SIMPLICITY, RELATIVIZATIONS AND NONDETERMINISM*

JOSÉ L. BALCÁZAR†

Abstract. Relativizations of complexity classes in which simple sets exist are considered. A recursive oracle is constructed relative to which a simple set exists for NP. Some other general theorems are proven, showing that the time bounds are not a crucial hypothesis; bounds on the way in which the oracle is accessible, namely, the number of queries and/or the number of nondeterministic steps, are shown to be the fundamental hypothesis. As a result, simple sets are shown to exist in many different relativized complexity classes.

Key words. complexity classes, relativizations, nondeterminism, bounded queries, immunity, simplicity, NP

Introduction. The relationship between deterministic and nondeterministic models of computation has been investigated for many years. The central problems appear to be fundamentally difficult. The open problem that has dominated recent work is the question of the deterministic and nondeterministic models restricted to polynomial running times, that is, the " $P = ?NP$ " problem.

A simple analogy may be drawn between the class P and the class of recursive sets on the one hand, and the class NP and the class of recursively enumerable sets on the other hand: the class NP can be defined by applying polynomially bounded existential quantifiers to predicates in P. Such an analogy suggests reasons for translating the definitions and, when possible, the results of elementary recursive function theory to the setting of polynomial time-bounded computation. As examples of these "translations," recall the Hartmanis-Berman conjecture that all of the NP-complete sets are polynomially isomorphic, and the polynomial hierarchy specified by alternation of polynomially bounded quantifiers on predicates in P.

However, even elementary propositions of recursive function theory become difficult in the setting of polynomial time bounds; in fact, some are unsolved problems. Recently, two such notions have been investigated, the notion of "immune" set and the notion of "simple" set. Since the question $P = ?NP$ is open, it is not surprising that the existence of a "P-immune" set in NP or of a "NP-simple" set in NP is not known. This is the subject of the present paper.

The proof of Baker, Gill, and Solovay [1] of the existence of a set A such that $P(A) \neq NP(A)$ sets the stage for numerous investigations of the properties of relativizations of P and NP. Other such separating theorems have been developed and there are two specific studies that are important for the present work. First, Kintala [6], [7] considered relativizations of machines that run in polynomial time but have restrictions on the number of nondeterministic steps in any computation. Thus, there is a recursive set A such that for every integer k , the class of sets recognized relative to A by polynomial time-bounded oracle machines with at most n^k nondeterministic steps in any computation is properly included in the corresponding class specified by machines that may use n^{k+1} nondeterministic steps. Second, Xu, Doner, and Book [11] observed that the separating theorems proved by methods similar to Baker, Gill, and Solovay

* Received by the editors July 5, 1983, and in revised form April 17, 1984. The research reported in this paper was performed while the author visited the Department of Mathematics, University of California at Santa Barbara. This work was supported in part by a grant from the USA-Spanish Joint Committee for Educational and Cultural Affairs, and by the National Science Foundation under grants MCS80-11979 and MCS83-12472.

† Facultat d'Informàtica, Universitat Politècnica de Barcelona, Jordi Girona Salgado, 31, Barcelona, 34, Spain.

do not really depend on time as a bound, but rather the number of oracle queries allowed in a computation and the number of nondeterministic steps in a computation combine to yield these results. Thus, they established a very general separating theorem for relativizations of classes specified by machines with restricted nondeterminism.

If $\mathcal{C}$ is a class of sets, then a set L_1 is $\mathcal{C}$ -immune if L_1 is infinite and no infinite subset of L_1 is in $\mathcal{C}$, and a set L_2 is $\mathcal{C}$ -simple if L_2 has an infinite complement, L_2 is in $\mathcal{C}$, and the complement of L_2 has no infinite subset in $\mathcal{C}$.

Homer and Maass [5] showed that there is a recursive set A such that there is a set in $NP(A)$ that is $P(A)$ -immune. Also, using priority methods, they showed that there is a recursively enumerable set B such that $NP(B)$ contains a set that is $NP(B)$ -simple. Schöning and Book [9] used a simple diagonalization in a different proof of the first result and they extended the argument to a wide variety of other classes by focusing not on time but rather on the number of oracle queries allowed in any computation and on the amount of nondeterminism allowed in any computation. Thus, Schöning and Book established two very general "immunity theorems" that establish "strong separation" of relativized classes, separations witnessed by the appropriate immune sets.

In this paper we establish a number of results about simple sets. The first result, Theorem 1, strengthens the result of Homer and Maass mentioned above: there is a recursive set A such that $NP(A)$ has a set that is $NP(A)$ -simple. The proof is by means of a straightforward diagonalization (a "slow" diagonalization in terms of [4]) and can be "lifted" to other circumstances. Thus, Theorem 3 and Theorem 5 provide very general results on the existence of simple sets that parallel the results of Schöning and Book. A number of applications are given.

These results add very strong evidence to the argument that the study of determinism vs. nondeterminism by means of relativizations has not illuminated the basic difficulties but instead has illustrated the power of nondeterminism in steps that write on the query tape and so generate a very large set of strings to be queried. This point is made stronger when one notes that Theorem 5 is established in the setting of an infinite hierarchy of functions that bound the amount of nondeterminism allowed in computations.

1. Preliminaries. Throughout this paper, we consider decision problems encoded as subsets of Γ^* where $\Gamma = \{0, 1\}$. For a word w , $|w|$ denotes the length of w . We assume some fixed ordering $\leq$ of Γ^* such that $|x| < |y|$ implies $x < y$.

The computational model considered here is the multitape oracle Turing machine, deterministic or nondeterministic. For relativized computation oracle machines are assumed to have a distinguished work tape, the *query* tape, and three distinguished states, QUERY, YES, and NO. If some computation of such a machine enters the state QUERY, then at the next step the machine transfers into the state YES if the string currently appearing on the query tape is in a fixed *oracle set*; otherwise, the machine transfers into the state NO; in either case the query tape is instantly erased. For such a machine M and oracle set A , the set of strings *accepted* by M relative to the oracle set A is $L(M, A) = \{w \mid \text{there is an accepting computation of } M \text{ on input } w \text{ relative to oracle set } A\}$.

Oracle machines are defined in the standard way and may be bounded with respect to time or space by appropriate bounding functions. Time (space) bounds are assumed to be running times so that a "clock" may be added to any such machine. Querying the oracle costs just one step in time and the length of the query tape is bounded by whatever space bound is imposed.

Let f be a running time and let T be a class of running times. We denote by $\text{NTIME}(f, A)$ the class of sets accepted by nondeterministic machines with running time f relative to oracle set A . Similarly, $\text{DTIME}(f, A)$ denotes the corresponding class specified by deterministic machines. Also, $\text{NTIME}(T, A) = \bigcup_{f \in T} \text{NTIME}(f, A)$ and $\text{DTIME}(T, A) = \bigcup_{f \in T} \text{DTIME}(f, A)$. Refinements of these classes will be introduced in § 4.

When considering machines that are nondeterministic, the expression "always halts" means that relative to every oracle set, every computation on every input must halt. For example, this is the case for time-clocked machines specifying classes such as $\text{DTIME}(f, A)$ or $\text{NTIME}(f, A)$.

Let $\mathcal{C}$ be a class of subsets of Γ^* . Denote by $\text{co-}\mathcal{C}$ the class $\{\Gamma^* - L \mid L \in \mathcal{C}\}$. A set L is $\mathcal{C}$ -immune if L is infinite and no infinite subset of L is in $\mathcal{C}$. A set L is $\mathcal{C}$ -simple if L is in $\mathcal{C}$ and $\Gamma^* - L$ is $\mathcal{C}$ -immune. In what follows, for any set L , the set $\Gamma^* - L$ will be denoted $\bar{L}$.

2. A simple set for NP relativized. The first result is the existence of a recursive set A such that $\text{NP}(A) = \{L(M, A) \mid M \text{ is a nondeterministic polynomial time-bounded oracle machine}\}$ has a set that is $\text{NP}(A)$ -simple. The existence of a $\mathcal{C}$ -simple set for any class $\mathcal{C}$ shows a strong separation between $\mathcal{C}$ and $\text{co-}\mathcal{C}$. The immunity results in [9] may be viewed as a strong separation between classes specified by deterministic machine vs. nondeterministic machines. Similarly, the existence of simple sets implies a strong separation between a class $\mathcal{C}$ and the corresponding class $\text{co-}\mathcal{C}$, where $\mathcal{C}$ is specified by nondeterministic machines; this separation is witnessed by a set in $\text{co-}\mathcal{C}$ which is not "infinitely approximable" within $\mathcal{C}$.

THEOREM 1. *There is a recursive set A such that $\text{NP}(A)$ contains a simple set.*

Proof. The basic construction diagonalizes over an enumeration of the clocked nondeterministic polynomial time-bounded oracle machines so that for any fixed oracle set A , each set in $\text{NP}(A)$ is presented infinitely often. Let $\text{NP}_1, \text{NP}_2, \dots$ be an enumeration of such machines; for each i , let q_i be a nondecreasing polynomial bounding NP_i 's running time.

For any set $A \subseteq \Gamma^*$, let $L(A) = \{w \mid w \notin \{0\}^*, \text{ or } w = 0^m \text{ and some word in } A \text{ has length of } m\}$. Clearly, $L(A) \in \text{NP}(A)$. The construction of A is based on a diagonalization over $\text{NP}(A)$ such that $L(A)$ is $\text{NP}(A)$ -simple. The set A is constructed in stages so that at each stage n , the intersection of $L(A)$ with $L(\text{NP}_j, A)$ for each $j \leq n$ is forced, when possible, to be nonempty.

Construct A by performing in the natural order $0, 1, 2, \dots$ the stages as follows:

Stage 0

$$A_0 := \{0\}^*;$$

$$m_0 := 0;$$

$$R_0 := \emptyset;$$

end stage;

Stage n ($n \geq 1$)

$$R_n := R_{n-1} \cup \{n\};$$

$$m_n := \min \{m \mid \max \{q_j(m_{n-1}) \mid j < n\} < m, \text{ and } \max \{q_j(m) \mid j \leq n\} < 2^m\};$$

$$A_n := A_{n-1} - \{0^{m_n}\};$$

if there exists $j \in R_n$ such that $0^{m_n} \in L(\text{NP}_j, A_n)$

then

let j_n be the least such j ;

choose any accepting computation of $L(\text{NP}_{j_n}, A_n)$ on input 0^{m_n} ;

let w_n be the least word of length m_n not queried in the chosen computation;

$$A_n := A_n \cup \{w_n\};$$

$$R_n := R_n - \{j_n\};$$

end if;

end stage.

The set A is defined as $A := \{x \in \Gamma^* \mid x \in A_n \text{ for almost every } n\}$.

The conditions imposed on m_n guarantee that adding or deleting words of length m_n does not alter the previous computations. In any single computation of a machine NP_i on an input x , at most $q_i(|x|)$ words can be queried; since there are 2^{m_n} words of length m_n , there is a word w_n available if it is needed. Thus, the construction can be performed.

It is clear that the set A is recursive. We show that $\bar{L(A)} = \Gamma^* - L(A)$ is infinite. By the definition of $L(A)$, $L(A)$ is finite if and only if $L(A)$ contains all but finitely many words of the form 0^{m_n} . This implies that words of length m_n are added to A in all but finitely many stages n ; hence, the "then" case occurred at all but finitely many stages, and by the construction we see that the set $R = \bigcup_{n \geq 0} R_n$ is finite; indeed, one number is added and one deleted at each such stage. But no index of the empty set can be deleted from R at any stage and every index is added to R at its own stage; since there are infinitely many indices of the empty set, R is infinite. Thus, $\bar{L(A)}$ is infinite as claimed.

Now suppose that for some j , $L(\text{NP}_j, A) \subseteq \bar{L(A)}$ and $L(\text{NP}_j, A)$ is infinite. Since $L(A) \subseteq \{0^{m_n} \mid n \geq 0\}$, this means $L(\text{NP}_j, A) \subseteq \{0^{m_n} \mid n \geq 0\}$ so for infinitely many n , $0^{m_n} \in L(\text{NP}_j, A)$. Since only finitely many indices are less than j , there is some stage n such that j is the least index in R with $0^{m_n} \in L(\text{NP}_j, A)$. At this stage w_n is added to $A_n \subseteq A$ so that $0^{m_n} \in L(A)$, contradicting $L(\text{NP}_j, A) \subseteq \bar{L(A)}$. Thus, for any j , if $L(\text{NP}_j, A) \subseteq \bar{L(A)}$, then $L(\text{NP}_j, A)$ is finite.

Hence, $\bar{L(A)}$ is $\text{NP}(A)$ -immune and so $L(A)$ is $\text{NP}(A)$ -simple. $\square$

It is not difficult to combine this diagonalization with the one used by Schöning and Book [9] so that the resulting set A has the property that *simultaneously* $\text{NP}(A)$ has one set that is $\text{P}(A)$ -immune and another set that is $\text{NP}(A)$ -simple. Here we give only the construction. We assume an enumeration $P_1, P_2, \dots$ of the clocked deterministic polynomial time-bounded oracle machines. For each i , let q_i be a nondecreasing polynomial bounding both P_i 's running time and also NP_i 's running time.

For any set $A \subseteq \Gamma^*$, define $L_{\text{even}}(A) = \{0^m \mid \text{there exists } w \in A \text{ such that } |w| = 2m\}$ and $L_{\text{odd}}(A) = \{0^m \mid \text{there is no } w \in A \text{ such that } |w| = 2m + 1\}$. Clearly, $L_{\text{even}}(A) \in \text{NP}(A)$ and $L_{\text{odd}}(A) \in \text{co-NP}(A)$.

The construction diagonalizes over $\text{P}(A)$ at even stages and $\text{NP}(A)$ at odd stages.

Stage 0

$$A_0 := \{0^k \mid k \text{ is odd}\};$$

$$m_0 := 0;$$

$$R_0 := \emptyset;$$

$$S_0 := \emptyset;$$

end stage;

Stage $2n - 1$ ($n \geq 1$)

$$R_{2n-1} := R_{2n-2};$$

$$S_{2n-1} := S_{2n-2} \cup \{n\};$$

$$m_{2n-1} := \min \{m \mid \max \{q_j(m_{2n-2}) \mid j < n\} < 2m + 1, \text{ and } \max \{q_j(m) \mid j \leq n\} < 2^{2m+1}\};$$

$A_{2n-1} := A_{2n-2} - \{0^{2m_{2n-1}+1}\};$

if there is a $j \in S_{2n-1}$ such that $0^{m_{2n-1}} \in L(NP_j, A_{2n-1})$

then

let j_{2n-1} be the least such j ;

choose an accepting computation of $L(NP_{j_{2n-1}}, A_{2n-1})$ on $0^{m_{2n-1}}$;

let w_{2n-1} be the least word of length $2m_{2n-1}+1$ not queried in this computation;

$A_{2n-1} := A_{2n-1} \cup \{w_{2n-1}\};$

$S_{2n-1} := S_{2n-1} - \{j_{2n-1}\};$

end if;

end stage;

Stage $2n$ ($n \geq 1$)

$R_{2n} := R_{2n-1} \cup \{n\};$

$S_{2n} := S_{2n-1};$

$m_{2n} := \min \{m \mid \max \{q_j(m_{2n-1}) \mid j < n\} < 2m, \text{ and } \sum_{j \leq n} q_j(m) < 2^{2m}\};$

if there is a $j \in R_{2n}$ such that $0^{m_{2n}} \in L(P_j, A_{2n-1})$

then

let j_{2n} be the least such j ;

$R_{2n} := R_{2n} - \{j_{2n}\};$

$A_{2n} := A_{2n-1};$

else

let w_{2n} be the least word of length $2m_{2n}$ not queried in any computation of $L(P_j, A_{2n-1})$ on $0^{m_{2n}}$ for every $j \in R_{2n}$;

$A_{2n} := A_{2n-1} \cup \{w_{2n}\};$

end if;

end stage.

The set A is defined as $A := \{x \in \Gamma^* \mid x \in A_n \text{ for almost every } n\}$.

Arguments similar to those used for the first construction show that both $L_{\text{even}}(A)$ and $L_{\text{odd}}(A)$ are infinite, that $L_{\text{even}}(A)$ is $P(A)$ -immune, and that $L_{\text{odd}}(A)$ is $NP(A)$ -immune. Thus, we have the following result.

THEOREM 2. *There is a recursive set A such that $NP(A)$ has both a $P(A)$ -immune set and also an $NP(A)$ -simple set.*

It is not known whether there is a set A such that some set L in $NP(A)$ is simultaneously $P(A)$ -immune and $NP(A)$ -simple, that is, L is in $NP(A)$, L is infinite, no infinite subset of L is in $P(A)$, and no infinite subset of $\bar{L}$ is in $NP(A)$. We continue to investigate this problem.

3. Simple sets for other relativized classes. The proof technique used to establish Theorem 1 is applicable to a wide variety of complexity classes other than NP . Clearly no class closed under complementation admits a simple set. But our investigation is concerned with the necessity of the polynomial time bound. A careful analysis of the proof of Theorem 1 shows that the polynomials are not used as a bound on the running times but rather that running times bound the number of oracle queries in computations and also the number of nondeterministic steps in computations. This is by no means surprising when one considers the results in [2], [3], [9], [10], [11]. In this section we state two results whose proofs are based on the constructions in § 2.

Let f be a function on the natural numbers. For any set A , let $L_f(A)$ be defined as $L_f(A) := \{0^m \mid \text{for all } w \in A, |w| \neq f(m)\}$.

If f is a running time, then it is clear that for every set A , $\overline{L_f(A)} \in \text{NTIME}(f, A)$. By diagonalizing over a class of machines, it is possible to construct A so that $L_f(A)$

is immune with respect to this class. Thus, if $\overline{L_f(A)}$ is in the class, then it will be simple with respect to this class.

THEOREM 3. *Let $M = \{M_i \mid i \geq 1\}$ be an effective enumeration of a class of nondeterministic oracle machines that always halt, and let $T = \{t_i \mid i \geq 1\}$ be a class of running times. For every set B , let $L(M, T, B)$ denote the collection of sets $L(M, B)$ such that M is in M and for some t in T and all inputs w to M , some accepting computation of M on w makes at most $t(|w|)$ oracle queries. Suppose that (i) for every $f, g \in T$, $f(n) < 2^{g(n)}$ for all but finitely many n , and (ii) there is a finite set F such that for every set B , there are infinitely many i with $L(M_i, B) = F$. Then for any fixed $f \in T$, there is a recursive set A such that $L_f(A)$ is $L(M, T, A)$ -immune; hence if $\overline{L_f(A)} \in L(M, T, A)$, then $\overline{L_f(A)}$ is $L(M, T, A)$ -simple.*

Clearly, Theorem 1 is a corollary of Theorem 3. Before proving it, let us show how the theorem applies. All these corollaries follow easily from Theorem 3.

For every set A , let $\text{NEXT}(A)$ denote the collection of sets recognized relative to A by nondeterministic oracle machines that run in time 2^{in} for some $i > 0$.

COROLLARY 3.1. *There is a recursive set A such that $\text{NEXT}(A)$ has a set that is $\text{NEXT}(A)$ -simple.*

For each integer $i > 0$, define $\exp(2, 1, in) = 2^{in}$ and for integer $j > 0$, define $\exp(2, j+1, in) = 2^{\exp(2, j, in)}$. Fix an integer $h > 0$ and let $T = \{\exp(2, h, in) \mid i > 0\}$.

COROLLARY 3.2. *There is a recursive set A such that $\text{NTIME}(T, A)$ has a set that is $\text{NTIME}(T, A)$ -simple.*

For every set A , let $\text{NPQUERY}(A)$ ($\text{PQUERY}(A)$) be the collection of sets $L(M, A)$ where M is a nondeterministic (deterministic) oracle machine that uses polynomial work space and can make at most a polynomial number of oracle queries in any accepting computation. See [2] for interesting properties of these classes.

COROLLARY 3.3. *There is a recursive set A such that $\text{NPQUERY}(A)$ has a set that is $\text{NPQUERY}(A)$ -simple.*

If one considers the "bounded query" machines [2], [3] that specify classes of the form $\text{NPQUERY}(A)$ and allow bounds of the form $\exp(2, h, in)$, then one can apply Theorem 3 to obtain a result similar to Corollary 3.3.

Let us turn to the general theorem.

Proof of Theorem 3. Without loss of generality we assume that each machine M_i operates within the bound $t_i \in T$ on the number of queries. This can be achieved by constructing a new enumeration $M_{(i,j)}$ obtained by adding a "clock" t_j from T that stops machine M_i if it attempts to query the oracle more than the allowed number of times. Then an effective "renaming" of the enumeration of T allows us to assume that t_i bounds the number of queries of M_i .

Fix $f \in T$ and perform the construction as follows. Note that it is an easy adaptation of the proof of Theorem 1.

Stage 0

$A_0 := \{0\}^*$;

$m_0 := 0$;

$R_0 := \emptyset$;

end stage;

Stage n ($n \geq 1$)

$R_n := R_{n-1} \cup \{n\}$;

$m_n := \min \{m \mid \max \{t_j(m) \mid j \leq n\} < 2^{f(m)} \text{ and } m \text{ is greater than } |w| \text{ for any } w \text{ queried to the oracle in a computation chosen at earlier stages}\};$

$A_n := A_{n-1} - \{0^{f(m_n)}\};$

if there is a $j \in R_n$ such that $0^{m_n} \in L(M_j, A_n)$

then

let j_n be the least such j ;

fix an accepting computation that queries the oracle at most $t_{j_n}(m_n)$ times;

let w_n be the least word of length $f(m_n)$ that has not been queried in the fixed computation;

$A_n := A_n \cup \{w_n\}$;

$R_n := R_n - \{j_n\}$;

end if;

end stage.

As in Theorem 1, the conditions imposed in m_n guarantee that the construction can be performed, and that previously considered computations do not change. On the other hand, such a m_n exists, as follows from the hypothesis. All but finitely many indices of the finite set cited in the hypothesis must remain forever in R , so $L_f(A)$ is infinite. For each i , if $L(M_i, A) \subseteq \{0\}^*$ and $L(M_i, A)$ is infinite, then at some stage n , the index i is removed from R_n . Thus, $L_f(A)$ is $L(\mathbf{M}, \mathbf{T}, A)$ -immune. $\square$

This theorem parallels the first immunity theorem in [9], which asserts that under similar hypotheses immune sets exist in relativizations of complexity classes (possibly, those specified by nondeterministic machines). Under hypotheses strong enough to imply both the hypothesis of Theorem 3 and that of the first immunity theorem, the constructions can be merged in the same way as was done in § 2 to obtain Theorem 2. We omit the proof; it involves no new ideas.

THEOREM 4. Let $\mathbf{T}$ be a class of running times, and let $\mathbf{M}_1 = \{M_{1,i} \mid i \geq 1\}$ and $\mathbf{M}_2 = \{M_{2,i} \mid i \geq 1\}$ be effective enumerations of deterministic and, respectively, nondeterministic oracle Turing machines that always halt. Further, assume that

- (i) for every $f \in \mathbf{T}$, and every integer, there is a $g \in \mathbf{T}$ such that $c \cdot f(n) < g(n)$ for almost all n ;
- (ii) for every $f, g \in \mathbf{T}$, $f(n) < 2^{g(n)}$ for almost all n ;
- (iii) for every i , there is a $t \in \mathbf{T}$ that bounds the number of queries in the computations of $M_{1,i}$;
- (iv) for every i , there is a $t \in \mathbf{T}$ that bounds the number of queries in some accepting computation on any input word $w \in L(M_{2,i}, A)$;
- (v) there is a finite set that appears infinitely often in both the classes $L(\mathbf{M}_1, \mathbf{T}, A)$ and $L(\mathbf{M}_2, \mathbf{T}, A)$;
- (vi) for some fixed $f_1, f_2 \in \mathbf{T}$ independent of A , the sets $\overline{L_{f_1}(A)}$ and $\overline{L_{f_2}(A)}$ are in $L(\mathbf{M}_2, \mathbf{T}, A)$ and the intersection of the range of f_1 and the range of f_2 is finite.

Then there is a recursive A such that $\overline{L_{f_1}(A)}$ and $\overline{L_{f_2}(A)}$ are, respectively, $L(\mathbf{M}_1, \mathbf{T}, A)$ -immune and $L(\mathbf{M}_2, \mathbf{T}, A)$ -simple.

Observe that hypotheses (i) and (ii) together imply that the sum of a finite fixed number of functions of $\mathbf{T}$ is bounded by $2^{g(n)}$ for any $g \in \mathbf{T}$. This is the only nontrivial fact needed in applying the hypotheses to a construction similar to that used in the proof of Theorem 2.

Theorem 4 applies to the classes DEXT and NEXT, to the complexity classes specified by other hyperexponential time bounds $\exp(2, h, in)$ as defined above, to PQUERY and NPQUERY, and to many other complexity classes.

4. Refining nondeterminism. Some interesting work has been done in recent years regarding the existence of properly infinite hierarchies of relativized complexity classes where each class in the hierarchy is defined by bounding the number of nondeterministic steps the machines specifying the class are allowed to make. Kintala [6], [7] considered

oracle machines that operate in polynomial time. For each integer i and each set A , let $P(A)_{n^i}$ be the collection of all sets $L(M, A)$ where M operates in polynomial time and in every accepting computation on any input of length n , M can make at most n^i nondeterministic steps. In a similar way, for each integer i and each set A , define $P(A)_{(\log n)^i}$. Kintala showed that there are recursive sets A and B such that for all $i \geq 0$, $P(A)_{n^i} \subsetneq P(A)_{n^{i+1}}$, and $P(B)_{(\log n)^{i+1}} \subsetneq P(B)_{(\log n)^{i+2}}$. Xu, Doner, and Book [11] established a general separating theorem that yields Kintala's results as corollaries. Schöning and Book [9] established a general immunity theorem that yields as corollaries the existence of recursive sets A and B such that for all $i > 0$, $P(A)_{n^{i+1}}$ has a set that is $P(A)_{n^i}$ -immune and $P(B)_{(\log n)^{i+2}}$ has a set that is $P(B)_{(\log n)^{i+1}}$ -immune. Here we establish a general simplicity theorem that parallels the result of Schöning and Book.

A machine M operates in nondeterminism $g(n)$ if for every input string x to M , every computation of M on x has at most $g(|x|)$ nondeterministic steps.

Let $\mathbf{M}$ be a class of nondeterministic oracle machines, and let $\mathbf{T}$ and $\mathbf{G}$ be classes of nondecreasing functions. Assume that for each M in $\mathbf{M}$ there are functions $t \in \mathbf{T}$ and $g \in \mathbf{G}$ such that for every input string x to M , every computation of M on x makes at most $t(|x|)$ oracle queries, and M operates in nondeterminism g . Further, assume that for every $t \in \mathbf{T}$ and $g \in \mathbf{G}$ there is some M in $\mathbf{M}$ satisfying this condition. For every set A and $g \in \mathbf{G}$, define $D(\mathbf{M}, A)_g = \{(M, A) \mid M \in \mathbf{M} \text{ operates in nondeterminism } g\}$.

THEOREM 5. Let $\mathbf{M} = \{M_i \mid i \geq 1\}$ be an effective enumeration of a class of nondeterministic oracle machines that always halt, and let $\mathbf{T} = \{t[i] \mid i \geq 1\}$ and $\mathbf{G} = \{g[i] \mid i \geq 1\}$ be classes of running times such that $\langle \mathbf{M}, \mathbf{T}, \mathbf{G} \rangle$ satisfy the above conditions. Suppose that the following hold:

- (i) for every $t \in \mathbf{T}$ and $g \in \mathbf{G}$, $t(n) < 2^{g(n)}$ for almost all n ;
- (ii) for every $g \in \mathbf{G}$ and every set B , $\overline{L_g(A)} \in D(\mathbf{M}, B)_g$;
- (iii) there is a finite set F such that for all $g \in \mathbf{G}$, all $t \in \mathbf{T}$, and all sets B , there are infinitely many i such that $F = L(M_i, B)$, M_i operates in nondeterminism g , and for every input w , every accepting computation of M_i on w relative to B queries the oracle at most $t(|w|)$ times.

Then there is a recursive set A such that for every $g \in \mathbf{G}$, the set $\overline{L_g(A)}$ is $D(\mathbf{M}, A)_g$ -simple.

Again we turn to some examples before giving the proof of Theorem 5.

COROLLARY 5.1. There is a recursive set A such that for every $i > 0$, $P(A)_{n^i}$ has a set that is $P(A)_{n^{i-1}}$ -simple.

COROLLARY 5.2. There is a recursive set A such that for every $i > 1$, $P(A)_{(\log n)^i}$ has a set that is $P(A)_{(\log n)^{i-1}}$ -simple.

For every set B and every integer $i > 0$, let $\text{PQUERY}(B)_{n^i}$ and $\text{PQUERY}(B)_{(\log n)^i}$ be the restrictions of $\text{NPQUERY}(B)$ (or extensions of $\text{PQUERY}(B)$) defined analogously to $P(B)_{n^i}$ and $P(B)_{(\log n)^i}$.

COROLLARY 5.3. There are recursive sets A and B such that for every $i > 0$, $\text{PQUERY}(A)_{n^i}$ has a set that is $\text{PQUERY}(A)_{n^{i-1}}$ -simple and $\text{PQUERY}(B)_{(\log n)^{i+1}}$ has a set that is $\text{PQUERY}(B)_{(\log n)^i}$ -simple.

Fix an integer $h > 0$ and let $\mathbf{T} = \{\exp(2, h, in) \mid i > 0\}$. For every set B and every integer $i > 0$, let $\text{DTIME}(T, B)_{\exp(2, h, in)}$ be the collection of all sets $L(M, B)$ where M operates in time bounded by some function in $\mathbf{T}$ and in nondeterminism $\exp(2, h, in)$, and let $\text{DQUSP}(T, B)_{\exp(2, h, in)}$ be the collection of all sets $L(M, B)$ where M operates in space bounded by some function in $\mathbf{T}$, the number of oracle queries made in any of M 's accepting computations is bounded by some function in $\mathbf{T}$, and M operates in nondeterminism $\exp(2, h, in)$. See [3].

COROLLARY 5.4. There is a recursive set A such that $\text{DTIME}(T, A)_{\exp(2, h, in)}$ has a $\text{DTIME}(T, A)_{\exp(2, h, in)}$ -set simple set.

COROLLARY 5.5. *There is a recursive set A such that $DQUSP(T, A)_{\exp(2, h, in)}$ has a $DQUSP(T, A)_{\exp(2, h, in)}$ -simple set.*

For other examples of classes to which Theorems 3–5 apply, see [3], [9], [10], [11].

Now we turn to the proof of Theorem 5. Let us assume any standard polynomial-time computable tripling function, so that $i = \langle \xi, \eta, \zeta \rangle$ is one-one and onto from $\mathbb{N}^3$ to $\mathbb{N}$. We assume that the inverses (the projections) are also polynomial-time computable.

Proof of Theorem 5. First we need a recursive presentation of $\bigcup_g D(M, A)_g$. In order to do this, we build a new enumeration of machines which we call $M' = \{M'_i \mid i \geq 1\}$, by constructing each machine M'_i , $i = \langle \xi, \eta, \zeta \rangle$, behaving like the ξ th machine in M , M_ξ , with a clock for $t[\eta]$ that stops the machine if more than $t[\eta]$ queries are made, and with a clock for $g[\zeta]$ bounding the number of nondeterministic steps in the computations. Notice that the machines M'_i such that $i = \langle \xi, \eta, \zeta_0 \rangle$ for a fixed ζ_0 form a recursive presentation of $D(M, A)_{g[\zeta_0]}$.

From the index of the machine M'_i it is possible to recover the bounds t and g corresponding to M'_i . For the sake of clarity, when these bounds are needed we will say “let t, g be the bounds corresponding to the machine M ”, instead of indicating the index of t and g by means of the projections.

Construct the oracle A by performing in their natural order stages $0, 1, \dots$, as follows:

Stage 0

$A_0 := \{0\}^*$;

$m_0 := 0$;

$R_0 := \emptyset$;

end stage;

Stage n (for $n \geq 1$)

$R_n := R_{n-1} \cup \{n\}$;

$A_n := A_{n-1}$;

$m_n := \min \{m \mid w \text{ for any } w \text{ queried at earlier stages, and } t(m) < 2^{g(m)} \text{ for the bounds } t \text{ and } g \text{ corresponding to machine } M_j, \text{ for each } 1 \leq j \leq n\}$;

$A_n := A_n - \{0^{g(m_n)} \mid g \text{ is the nondeterminism bound corresponding to machine } M_j \text{ for each } 1 \leq j \leq n\}$;

if there is a $j \in R_n$ such that $0^{m_n} \in L(M_j, A)$

then

let j_n be the smallest such j ;

let t and g be the bounds corresponding to M_{j_n} ;

fix an accepting computation with query bound $t(m_n)$ and nondeterminism bound $g(m_n)$;

let w_n be the least word of length $g(m_n)$ not queried in this computation;

$R_n := R_n - \{j\}$;

$A_n := A_n \cup \{w_n\}$;

end if

end stage.

As in the construction in § 2, every index of the finite set F must remain in R_n from some stage on. So, case “then” must fail to appear infinitely often.

For any $g \in G$, once some machine operating in nondeterminism g has entered R_n at each stage in which case “then” does not occur no word of length $g(m_n)$ is allowed to remain in A ; so $L_g(A)$ is infinite.

On the other hand, let M_j be any machine of M' operating in nondeterminism g , and assume that $L(M_j, A)$ is infinite and that $L(M_j, A) \subseteq L_g(A)$; eventually a word

$0^{m_n} \in L(M_j, A)$ will be found, because after finitely many stages j must be the least index to be deleted from R_n . But this stage will add to A a word w_n of length $g(m_n)$ yielding $0^{m_n} \notin L_g(A)$. So, no infinite set in $D(M, A)_g$ is included in $L_g(A)$.

Hence, $L_g(A)$ is $D(M, A)_g$ -immune. But $\overline{L_g(A)} \in D(M, A)_g$ by hypothesis, so $\overline{L_g(A)}$ is $D(M, A)_g$ -simple. $\square$

It is possible to combine this result with the second immunity theorem in [9] in the same way as Theorems 2 and 4. No new ideas are needed. We omit this combined result.

Acknowledgments. The author wishes to thank Ronald Book and Uwe Schöning for their help, friendship, and kind sharing of their own ideas. Contributions of an anonymous referee to the clarity and correctness of this paper, particularly Theorem 5, are gratefully acknowledged. Finally, the author wishes to thank Ms. Leslie Wilson for her assistance with the manuscript.

REFERENCES

- [1] T. BAKER, J. GILL AND R. SOLOVAY, *Relativizations of the $P = ?NP$ question*, this Journal, 4 (1975), pp. 431–442.
- [2] R. BOOK, *Bounded query machines: on NP and PSPACE*, Theoret. Comput. Sci., 15 (1981), pp. 27–39.
- [3] R. BOOK, C. WILSON AND XU MEI-RUI, *Relativizing time, space, and time-space*, this Journal, 11 (1982), pp. 571–581.
- [4] W. GASARCH AND S. HOMER, *Relativizations of the exponential time hierarchy*, unpublished manuscript, 1982.
- [5] S. HOMER AND W. MAASS, *Oracle dependent properties of the lattice of NP sets*, Theoret. Comput. Sci., 24 (1983), pp. 279–289.
- [6] C. M. R. KINTALA, *Computations with a restricted number of nondeterministic steps*, Ph.D. dissertation, Pennsylvania State University, 1977.
- [7] C. M. R. KINTALA AND P. FISCHER, *Refining nondeterminism in relativized polynomial time-bounded computations*, this Journal, 9 (1980), pp. 46–53.
- [8] N. LYNCH, *On reducibility to complex or sparse sets*, J. Assoc. Comput. Mach., 22 (1975), pp. 341–345.
- [9] U. SCHÖNING AND R. BOOK, *Immunity, relativizations, and nondeterminism*, this Journal, 13 (1984), pp. 329–337.
- [10] A. SELMAN, XU MEI-RUI AND R. BOOK, *Positive relativizations of complexity classes*, this Journal, 12 (1983), pp. 565–579.
- [11] XU MEI-RUI, J. DONER AND R. BOOK, *Refining nondeterminism in relativized complexity classes*, J. Assoc. Comput. Mach., 30 (1983), pp. 677–685.