théorique et applications, tome 25, nº 3 (1991), p. 293-302.

<http://www.numdam.org/item?id=ITA_1991__25_3_293_0>

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 25, n° 3, 1991, p. 293 à 302)

REDUCIBILITIES ON TALLY AND SPARSE SETS (*)

by Shouwen TANG (1) and Ronald V. BOOK (2)

Communicated by P. VAN EMDE BOAS

Abstract. – *Classes of sets that are inter-reducible to tally sets by polynomial-time computable reducibilities are studied.*

Résumé. – *On considère les classes d'ensembles qui sont inter-réductibles vis-à-vis des ensembles de comptage (tally sets) par réductibilités calculables en polynôme-temps.*

1. INTRODUCTION

The notions of tally set and sparse set represent the best known examples of sets with "small information content". As such, sparse sets and tally sets have played important roles in the investigation of the structure of complexity classes and of polynomial time-bounded reducibilities. For example, the class of sets that have polynomial-size circuits is the class of sets that are Turing reducible in polynomial time to sparse sets; the class of sets with small generalized Kolmogorov complexity is precisely the class of sets that are polynomial-time isomorphic to tally sets (Allender and Rubinstein [1]); and the class of sets with self-producible circuits (as defined by Ko [8]) is precisely the class of sets that are Turing equivalent in polynomial-time to tally sets (Balcázar and Book [4]). These results, as well as those of Book and Ko [6],

(*) Received March 1990, final version August 1990.
The results presented here were announced at the Fifteenth International Colloquium on Automata, Languages, and Programming, Tampere, Finland, July 1988. This research was supported in part by the National Science Foundation under Grant CCR-8611980 and CCR89-13584. This paper was written while the first author was visiting the Department of Mathematics, University of California, Santa Barbara.
(1) Beijing Computer Institute, Beijing 100044, People's Republic of China.
(2) Department of Mathematics, University of California, Santa Barbara, Ca. 93106, U.S.A.

are considered to be steps towards understanding how different resource-bounded reducibilities can (and cannot) be used to retreive information encoded in sets with small information content.

Let SPARSE denote the set of all sparse sets and let TALLY denote the set of all tally sets. For any of the standard (*i.e.*, many-one $\leqq_m$, k-truth-table $\leqq_{k-tt}$, bounded truth-table $\leqq_{btt}$, truth-table $\leqq_{tt}$, Turing $\leqq_T$) reducibilities R and any class $\mathbf{C}$ of sets, let $P_R(\mathbf{C}) = \{A \mid \text{there exists } C \in \mathbf{C} \text{ such that } A \leqq_R^P C\}$ and let $E_R^P(\mathbf{C}) = \bigcup_{C \in \mathbf{C}} \{A \mid A \leqq_R^P C \textit{ and } C \leqq_R^P A\}$; $P_R(\mathbf{C})$ is the *reduction class* of $\mathbf{C}$ under $\leqq_R^P$. [If $\mathbf{C}$ consists of a single set, $\mathbf{C} = \{C\}$, then we write $P_R(C)$ for $P_R(\mathbf{C})$ and $E_R^P(C)$ for $E_R^P(\mathbf{C})$.]

Book and Ko [6] observed that the class of sets with polynomial size circuits, denoted P/poly, has the property that

$$P/\text{poly} = P_T\ (\text{SPARSE}) = P_{tt}\ (\text{SPARSE}) = P_{tt}\ (\text{TALLY}) = P_T\ (\text{TALLY}).$$

In addition, P_{btt} (SPARSE) $\neq P$/poly and for every $k > 0$,

$$P_{k-tt}\ (\text{SPARSE}) \neq P_{(k+1)-tt}\ (\text{SPARSE}).$$

Thus, the class P/poly can be decomposed into an infinite hierarchy of classes based on the number of queries made to sparse oracles. In contrast, Book and Ko showed that

$$P_m(\text{TALLY}) = P_{btt}(\text{TALLY}) \qquad \text{and} \qquad P_{btt}(\text{TALLY}) \neq P_{tt}(\text{TALLY})$$

so that when considering the number of queries made to tally oracles, there are only two classes, P/poly and P_m(TALLY).

Since for any tally set T, $P_{tt}(T) = P_T(T)$, neither an individual tally set nor the union of the reduction classes of all tally sets can be used to distinguish between polynomial time truth-table reducibility and polynomial time Turing reducibility. But the situation is different when the classes of sets interreducible to tally sets under these reducibilities are considered: we show in Theorem 3.5 that the resulting classes are not the same, more specifically, that $E_{tt}^P(\text{TALLY}) \neq E_T^P(\text{TALLY})$; thus, there are sets with self-producible circuits that are not truth-table equivalent in polynomial time to tally sets. Hence, it is possible to hide information about a tally set in a non-tally set in such a way that it is retrievable by Turing reductions to that non-tally set but it is no retrievable by truth-table reductions to the same set.

In addition, we consider strong nondeterministic polynomial-time Turing reducibility as defined by Long [11] and we show in Theorem 3.3 that

E_T^{SN} (TALLY) $\neq E_T^P$ (TALLY). Thus, we have the proper inclusions

$$E_{tt}^P(\text{TALLY}) \subset E_T^P(\text{TALLY}) \subset E_T^{SN}(\text{TALLY}).$$

Whether the corresponding inclusions

$$E_{tt}^P(\text{SPARSE}) \subseteq E_T^P(\text{SPARSE}) \subseteq E_T^{SN}(\text{SPARSE})$$

are proper is an open question.

Based on the fact that for every k, $P_{k-tt}(\text{SPARSE}) \neq P_{(k+1)-tt}(\text{SPARSE})$, we observe that $E_{k-tt}^P(\text{SPARSE}) \neq E_{(k+1)-tt}^P(\text{SPARSE})$. But the analogous situation with TALLY is unresolved. It can be shown that for each $k>0$, there exists a tally set T such that $E_{k-tt}^P(T) \neq E_{(k+1)-tt}^P(T)$. But for each k, the question $E_{k-tt}^P(\text{TALLY}) = ?\, E_{(k+1)-tt}^P(\text{TALLY})$ is open, as are the questions $E_m^P(\text{TALLY}) = ?\, E_{1-tt}^P(\text{TALLY})$ and $E_m^P(\text{TALLY}) = ?\, E_{btt}^P(\text{TALLY})$. Allender and Watanabe [2] have shown each of these three questions to be equivalent to the question of whether every honest polynomial-time computable function $f: \Sigma^* \to \{0\}^*$ is weakly invertible. Thus, there is reason to believe that these questions will be difficult to resolve.

2. PRELIMINARIES

In this section we review some definitions and establish notation additional to that given in the Introduction.

Throughout this paper we will consider the alphabet $\Sigma = \{0, 1\}$. The length of a string x will be denoted by $|x|$. The cardinality of a set S will be denoted by $\|S\|$. For a set S and an integer n, $S^n = \{x \in S \mid |x| = n\}$ and $S^{\leqq n} = \{x \in S \mid |x| \leqq n\}$. For a set S, χ_S denotes the characteristic function of S, and if $S \subseteq \Sigma^*$, then $\bar{S} = \Sigma^* - S$.

A set S is *sparse* if there is a polynomial q such that for all n, $\|S^{\leqq n}\| \leqq q(n)$. Let SPARSE denote the class of all sparse sets. A *tally* set is any subset of $\{0\}^*$. Let TALLY denote the class of all tally sets.

We will assume the existence of a pairing function $\langle\ ,\ \rangle : \Sigma^* \times \Sigma^* \to \Sigma^*$ with the properties that function and its inverses are computable in polynomial time and that when restricted to $\{0\}^* \times \{0\}^*$ yields a string in $\{0\}^*$.

For an oracle machine M, $L(M, A)$ denotes the set of strings accepted by M relative to oracle set A, and $L(M)$ denotes the set of strings accepted by M when no oracle queries are allowed by M.

We assume that the reader is familiar with the well-studied polynomial-time computable reducibilities referred to in the Introduction, with the corresponding reducibilities that are computed nondeterministically (*i.e.*, $\leqq_T^{NP}$, etc.), with the complexity classes P, NP, and PSPACE and their relativizations, and with the basic properties of these notions. An appropriate (and sufficiently comprehensive) reference for these topics is the book by Balcázar, Díaz, and Gabarró [5].

Long [11] studied the notion of "strong" nondeterministic polynomial-time reducibilities; $\leqq_T^{SN}$ denotes strong nondeterministic Turing reducibility. For our purposes it is sufficient to use the following characterization of $\leqq_T^{SN}$: $A \leqq_T^{SN} B$ if $A \leqq_T^{NP} B$ and $\bar{A} \leqq_T^{NP} B$. (*See* [5].) Consistent with the notation established in Section 1, we let

$$E_T^{NP}(\text{SPARSE}) = \bigcup_{S \in \text{SPARSE}} \{A \mid A \leqq_T^{NP} S \text{ and } S \leqq_T^{NP} A\},\ E_T^{NP}(\text{TALLY})$$
$$= \bigcup_{T \in \text{TALLY}} \{A \mid A \leqq_T^{NP} T \text{ and } T \leqq_T^{NP} A\},\ E_T^{SN}(\text{TALLY})$$
$$= \bigcup_{T \in \text{TALLY}} \{A \mid A \leqq_T^{SN} T \text{ and } T \leqq_T^{SN} A\}, \text{ and } E_T^{SN}(\text{SPARSE})$$
$$= \bigcup_{S \in \text{SPARSE}} \{A \mid A \leqq_T^{SN} S \text{ and } S \leqq_T^{SN} A\}.$$

The reader should be cautioned. It is known (*see* [3] or [5]) that the reducibility $\leqq_T^{NP}$ is not transitive so that "$A \leqq_T^{NP} S$ and $S \leqq_T^{NP} A$" is not an equivalence relation. Thus, for a set A, $E_T^{NP}(A)$ is *not* what might be called the "nondeterministic polynomial-time Turing degree of A". Similarly, it is known (*see* [10] or [5]) that for each $k > 0, \leqq_{k-tt}^P$ is not transitive, but that $\leqq_{tt}^P$ and $\leqq_T^P$ are transitive and so "$A \leqq_{tt}^P S$ and $S \leqq_{tt}^P A$" and "$A \leqq_T^P S$ and $S \leqq_T^P A$" are equivalence relations. In addition, it is known (*see* [11] or [5]) that $\leqq_T^{SN}$ is transitive.

3. MAIN RESULTS

In this section we establish the main results of this paper. There are various relationships between the various classes defined from the class TALLY of all tally sets and from the class SPARSE of all sparse sets that follow naturally from the definitions. The principal new results are the inequalities $E_T^P(\text{TALLY}) \neq E_T^{SN}(\text{TALLY})$ and $E_{tt}^P(\text{TALLY}) \neq E_T^P(\text{TALLY})$, established in Theorems 3.3 and 3.5 respectively. Other results follow from relationships established elsewhere but are given here for the sake of completeness.

We begin by establishing an equality.

THEOREM 3.1: $E_T^{NP}(\text{SPARSE}) = E_T^{NP}(\text{TALLY})$.

Proof: Since TALLY $\subseteq$ SPARSE, it follows that

$$E_T^{NP}(\text{TALLY}) \subseteq E_T^{NP}(\text{SPARSE}).$$

To see the converse, recall that for every $S \in$ SPARSE there exists $T_S \in$ TALLY such that $S \leq_T^P T_S$ [7]; the proof of this fact given by Schöning (Theorem 4.6 of [13]) shows that $T_S \leq_T^{NP} S$ and the reduction of T_S to S can be implemented by a nondeterministic oracle machine in which acceptance corresponds to computations where all of the oracle queries yield the answer "yes"; hence, the reduction $T_S \leq_T^{NP} S$ is a conjunctive reduction. For any sparse set S, if $A \in E_T^{NP}(S)$, then $A \leq_T^{NP} T_S$ since $A \leq_T^{NP} S$ and $S \leq_T^P T_S$, the latter reduction being carried out deterministically; also, $T_S \leq_T^{NP} A$ since $T_S \leq_T^{NP} S$ by means of a conjunctive reduction and $S \leq_T^{NP} A$ (recall that $\leq_T^{NP}$ is not transitive so the use of some modification of $\leq_T^{NP}$ such as being conjunctive is necessary). Thus, for any sparse set S, if $A \in E_T^{NP}(S)$, then $A \in E_T^{NP}(T_S)$. Hence, $E_T^{NP}(\text{SPARSE}) \subseteq E_T^{NP}(\text{TALLY})$. ■

Consider strong nondeterministic polynomial time reducibilities studied by Long [11].

THEOREM 3.2:

$$E_T^{SN}(\text{TALLY}) \neq E_T^{NP}(\text{TALLY}), \qquad E_T^{SN}(\text{TALLY}) \neq E_T^{SN}(\text{SPARSE}),$$

and for each reducibility

$$R \in \{\leq_m, \leq_{k-tt}, \leq_{btt}, \leq_{tt}, \leq_T\}, \qquad E_R^P(\text{TALLY}) \neq E_R^P(\text{SPARSE}).$$

Proof: Since $E_T^{NP}(\text{TALLY}) = E_T^{NP}(\text{SPARSE})$ by Theorem 3.1, we have

$$\text{SPARSE} \subseteq E_R^P(\text{SPARSE}) \subseteq E_T^{NP}(\text{TALLY}).$$

Long [12] has shown that there exists $S \in$ SPARSE such that for all $T \in$ TALLY, $S \notin E_T^{SN}$. Thus, SPARSE is not included in $E_T^{SN}(\text{TALLY})$ so SPARSE is not included in

$$E_R^P(\text{TALLY}) \qquad \text{and} \qquad E_R^P(\text{TALLY}) \neq E_R^P(\text{SPARSE})$$

for any R. It follows that $E_T^{SN}(\text{TALLY}) \neq E_T^{SN}(\text{TALLY})$ since

$$\text{SPARSE} \subseteq E_T^{NP}(\text{SPARSE}) = E_T^{NP}(\text{TALLY}). \quad ■$$

The problem E_T^{SN}(SPARSE)$=?\,E_T^{NP}$(SPARSE) remains open.

Consider the class E_T^P(TALLY). The inequality

$$E_T^P(\text{TALLY}) \neq E_T^P(\text{SPARSE})$$

is established in Theorem 3.2 above. We show that E_T^P(TALLY) is properly included in E_T^{SN}(TALLY); the question of whether E_T^P(SPARSE) is properly included in E_T^{SN}(SPARSE) remains open.

THEOREM 3.3: $E_T^P(\text{TALLY}) \neq E_T^{SN}(\text{TALLY})$.

Proof: A set $S \subseteq \{0, 1\}^*$ will be called *special* if for all $n \geqq 1$, S has exactly one element of length n. Thus, every special set is sparse. For any special set S, let $T(S) = \{\langle 0^n, 0^i \rangle \mid 1 \leqq i \leqq n$ and the i-th bit of the unique element of length n in S is $0\}$; recall that we assume that the pairing function $\langle\ ,\ \rangle$ used here satisfies the condition that $x, y \in \{0\}^*$ implies $\langle x, y \rangle \in \{0\}^*$, so that $T(S)$ is a tally set.

It is clear that $S \leqq_T^P T(S)$ so that $S \leqq_T^{SN} T(S)$. Notice that $T(S) \leqq_T^{NP} S$ since on input a string $\langle 0^n, 0^i \rangle$ a nondeterministic machine can determine whether $1 \leqq i \leqq n$, can guess a string y of length n whose i-th bit is 0, and determine (by querying the oracle) whether $y \in S$; if all conditions are satisfied, then the machine can accept $\langle 0^n, 0^i \rangle$. Similarly, notice that $\overline{T(S)} \leqq_T^{NP} S$ since on input a string $\langle 0^n, 0^i \rangle$ a nondeterministic machine can determine whether $1 \leqq i \leqq n$, can guess a string y of length n whose i-th bit is 1, and determine (by querying the oracle) whether $y \in S$; if all conditions are satisfied, then the machine can accept $\langle 0^n, 0^i \rangle$. Thus $T(S) \leqq_T^{SN} S$. Hence, $S \leqq_T^{SN} T(S)$ and $T(S) \leqq_T^{SN} S$ so that $S \in E_T^{SN}(T(S)) \subseteq E_T^{SN}$(TALLY).

We claim that there is a special set that is not in E_T^P(TALLY) so that $E_T^P(\text{TALLY}) \neq E_T^{SN}(\text{TALLY})$.

2

Let S be a special set with the property that for each n of the form

2

the unique string s_n of length n in S is random in the sense that s_n has Kolmogorov complexity greater than $n/2$; let $s_n = 0^n$ for other n. The fact that each string s_n has Kolmogorov complexity greater than $n/2$ may be interpreted as saying that S is not self-reductible.

Suppose $S \in E_T^P$(TALLY). Let T be a tally set such that $S \leqq_T^P$T and $T \leqq_T^P S$. Let M be a deterministic oracle machine that witnesses $T \leqq_T^P S$, and let $q(n)$ be a polynomial that bounds M's running time. For each n, every tally string in T of length at most $q(n)$ can be recognized by M relative to S, and each such string can also be recognized by M relative to $S - \{s_n\}$ (since

s_n has Kolmogorov complexity greater than $n/2$, no machine can generate s_n from a string with only $O(\log n)$ bits so in M's computation on a tally string of length at most $q(n)$, s_n is never generated as a query string). Since every tally string in T of length at most $q(n)$ can be recognized by M relative to $S-\{s_n\}$, the fact that S the strings in S whose length is of the form $2^{2^{\cdot^{\cdot^{2}}}}$ are random allows us to conclude that every tally string in T of length at most $q(n)$ can be generated in polynomial time by using $c+O(\log n)$ bits, where c is a constant (that accounts for the short strings in S).

Recall that $S \leqq_T^P T$. Since $S \leqq_T^P T$ and $T \leqq_T^P S$, the argument in the last paragraph shows that S is self-reducible, contradicting the choice of S. (An alternative view is that for each n of the form $2^{2^{\cdot^{\cdot^{2}}}}$, a machine generate in turn each string of length n and reduce each to T, thus discovering which of these strings is s_n. Thus, s_n can be determined by using $O(\log n)$ bits, contradicting the choice of s_n as a string with Kolmogorov complexity greater than $n/2$.) ■

Recall that

$$P_T(\text{TALLY}) = P_{tt}(\text{TALLY}) = P_{tt}(\text{SPARSE}) = P_T(\text{SPARSE}) = P/\text{poly}.$$

We know that

$$E_T^P(\text{TALLY}) \neq E_T^P(\text{SPARSE}) \quad \text{and} \quad E_{tt}^P(\text{TALLY}) \neq E_{tt}^P(\text{SPARSE}).$$

In Theorem 3.5 below we separate $E_{tt}^P(\text{TALLY})$ from $E_T^P(\text{TALLY})$ even though $P_{tt}(\text{TALLY}) = P_T(\text{TALLY})$. The problem

$$E_{tt}^P(\text{SPARSE}) = ?\, E_T^P(\text{SPARSE})$$

remains open.

LEMMA 3.4: *If* $A \in E_{tt}^P(\text{TALLY})$, *then* $P_T(A) = P_{tt}(A)$.

Proof: This follows immediately from the fact that for any tally set T, $P_T(T) = P_{tt}(T)$. ■

THEOREM 3.5: $E_{tt}^P(\text{TALLY}) \neq E_T^P(\text{TALLY})$.

Proof: For any tally set $T \subseteq \{0\}^*$, define $A(T) = \{x_1 1 x_2 1 \ldots 1 x_t \mid$ each $x_i \in \{0\}^*$, $t \geqq 1$, $|x_1| < |x_2| < \ldots < |x_t|$, and there exists n such that $T^{\leqq n} = \{x_1, x_2, \ldots, x_t\}\}$. It is clear that $A(T) \leqq_T^P T$. Since $A(T)$ is a set that is linearly ordered, on input 0^n a machine computing in polynomial time relative to $A(T)$ can obtain the longest $w \in A(T)$ such that $|w| \leqq n^2$. From this w it is easy to determine whether 0^n is in T. Hence, $T \leqq_T^P A(T)$. Thus, for every tally set T, $A(T) \in E_T^P(T) \subseteq E_T^P(\text{TALLY})$. Thus, to prove that $E_{tt}^P(\text{TALLY}) \neq E_T^P(\text{TALLY})$, it is sufficient by Lemma 3.4 to show the existence of a tally set X such that $X \nleqq_{tt}^P A(X)$.

We use the following notation: for any $w \in \{0, 1\}^*$, if $w = x_1 1 x_2 1 \ldots 1 x_t$, $t \geqq 1$, each $x_i \in \{0\}^*$, then tally-set $(w) = \{x_1, \ldots, x_t\}$.

[Recall that a set B is truth-table reducible to set C, written $B \leqq_{tt}^P C$, if there exist polynomial time computable functions f (the generator) and g (the evaluator) such that for all x, $f(x)$ is a list of strings, $g(x)$ is a Boolean circuit with the number of input variables being equal to the number of strings in the list $f(x)$, and $x \in B$ if and only if $g(x)$ evaluates to true on $\langle \chi_C(y_1), \ldots, \chi_C(y_k) \rangle$ where $f(x) = \langle y_1, \ldots, y_k \rangle$.]

The existence of a tally set X such that $X \nleqq_{tt}^P A(X)$ can be shown by diagonalization. Since each function f and each function g making up reduction $\langle f, g \rangle$ is polynomial time computable, we can assume an enumeration of $\leqq_{tt}^P$-reductions: $\{\langle f_i, g_i \rangle \mid i \leqq 1\}$ where for every i, there is a machine that computes both f_i and g_i and has running time at most $p_i(n) = n^i + i$.

The set X can be constructed by stages and only a sketch is presented.

Stage 0: $X_0 := \varnothing$ and $n_0 := 1$.

Stage $i > 0$: Choose m so that $m > p_{i-1}(n_{i-1})$ and $2^m > p_i(2m) \geqq t$. Let $n_i := 2m$. At the beginning of this stage X_{i-1} is a tally set containing no string longer than n_{i-1}.

If $f_i(0^{2m}) = \langle y_1, y_2, \ldots, y_t \rangle$, then choose $S \subseteq \{0^m, 0^{m+1}, \ldots, 0^{2m-1}\}$ so that for every y_j, $X_{i-1} \cup S \neq (\text{tally-set}(y_j) - \{0^{2m}\})$ [the choice of m such that $2^m > p_i(2m) \geqq t$ guarantees that such an S exists]. For such a set S, observe that for every y_j, $y_j \notin A(X_{i-1} \cup S)$ and $y_j \notin A(X_{i-1} \cup S \cup \{0^{2m}\})$. Hence,

$$g_i(0^{2m}, \chi_{A(X_{i-1} \cup S)}(f_i(0^{2m}))) = g_i(0^{2m}, \chi_{A(X_{i-1} \cup S \cup \{0^{2m}\})}(f_i(0^{2m}))).$$

Finally, if

$$g_i(0^{2m}, \chi_{A(X_{i-1} \cup S)}(f_i(0^{2m}))) = 1,$$

then let $X_i := X_{i-1} \cup S$; otherwise, let $X_i := X_{i-1} \cup S \cup \{0^{2m}\}$.

This completes Stage i.

Let $X := \bigcup_{i \geqq 0} X_i$.

From the construction, $0^{n_i} \in X$ if and only if $g_i(0^{n_i}, \chi_{A(X)}(f_i(0^{n_i}))) = 0$. Thus, for every i, $\langle f_i, g_i \rangle$ does not witness $X \leqq_{tt}^P A(X)$. Hence, $X \nleqq_{tt}^P A(X)$ as desired. ■

To obtain other separations from known results, we use the following simple fact.

LEMMA 3.6: *Let* $\mathbf{C}$ *be a class of sets and let* α *and* β *be two different reducibilities in* $\{\leqq_T, \leqq_{tt}, \leqq_{btt}, \leqq_{k-tt}$ *for each* $k>0$, $\leqq m\}$. *If* $P_\alpha(\mathbf{C}) \neq P_\beta(\mathbf{C})$, *then* $E_\alpha^P(\mathbf{C}) \neq E_\beta^P(\mathbf{C})$.

Proof: For any sets $A, B \subseteq \Sigma^*$, let $A \oplus B$ denote $\{0x, 1y \mid x \in A, y \in B\}$.

Suppose that there exists $A \in P_\alpha(\mathbf{C}) - P_\beta(\mathbf{C})$. Since $A \in P_\alpha(\mathbf{C})$, there exists $C_0 \in \mathbf{C}$ such that $A \leqq_\alpha^P C_0$. Thus, $A \oplus C_0 \leqq_\alpha^P C_0$ and $C_0 \leqq_\alpha^P A \oplus C_0$, so that $A \oplus C_0 \equiv_\alpha^P C_0$ and $A \oplus C_0 \in E_\alpha^P(\mathbf{C})$. Since $A \notin P_\beta(\mathbf{C})$, for every $C \in \mathbf{C}$, $A \nleqq_\beta^P C$. If there exists $C_1 \in \mathbf{C}$ such that $A \oplus C_0 \equiv_\beta^P C_1$, then $A \oplus C_0 \leqq_\beta^P C_1$ and, hence, $A \leqq_\beta^P C_1$, contradicting the fact that $A \notin P_\beta(\mathbf{C})$. Thus, $A \oplus C_0 \notin E_\beta^P(\mathbf{C})$. ■

Notice that $E_{btt}^P(\text{TALLY}) \neq E_{btt}^P(\text{SPARSE})$ by Theorem 3.2. Furthermore, $E_{btt}^P(\text{TALLY}) \neq E_{tt}^P(\text{TALLY})$ and $E_{btt}^P(\text{SPARSE}) \neq E_{tt}^P(\text{SPARSE})$ as in Theorem 3.7 (*a*) and (*b*) below.

THEOREM 3.7:

(*a*) $E_{btt}^P(\text{TALLY}) \neq E_{tt}^P(\text{TALLY})$.

(*b*) $E_{btt}^P(\text{SPARSE}) \neq E_{tt}^P(\text{SPARSE})$.

(*c*) For every k, $E_{k-tt}^P(\text{SPARSE}) \neq E_{(k+1)-tt}^P(\text{SPARSE})$.

(*d*) $E_m^P(\text{SPARSE}) \neq E_{1-tt}^P(\text{SPARSE})$.

(*e*) $E_{ctt}^P(\text{SPARSE}) \neq E_{tt}^P(\text{SPARSE})$.

(*f*) $E_{ctt}^P(\text{SPARSE}) \neq E_{dtt}^P(\text{SPARSE})$.

(*g*) $E_{ctt}^P(\text{SPARSE}) \neq E_{btt}^P(\text{SPARSE})$.

(*h*) $E_{dtt}^P(\text{SPARSE}) \neq E_{btt}^P(\text{SPARSE})$.

(*i*) $E_{ctt}^P(\text{TALLY}) \neq E_{tt}^P(\text{TALLY})$.

(*j*) $E_{ctt}^P(\text{TALLY}) \neq E_{dtt}^P(\text{TALLY})$.

(*k*) $E_{ctt}^P(\text{TALLY}) \neq E_{btt}^P(\text{TALLY})$.

(*l*) $E_{dtt}^P(\text{TALLY}) \neq E_{tt}^P(\text{TALLY})$.

(*m*) $E_{dtt}^P(\text{TALLY}) \neq E_{btt}^P(\text{TALLY})$.

The proof of Theorem 3.7 follows easily from Lemma 3.6 and by results of Book and Ko [5] and of Ko [9] on the separation of reduction classes.

The most interesting open questions are those stated in Section 1; they have been studied by Allender and Watanabe [2]. Other questions remain open; for example, the question of whether E_T^P(SPARSE) is properly included in P/poly.

Note added in proof: There are some recent results concerning the open problems. These results are reported in the following:

R. GAVALDA and O. WATANABE, Computational Complexity of Small Descriptions, *Proc. 6th I.E.E.E. Conference on Structure in Complexity Theory*, July 1991, Chicago, IL. (to appear).

E. ALLENDER, L. HEMACHANDRA, M. OGIWARA and O. WATANABE, Relating Equivalence and Reducibility to Sparse Sets, *Proc. 6th I.E.E.E. Conference on Structure in Complexity Theory*, July 1991, Chicago, IL. (to appear).

REFERENCES

1. E. ALLENDER and R. RUBINSTEIN, P-printable sets, *S.I.A.M. J. Comput.*, 1988, *17*, pp. 1193-1202.
2. E. ALLENDER and O. WATANABE, Kolmogorov complexity and degrees of tally sets, *Inform. and Comput.*, 1990, *86*, pp. 160-178.
3. T. BAKER, J. GILL and R. SOLOWAY, Relativizations of the $P=?$ NP question, *S.I.A.M. J. Comput.*, 1975, *4*, pp. 431-442.
4. J. BALCÁZAR and R. BOOK, Sets with small generalized Kolmogorov complexity, *Acta Inform.*, 1986, *23*, pp. 679-688.
5. J. BALCÁZAR, J. DÍAZ and J. GABARRÓ, *Structural Complexity, I* and *II*, Springer-Verlag, 1988 and 1990.
6. R. BOOK and K. KO, On sets truth-table reducible to sparse sets, *S.I.A.M. J. Comput.*, 1988, *17*, pp. 903-919.
7. J. HARTMANIS, N. IMMERMAN and V. SEWELSON, Sparse sets in *NP-P*: EXPTIME versus NEXPTIME, *Proc. 15th A.C.M. Symp. Theory of Computing*, 1983, pp. 382-391.
8. K. KO, Continuous optimization problems and a polynomial hierarchy of real functions, *J. Complexity*, 1985, *1*, pp. 210-231.
9. K. KO, Distinguishing bounded reducibilities by sparse sets, *Inform. and Comput.*, 1989, *81*, pp. 62-87.
10. R. LANDER, N. LYNCH and A. SELMAN, A comparison of polynomial-time reducibilities, *Theoret. Comput. Sci.*, 1975, *1*, pp. 103-123.
11. T. LONG, Strong nondeterministic polynomial-time reducibilites, *Theoret. Comput. Sci.*, 1982, *21*, pp. 1-25.
12. T. LONG, On restricting the size of oracles compared with restricting access to oracles, *S.I.A.M. J. Comput.*, 1985, *14*, pp. 585-597.
13. U. SCHÖNING, *Complexity and Structure*, L.N.C.S., No. 211, 1986, Springer-Verlag Publ. Co.