International Journal of Foundations of Computer Science

NONDETERMINISTICALLY SELECTIVE SETS [*]

LANE A. HEMASPAANDRA[†], ALBRECHT HOENE[‡], ASHISH V. NAIK[§], MITSUNORI OGIHARA[¶], ALAN L. SELMAN[‖], THOMAS THIERAUF[**], and JIE WANG[††]

Received 11 October 1994
Revised July 29, 1996
Communicated by D. P. Bovet

ABSTRACT

In this note, we study NP-selective sets (formally, sets that are selective via NPSV_t functions) as a natural generalization of P-selective sets. We show that, assuming $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$, the class of NP-selective sets properly contains the class of P-selective sets. We study several properties of NP-selective sets such as self-reducibility, hardness under various reductions, lowness, and nonuniform complexity. We prove many of our results via a "relativization technique," by using the known properties of P-selective sets. Using this technique, we strengthen a result of Longpré and Selman on hard promise problems and show that the result "$\mathrm{NP} \subseteq (\mathrm{NP} \cap \mathrm{coNP})/\mathrm{poly} \Rightarrow \mathrm{PH} = \mathrm{NP}^{\mathrm{NP}}$" is implicit in Karp and Lipton's seminal result on nonuniform classes.

Keywords: Computational Complexity, Nonuniform Complexity, Selectivity, Lowness

---

*Some of these results appeared in preliminary form in "Selectivity" (a 1993 ICCI Conference contribution; L. Hemaspaandra, A. Hoene, M. Ogiwara, A. Selman, T. Thierauf, and J. Wang).

†Department of Computer Science, University of Rochester, Rochester, NY 14627, USA. Supported in part by the NSF under grants NSF-CCR-8957604, NSF-INT-9116781/JSPS-ENG-207, and NSF-CCR-9322513. Work done in part while visiting the Tokyo Institute of Technology, the University of Amsterdam, and the University of Electro-communications–Tokyo.

‡Fachbereich 20, Informatik, Technische Universität Berlin, D-W-1000 Berlin 10, Germany. Supported in part by a DFG Postdoctoral Fellowship and the NSF under grant NSF-CCR-8957604. Work done in part while visiting the University of Rochester and the University of Washington–Seattle.

§Department of Computer Science, University of Chicago, Chicago IL 60637. Work done while at the State University of New York at Buffalo. Supported in part by the NSF under grant NSF-CCR-9002292.

¶Department of Computer Science, University of Rochester, Rochester, NY 14627, USA. Supported in part by the NSF under grant NSF-CCR-9002292 and the JSPS under grant NSF-INT-9116781/JSPS-ENG-207. Work done in part while visiting the State University of New York at Buffalo and while at the University of Electro-communications–Tokyo, Japan.

‖Department of Computer Science, State University of New York at Buffalo, Buffalo, NY 14260, USA. Supported in part by the NSF under grant NSF-CCR-9002292, NSF-INT-9123551, and NSF-CCR-9400229.

**Abt. Theoretische Informatik, Universität Ulm, D-W-7900 Ulm, Germany. Supported in part by a DFG Postdoctoral Stipend and by the NSF under grant NSF-CCR-8957604. Work done in part while visiting the University of Rochester.

††Department of Mathematical Sciences, University of North Carolina at Greensboro, Greensboro, NC 27412, USA. Supported in part by the NSF under grant NSF-CCR-9108899 and NSF-CCR-9424164.

1. Introduction

Given a set A, suppose that instead of solving the decision problem for A for an arbitrary input x, we are interested in obtaining the following partial information about A: which one of two given input strings x and y is more likely to be in A? More precisely, is there a polynomial-time algorithm that works as follows? If at least one of x or y belongs to A, then output a member of $\{x, y\}$ that belongs to A; else, if neither x nor y belongs to A, then either of the strings can be output. If such a polynomial-time algorithm exists, then A is said to be P-*selective*.[34] P-selective sets were defined by Selman[34] as a complexity-theoretic analog of semi-recursive sets in recursion theory. [20] Subsequently, this property has been studied by many researchers (e.g. see Ref. [39,19,18,11,40,10,32,7,1]).

This research has revealed that P-selective sets are an important tool in studying several important structural concepts such as function complexity classes,[19,32,7,1,12] reducing search to decision and self-reducibility,[18,41,11] and promise problems.[36,29] A survey of the current state of knowledge about selective sets can be found in Denny-Brown *et al.*.[13]

Selman [34] proved that SAT, the set of all satisfiable boolean formulas, is P-selective if and only if P = NP. Thus P is the largest level of the polynomial hierarchy that is known to contain only P-selective sets. In as much as the power of nondeterministic computation is one unifying theme of complexity theory, it is natural to wonder whether some broader notion of selectivity can capture more of the polynomial hierarchy. Thus motivated, we study the class of NP-selective sets—sets having an "NP function"[8] that serves as a selector. That is, a language L is NP-selective if it has a selector function that is computable by a *single-valued and total* NP *transducer*. (A formal definition is given in Section 2.)

We ask whether each NP set has a nondeterministic but total polynomial-time selector. Our results provide a negative answer to this question despite the fact that NP-selectivity is a more inclusive notion than P-selectivity and that every set in NP $\cap$ coNP is NP-selective. We study several properties of NP-selective sets such as self-reducibility, hardness under various reductions, lowness, and nonuniform complexity. Thus, in this note, we construct a theory of NP-selective sets that is parallel to that of P-selective sets.

Self-reducibility[31] has widely been discussed as a property possessed by most "natural" sets such as SAT. It is known that a language L is in P if and only L is P-selective and Turing self-reducible.[11] Analogously, we show that a language L is in NP $\cap$ coNP if and only if L is Turing self-reducible and NPMV_t-selective. As a consequence of this, all NP sets are NP-selective only if NP = coNP. Wang[41] has recently shown that such characterizations hold for arbitrary time complexity classes.

One important line of research on P-selective sets has been to determine the strongest consequence of NP sets reducing to a P-selective set under various reductions.[27] Selman[35] showed that if there exists a P-selective set that is NP-hard under *positive truth-table* reductions, then P = NP. Buhrman, Torenvliet, and van Emde Boas[10] generalized this to show that if there exists a P-selective that is NP-hard under *positive Turing reductions*, then P = NP. Recently, Agrawal and Arvind,[1] Beigel, Kummer, and Stephan,[7] and Ogihara[32] independently have proved that the existence of a $\leq_{btt}^{\mathrm{P}}$-hard P-selective set for NP implies P = NP. We show that the existence of an NP-selective set that is NP-hard under $\leq_{\gamma}$ or $\leq_{pos}^{\mathrm{P}}$ or $\leq_{btt}^{\mathrm{P}}$ reductions implies that NP = coNP. These results are described in Section 3.

Section 4 studies the lowness and nonuniform complexity of NP-selective sets. We show that NP-selective sets are of simple nonuniform complexity; all NP-selective sets are in (NP $\cap$ coNP)/poly. Although inclusion in the third level of the low hierarchy[33] for all NP-selective sets in NP follows immediately from this, we show the stronger result that NP-selective sets are as low as P-selective sets: all NP-selective sets in NP are in the second level of the low hierarchy. This upper bound on the lowness of the NP-selective sets is optimal (with respect to relativizable proof techniques), due to the recently proven lower bound on the lowness of P-selective sets.[3] As to extended lowness,[5] we note that all NP-selective sets are ExtendedLowΘ_3.

Several of our results are obtained by relativizing known results for P-selective sets. In Section 5, we apply this technique to study the properties of certain promise problems. Longpré and Selman[29] showed that if a set A is $\leq_{d}^{\mathrm{P}}$-hard for NP, then a natural promise problem associated with A, PP-A, is Turing-hard for NP. We improve this to show that: If A is $\leq_{pos}^{\mathrm{P}}$-hard for NP, then PP-A is Turing-hard for NP.

Finally, using the relativization technique, we show that the result "NP $\subseteq$ (NP $\cap$ coNP)/poly $\Rightarrow$ PH = Σ_2^{P}," first explicitly proved by Abadi, Feigenbaum, and Kilian,[2] and Kämper,[21] is implicit in Karp and Lipton's (Ref. [22]) seminal result: NP $\subseteq$ P/poly $\Rightarrow$ PH = Σ_2^{P}.

2. Definitions

All languages are defined over strings in the alphabet $\{0,1\}$ and all functions map strings to strings. We use the standard definitions of nondeterministic function classes[8] (see also Ref. [37]) to formalize our notion of a nondeterministic selector. A transducer M outputs a string y on input x if there exists an accepting path of M on input x that outputs y. Such transducers compute *partial, multivalued functions*. For each partial, multivalued function f, let $dom(f) = \{x \mid \exists y(y \text{ is an output of } f(x))\}$. We say that f is a *total* function if $dom(f) = \{0,1\}^*$. A partial function is *single-valued* if for all $x \in dom(f)$, $\|\{y \mid y \text{ is an output of } f(x)\}\| = 1$.

Definition 1 *Ref. [8]*

1. NPMV *is the class of all partial multivalued functions f such that there exists a nondeterministic polynomial-time transducer M such that for all strings x and y, $M(x)$ outputs y if and only if $f(x)$ maps to y.*
2. NPSV *is the class of all* single-valued NPMV *functions.*
3. NPMV$_t$ *is the class of all total functions in* NPMV.

4. NPSV_t *is the class of all* single-valued NPMV_t *functions.*

5. PF *is the class of functions computable by deterministic poly-time transducers.*

The following definitions are useful for studying partial multivalued functions.

Definition 2 *Ref. [8,37]*

1. *Given a partial multivalued function f, for all x, we define set-$f(x) = \{y \mid y$ is an output of $f(x)\}$.*

2. *Given partial multivalued functions f and g, g is an* extension *of f if $dom(g) \supseteq dom(f)$ and for all $x \in dom(f)$, set-$g(x) =$ set-$f(x)$.*

3. *Given partial multivalued functions f and g, g is a* refinement *of f if $dom(g) = dom(f)$ and for all $x \in dom(g)$, set-$g(x) \subseteq$ set-$f(x)$.*

Our next definition can be used to define selectivity for any partial, multivalued function class.

Definition 3 *Ref. [19]* **[Selectivity by Classes of Functions]**

1. *Let $\mathcal{FC}$ be a class of (possibly multivalued, possibly partial) functions mapping from Σ^* to Σ^*. A set A is $\mathcal{FC}$-selective if there is a function $f \in \mathcal{FC}$ so that, for every $x, y \in \Sigma^*$,*

   (a) *set-$f(x, y) \subseteq \{x, y\}$, and*

   (b) *if $x \in A$ or $y \in A$, then $\emptyset \neq$ set-$f(x, y) \subseteq A$.*

2. *Let $\mathcal{FC}$ be any class of functions mapping from Σ^* to Σ^*. We define $\mathcal{FC}$-sel $= \{A \mid A$ is $\mathcal{FC}$-selective$\}$.*

The function f is called the selector *functions for A.*

Observe that the definition of a P-selective set is identical to that of a PF_t-selective set. We say that a set L is NP-selective if L is NPSV_t-selective. We will use P-sel to denote the class of P-selective sets, NP-sel to denote the class of NP-selective sets, and NPMV_t-sel to denote the class of NPMV_t-selective sets. In this note, we will focus on NP-selective sets and NPMV_t-selective sets. Hemaspaandra *et al.*[19] study the partial counterparts, NPSV-selective sets and NPMV-selective sets.

The following proposition, although easy to prove, will be extensively used in the later sections.

Proposition 1

1. *If L is* NP*-selective, then there is an* NPSV_t*-selector for L such that $(\forall x, y \in \Sigma^*)[f(x,y) = f(y,x)]$.*

2. $\mathrm{NPSV}_t = \mathrm{PF}_t^{\mathrm{NP}\cap\mathrm{coNP}}$.

3. $\mathrm{NP} = \mathrm{NP}^{\mathrm{NPSV}_t}$.[a]

[a] We use the natural notion of access to a single-valued function oracle; the value of the function on the queried string is returned.

4. $\mathrm{NPSV}_t = (\mathrm{NPSV}_t)^{\mathrm{NPSV}_t}$.

We assume that the reader is familiar with the standard notations and definitions of polynomial-time reducibilities.[27] We will use the γ *reductions* of Adleman and Manders, which are the same as *many-one strong nondeterministic reductions*.[4,28]

We say that $A \leq_\gamma B$ if there is a nondeterministic polynomial-time transducer N such that

(i) for each string x, $N(x)$ has at least one accepting path $p(x)$, and

(ii) for each accepting path $p(x)$ of $N(x)$, it holds that

$$x \in A \iff output(x, p(x)) \in B,$$

where $output(x, p(x))$ denotes the output value on path $p(x)$.

For sets A and B, we let $A \oplus B$ denote the disjoint union of A and B, namely, $A \oplus B = \{0x \mid x \in A\} \cup \{1x \mid x \in B\}$.

The standard definition of self-reducibility that is used in most contemporary research in complexity theory was given by Meyer and Paterson.[31]

Definition 4 *Ref. [31] A polynomial time computable partial order $<$ on Σ^* is* OK *if there exists a polynomial p such that,*

1. *each strictly decreasing chain is finite and every finite $<$-decreasing chain is shorter than p of the length of its maximum element, and*

2. *for all $x, y \in \Sigma^*$, $x < y$ implies that $|x| \leq p(|y|)$.*

Definition 5 *Ref. [31] A set L is* Turing self-reducible *if there is an OK partial order $<$ and a deterministic polynomial time-bounded oracle machine M such that M accepts L with oracle L and, on any input x, M asks its oracle only about strings strictly less than x in the OK partial order $<$. If the Turing self-reduction of the oracle machine M in fact is also a polynomial-time disjunctive (conjunctive) truth-table reduction, then L is said to be disjunctive (conjunctive) self-reducible.*

Lowness and extended lowness are used here as defined, respectively, by Schöning[33] and Balcázar, Book, and Schöning.[5]

Definition 6

1. *[Ref. [33]] For each $k \geq 1$, define $Low_k = \{L \in \mathrm{NP} \mid \Sigma_k^{\mathrm{P}, L} = \Sigma_k^{\mathrm{P}}\}$, where the Σ_k^{P}[38] are the Σ levels of the polynomial hierarchy .*

2. *[Ref. [5]] For each $k \geq 2$, define $ExtendedLow_k = \{L \mid \Sigma_k^{\mathrm{P}, L} = \Sigma_{k-1}^{\mathrm{P}, \mathrm{SAT} \oplus L}\}$. For each $k \geq 3$, define*

$$ExtendedLow\Theta_k = \{L \mid \mathrm{P}^{(\Sigma_{k-1}^{\mathrm{P}, L})[\mathcal{O}(\log n)]} \subseteq \mathrm{P}^{(\Sigma_{k-2}^{\mathrm{P}, \mathrm{SAT} \oplus L})[\mathcal{O}(\log n)]}\},$$

where $\mathrm{P}^{(\Sigma_{k-1}^{\mathrm{P}, L})[\mathcal{O}(\log n)]}$ denotes the class of languages computable in polynomial time by querying at most $\mathcal{O}(\log n)$ strings to a $\Sigma_{k-1}^{\mathrm{P}}$ oracle.

The first question that arises is whether NP-sel properly contains P-sel. The following theorem answers this question conditionally in the affirmative.

Theorem 1 P-sel $\neq$ NP-sel *if and only if* $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$.

Proof. If $\mathrm{P} = \mathrm{NP} \cap \mathrm{coNP}$ then all NPSV_t functions are computable in polynomial time (Ref. [8], or see Part 2 of Proposition 1), and thus P-sel = NP-sel. By the results of Selman,[35] it follows that if $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$, then there is a set $B \in (\mathrm{NP} \cap \mathrm{coNP}) - \mathrm{P}$ such that B is not P-selective. However, observe that all sets in $\mathrm{NP} \cap \mathrm{coNP}$ are NP-selective. $\square$

Let us now turn to our main question: how do various properties of NP-selective sets compare with those of P-selective sets? Buhrman, van Helden, and Torenvliet[11] showed that if a Turing self-reducible set is P-selective, then it is in P. The next theorem is a nondeterministic analog of this result.

Theorem 2 *If a set A is polynomial-time Turing self-reducible and is $\leq_\gamma$-reducible to $\overline{S} \oplus S$, for some NPMV_t-selective set S, then A is in $\mathrm{NP} \cap \mathrm{coNP}$.*

Proof. Let A be polynomial-time Turing self-reducible via machine M, let A be $\leq_\gamma$-reducible to $\overline{S} \oplus S$ via a nondeterministic machine N, and let S be NPMV_t-selective via a nondeterministic machine F. Let x be a string whose membership in A we are testing. Suppose that N on x outputs cu for some accepting computation path so that $x \in A$ if and only if $\chi_S(u) = c$. Let us fix such c and u. For any v and w ($v \neq w$), let us write $v <_F w$ if $w \in$ *set-*$F(v, w)$; that is, it is witnessed by F that $v \in S \Rightarrow w \in S$. By convention, let $\perp$ and $\top$ be strings such that $\perp <_F v$ for any v, and $v <_F \top$ for any v.

Consider a simulation M' of M on x defined as follows: The simulation will use two strings, a and b. Initially, a is set to $\top$ and b is set to $\perp$. M' simulates M such that when M makes the i^{th} query y_i, M' performs the following steps:

1. Simulate N on y_i to compute $d_i v_i$ such that $y_i \in A \Leftrightarrow \chi_S(v_i) = d_i$.
2. If $a <_F v_i$, then choose the branch corresponding to $v_i \in S$.

   If $v_i <_F b$, then choose the branch corresponding to $v_i \notin S$.

   If $b <_F v_i <_F a$, then simulate F on (u, v_i). If $u <_F v_i$, then set a to v_i and choose the branch corresponding to $v_i \in S$. If $v_i <_F u$, then set b to v_i and choose the branch corresponding to $v_i \notin S$.

Let r be 1 if M accepts in the simulation and 0 if M rejects in the simulation. Let a_0 and b_0 be the final values of a and b, respectively. Let i and j be such that a_0 is set to v_i and b_0 is set to v_j. The following properties hold:

1. $b_0 <_F u <_F a_0$.
2. If $b_0 \notin S$ and $a_0 \in S$, then M^A on x accepts if and only if $r = 1$, so $\chi_A(x) = r$.
3. If $b_0 \in S$, then $u \in S$, so $\chi_A(x) = c$.
4. If $a_0 \notin S$, then $u \notin S$, so $\chi_A(x) = 1 - c$.

Suppose $r = c$. Then we have $\chi_A(x) = r$ if and only if $a_0 \in S$. So, $\chi_A(x) = r$ if and only if $v_i \in S$ if and only if $\chi_A(y_i) = d_i$. If this case holds, let $z = y_i$ and $e = 1$ if $r = d_i$ and 0 otherwise. Suppose $r = 1 - c$. Then we have $\chi_A(x) = r$ if and only if $b_0 \notin S$. So, $\chi_A(x) = r$ if and only if $v_j \notin S$ if and only if $\chi_A(y_j) = 1 - d_j$. If this case holds, let $z = y_j$ and $e = 0$ if $r = d_j$ and 1 otherwise. It holds that $x \in A$ if and only if $\chi_A(z) = e$.

Thus, M' will find strings z and e such that $\chi_A(x) = 1$ if and only if $\chi_A(z) = e$. It is not hard to see that (i) there is some computation path of M' that finds such z and e, (ii) the simulation runs in time polynomial in $|x|$, and (iii) z is a string appearing in the self-reduction tree of M on x. By repeating the above simulation polynomially many times, we eventually find strings z' and e' such that $x \in A$ if and only if $\chi_A(z') = e'$, and M on z' determines the membership of z' in A in polynomial time without making any query. Thus, we have nondeterministic polynomial time procedures for both membership in A and non-membership in A. □

Corollary 1 *If a set A is Turing self-reducible and 1-tt reducible to an* NPMV_t*-selective set, then $A \in \mathrm{NP} \cap \mathrm{coNP}$.*

Corollary 2 *If there exists an* NPMV_t*-selective set L such that L is $\leq_\gamma$-hard for* NP*, then* NP = coNP.

Since PSPACE, PP and $\oplus$P contain Turing self-reducible complete languages,[15] a similar relationship holds for these classes.

Corollary 3 *If every language in* PSPACE *(respectively,* $\oplus$P, PP*) is γ reducible to $S \oplus \overline{S}$ for some* NP*-selective set S, then* $\mathrm{NP} \cap \mathrm{coNP} = \mathrm{PSPACE}$ *(respectively,* $\mathrm{NP} \cap \mathrm{coNP} \supseteq \oplus\mathrm{P}$, $\mathrm{NP} \cap \mathrm{coNP} = \mathrm{PP}$*).*

It follows from Corollary 1 that $\mathrm{NP} \subseteq \mathrm{NPMV}_t$-sel if and only if NP = coNP. However the next theorems demonstrate that (unlikely) assertions such as $\mathrm{NP} \subseteq \mathrm{NPMV}_t$-sel are equivalent to (equally unlikely) assertions about the complexity of computing satisfying assignments, from which, we see that the above implication holds directly without use of Theorem 2 or its corollaries.

Let *sat* denote the partial multivalued function that, on input x, computes a satisfying assignment of x, if it exists. Note that *sat* belongs to the class NPMV.

Theorem 3 *The following are equivalent:*

1. SAT *is* NPSV_t*-selective.*
2. $\mathrm{NP} \subseteq \mathrm{NPSV}_t$-sel.
3. *There is a single-valued refinement g of sat such that some extension of g to a total function belongs to* NPSV_t*.*
4. *For every $f \in$* NPMV*, there is a single-valued refinement g of f such that some extension of g to a total function belongs to* NPSV_t*.*
5. NP = coNP*.*

Proof. The fact that assertion (1) is equivalent to (2) follows by NP-completeness of SAT and that (3) is equivalent to (4) follows by a result of Selman.[37] It suffices to show that (3) $\Rightarrow$ (5) $\Rightarrow$ (1) and that (1) $\Rightarrow$ (3).

To see that (3) implies (5), let g be a single-valued refinement of sat and let h be an extension of g that belongs to NPSV_t. Observe that the following NP machine M accepts $\overline{\mathrm{SAT}}$. On input x, M simulates $h(x)$. If the output of $h(x)$ is a satisfying assignment of x, it rejects, else it accepts x. Thus (5) holds. It is easy to observe that (5) implies (1), since all sets in $\mathrm{NP} \cap \mathrm{coNP}$ are NPSV_t-selective.

Finally, suppose that SAT is NPSV_t-selective. Then, we can find a satisfying assignment of a boolean formula by an NPSV_t function that generates a satisfying assignment by traversing the disjunctive self-reduction tree of SAT and using the NPSV_t-selector to decide, at each node, whether to take the left branch or the right branch. If the leaf reached is a satisfying assignment then output the assignment, else output a special string $\perp$. This proves that (1) implies (3). □

Theorem 4 *The following are equivalent:*

1. SAT *is* NPMV_t*-selective.*
2. $\mathrm{NP} \subseteq \mathrm{NPMV}_t\text{-sel}$.
3. *There is a refinement g of sat such that some extension of g to a total function belongs to* NPMV_t.
4. *For every $f \in \mathrm{NPMV}$, there is a refinement g of f such that some extension of g to a total function belongs to* NPMV_t.
5. $\mathrm{NP} = \mathrm{coNP}$.

The proof of Theorem 4 is similar to that of Theorem 3, though a bit of care has to be used in the arguments that Part (1) implies Part (3) and that Part (3) implies Part (5) to correctly handle, respectively, the fact that multiple leaves may be reached and that multiple outputs may occur.

Next, we investigate the existence of NP-hard NP-selective sets under various reducibilities. Buhrman, Torenvliet, and van Emde Boas[10] have proved that if there exists a P-selective set that is $\leq^{\mathrm{P}}_{pos}$-hard for NP, then $\mathrm{P} = \mathrm{NP}$. Also, recent research[1,7,32] has revealed that if there exists a P-selective set that is $\leq^{\mathrm{P}}_{btt}$-hard for NP, then $\mathrm{P} = \mathrm{NP}$. We now obtain analogous results for NP-selective sets, which are proved by relativizing the corresponding results for P-selective sets.

Lemma 1 *(Relativizing Ref. [10]) If $A \leq^{\mathrm{P}}_{pos} B$, B is PF^L_t-selective for some set L, $B \neq \emptyset$ and $B \neq \Sigma^*$, then $A \leq^{\mathrm{P},L}_m B$ and hence A is PF^L_t-selective.*

Theorem 5 *If $A \leq^{\mathrm{P}}_{pos} B$ and B is* NP*-selective, then*

1. *A is* NP*-selective, and*
2. *if $B \neq \Sigma^*$ and $B \neq \emptyset$ then $A \leq^{NPSV_t}_m B$.*

Proof. Let B be NP-selective with selector $f \in \mathrm{NPSV}_t$. There exists a language $L \in \mathrm{NP} \cap \mathrm{coNP}$ such that B is PF^L_t-selective. Thus by Lemma 1, $A \leq^{\mathrm{P},L}_m B$. Since $L \in \mathrm{NP} \cap \mathrm{coNP}$, by Proposition 1, Part 2, it follows that $A \leq^{NPSV_t}_m B$ and that A is NP-selective. □

Corollary 4 *If $A \leq^{\mathrm{P}}_{pos} \overline{A}$ and A is* NP*-selective, then $A \in \mathrm{NP} \cap \mathrm{coNP}$.*

Similarly, it is easy to see that if $A \leq_\gamma \overline{A}$ and A is NP-selective, then $A \in \mathrm{NP} \cap \mathrm{coNP}$.

Corollary 5 *If there exists an* NP*-selective set that is* $\leq^{\mathrm{P}}_{pos}$*-hard for* NP*, then* NP = coNP*.*

Lemma 2 *(Relativizing Ref. [1,7,32]) If B is* $\leq^{\mathrm{P}}_{btt}$*-hard for* NP *and B is* PF^L_t*-selective for some set L, then* $\mathrm{P}^L = \mathrm{NP}^L$*.*

Theorem 6 *If there exists an* NP*-selective set that is* $\leq^{\mathrm{P}}_{btt}$*-hard for NP, then* NP = coNP*.*

Proof. Let B be NP-selective with selector $f \in \mathrm{NPSV}_t$. There exists a set $L \in \mathrm{NP} \cap \mathrm{coNP}$ such that B is PF^L_t-selective. By Lemma 2, it follows that $\mathrm{P}^L = \mathrm{NP}^L$, which implies that NP = coNP. □

Thus, not only is NP unlikely to be contained in the class of NP-selective sets, but even NP-selective sets that are hard for NP with respect to such powerful reductions as $\leq_\gamma$, $\leq^{\mathrm{P}}_{btt}$ or $\leq^{\mathrm{P}}_{pos}$ reductions are unlikely to exist, unless NP = coNP.

Our results for the general question, "Is $\mathcal{C}$ contained in $\mathcal{FC}$-sel?" for $\mathcal{C} = \{\mathrm{NP}, \mathrm{coNP}\}$ and $\mathcal{FC} = \{\mathrm{NPSV}_t, \mathrm{NPMV}_t\}$ can be summarized by the following table. Results about the partial function classes NPMV and NPSV were obtained in Ref. [19] and have been included here for completeness.

Theorem 7 *The following results hold:*

$\mathcal{FC}$	$\mathrm{NP} \subseteq \mathcal{FC}$*-selective*	$\mathrm{coNP} \subseteq \mathcal{FC}$*-selective*
$NPSV_t$	*holds iff* NP = coNP	*holds iff* NP = coNP
NPSV [19]	*holds if* NP = coNP *holds only if* $\mathrm{NP}^{\mathrm{NP}} = \mathrm{coNP}^{\mathrm{NP}}$	*holds iff* NP = coNP
$NPMV_t$	*holds iff* NP = coNP	*holds iff* NP = coNP
NPMV [19]	*holds (without any assumption)*	*holds iff* NP = coNP

3. Lowness and Nonuniform Complexity

Ko showed that P-selective sets have low circuit complexity; they are in P/poly (see Ref. [22] for formal definitions). By relativizing Ko's result (see also Ref. [29]), we can show that NP-selective sets have low circuit complexity too; they are in (NP ∩ coNP)/poly.

Lemma 3 *Ref. [23] For all sets A and L, if A is* PF^L_t*-selective, then* $A \in \mathrm{P}^L/\mathrm{poly}$*.*

Theorem 8 NP-sel ⊆ (NP ∩ coNP)/poly*.*

Proof. Suppose A is NP-selective via a selector $f \in \mathrm{NPSV}_t$. Then, by Part 2 of Proposition 1, there exists a language L in NP∩coNP such that A is PF^L_t-selective. By Lemma 3, A is in $\mathrm{P}^L/\mathrm{poly}$. Since $\mathrm{P}^{\mathrm{NP}\cap\mathrm{coNP}} = \mathrm{NP} \cap \mathrm{coNP}$, the theorem follows. □

From Theorem 8, it follows immediately that the NP-selective sets in NP are Low_3 (since (NP∩coNP)/poly∩NP ⊆ (NP/poly)∩(coNP/poly)∩NP ⊆ (coNP/poly)∩NP, which due to Kämper[21] is Low_3). However, we will directly prove that the NP-selective sets in NP are even lower. Indeed, they are as low as P-selective sets. We use the following restatement of a theorem by Longpré and Selman[29] (see also Ref. [23]) to prove our theorem.

Lemma 4 *Ref. [29,23] If* A *in* NP *is* P^L*-selective for some* L*, then* $\Sigma_2^{\mathrm{P},A} \subseteq \Sigma_2^{\mathrm{P},L}$.

Theorem 9 *The* NP*-selective sets in* NP *are* Low_2.

Proof. Suppose $A \in \mathrm{NP}$ is NP-selective. Then by Part 2 of Proposition 1, there exists a set $L \in \mathrm{NP} \cap \mathrm{coNP}$ such that A is PF_t^L-selective. By applying Lemma 4, and by using the fact that $L \in \mathrm{NP} \cap \mathrm{coNP}$ is Low_1,[33] it follows that $\Sigma_2^{\mathrm{P},A} \subseteq \Sigma_2^{\mathrm{P}}$. □

Hemaspaandra *et al.*[19] have generalized this result to show that all NPSV-selective sets in NP are Low_2.

A set A is said to be $\mathrm{GeneralizedLow}_2$ if $\Sigma_2^{\mathrm{P},A} = \Sigma_2^{\mathrm{P}}$.[5] We obtain the following generalized lowness result for NP-selective sets by relativizing the following result of Balcázar, Book, and Schöning[5] (see also Ref. [29,22]).

Lemma 5 *Ref. [5] If* A *is Turing self-reducible and* A *is Turing reducible to a P-selective set then* A *is* $GeneralizedLow_2$.

Theorem 10 *If* A *is Turing self-reducible and* A *is Turing reducible to an* NP*-selective set, then* A *is* $GeneralizedLow_2$.

Proof. Suppose A is Turing self-reducible and Turing reducible to an NP-selective set. Then A is Turing reducible to a PF_t^L-selective set for some set $L \in \mathrm{NP} \cap \mathrm{coNP}$. By relativizing Lemma 5, it follows that $A \in \Sigma_2^{\mathrm{P},L}$. Since $\mathrm{NP} = \mathrm{NP}^{\mathrm{NP}\cap\mathrm{coNP}}$, $\Sigma_2^{\mathrm{P},L} \subseteq \Sigma_2^{\mathrm{P}}$, which completes the proof. □

As to extended lowness, Köbler[24] has shown that $(\mathrm{NP} \cap \mathrm{coNP})/\mathrm{poly}$ is $\mathrm{ExtendedLow}\Theta_3$. From this and Theorem 8, we can immediately conclude that the NP-selective sets are $\mathrm{ExtendedLow}\Theta_3$.[b]

4. Applications of the Relativization Technique

The proofs of Theorems 5, 6, 8, and 9 used relativization of well-known results on P-selective sets to obtain the corresponding properties of NP-selective sets. In this section, we use relativization to obtain some results of independent interest. We note that a nice example of this approach can be found in the literature. Buhrman and Torenvliet[9] have noted that, since the deterministic time hierarchy theorem relativizes,[42,16] simply by relativizing the deterministic time hierarchy it follows that (for each k) the Δ_k level of the polynomial hierarchy differs from the Δ_k level of the exponential hierarchy .[16,17] This result is incomparable with the recent result of Mocas[30] that for all k, $\mathrm{P}^{\mathrm{NP}[n^k]} \subset \mathrm{NEXP}$, where $\mathrm{NEXP} = \cup_{c>0} NTIME[2^{n^c}]$. Of course, since the nondeterministic time hierarchy theorem relativizes,[42,16] it similarly holds that for each k the Σ_k level of the polynomial hierarchy differs from the Σ_k level of the exponential hierarchy.

4.1. Relativizing Karp-Lipton

We note now that a result proved a half-decade ago by Kämper[21] and Abadi, Feigenbaum, and Kilian[2] is, in fact, merely a relativized version of a famous 1980 result by Karp and Lipton.[22]

[b] Very recently, Köbler[25] has shown that NP-selective sets are $\mathrm{ExtendedLow}_2$.

Theorem 11 *Ref. [22] For all oracles L, if $\mathrm{NP}^L \subseteq \mathrm{P}^L/\mathrm{poly}$, then $\mathrm{PH}^{\mathrm{L}} = \Sigma_2^{\mathrm{P},L}$.*

Corollary 6 *Ref. [2,21] If $\mathrm{NP} \subseteq (\mathrm{NP} \cap \mathrm{coNP})/\mathrm{poly}$, then $\mathrm{PH} = \Sigma_2^{\mathrm{P}}$.*

Proof. Suppose that $\mathrm{NP} \subseteq (\mathrm{NP} \cap \mathrm{coNP})/\mathrm{poly}$. Since $\mathrm{NP}^{\mathrm{NP}\cap\mathrm{coNP}} = \mathrm{NP}$, there exists some $L \in \mathrm{NP} \cap \mathrm{coNP}$ such that $\mathrm{SAT} \in \mathrm{P}^L/\mathrm{poly}$. Thus, by downward closure of NP under $\leq_m^{\mathrm{P}}$ reductions, $\mathrm{NP}^L = \mathrm{NP} \subseteq \mathrm{P}^L/\mathrm{poly}$. Thus, by Theorem 11, $\mathrm{PH}^L \subseteq \Sigma_2^{\mathrm{P},L}$. Since $L \in \mathrm{NP} \cap \mathrm{coNP}$, it follows that $\mathrm{PH}^L = \mathrm{PH}$ and $\Sigma_2^{\mathrm{P},L} = \Sigma_2^{\mathrm{P}}$. The theorem follows. □

Recently, the Karp-Lipton result has been improved[6,26] to show that that if $\mathrm{NP} \subseteq \mathrm{P}/\mathrm{poly}$, then $\mathrm{PH} = \mathrm{ZPP}^{\mathrm{NP}}$. Further, it has similarly been noted[26] that $\mathrm{NP} \subseteq (\mathrm{NP} \cap \mathrm{coNP})/\mathrm{poly} \Rightarrow \mathrm{PH} = \mathrm{ZPP}^{\mathrm{NP}}$, which follows, again, simply via relativization.

4.2. Hard Promise Problems

With the help of relativization, we improve a result of Longpré and Selman[29] about promise problems.

Definition 7 *Ref. [14,29]*

1. *Given any set A, we say that a set B* is a solution of PP-A *if for all strings x and y,*
$$(x \in A \oplus y \in A) \Rightarrow [\langle x, y\rangle \in B \Leftrightarrow x \in A].$$
2. *For sets C and D, we say* PP-C is Turing-hard for D *if for every solution L of* PP-C *it holds that $D \leq_T^{\mathrm{P}} L$. For any class $\mathcal{E}$, we say* PP-C *is Turing-hard for $\mathcal{E}$ if* PP-C *is Turing-hard for each set in $\mathcal{E}$.*

The following relationship between NP-selectivity and the complexity of solutions follows easily from Proposition 1.

Lemma 6 *For each set A, A is* NP*-selective if and only if* PP-A *has a solution in* $\mathrm{NP} \cap \mathrm{coNP}$.

Lemma 7
1. *Let $A \leq_{pos}^{\mathrm{P}} B$ and let L be any solution of* PP-B. *Then there is a function g computable by a polynomial-time machine with oracle L such that $x \in A \Leftrightarrow g(x) \in B$.*
2. *If A is polynomial-time Turing self-reducible and $A \leq_{pos}^{\mathrm{P}} B$, then* PP-$B$ *is Turing-hard for A.*

Proof. Let $A \leq_{pos}^{\mathrm{P}} B$ and L be a solution for PP-B. We need to show that $A \leq_m^{\mathrm{P},L} B$. Define $f_L(x, y) = x$ if $\langle x, y\rangle \in L$ and $f_L(x, y) = y$ otherwise. f_L is a selector for B and f_L is computable in polynomial time relative to L. Since $A \leq_{pos}^{\mathrm{P}} B$ and B is PF_t^L-selective, by Lemma 1, $A \leq_m^{\mathrm{P},L} B$. Part 1 now follows immediately since this implies that there exists $g \in \mathrm{PF}_t^L$ such that $x \in A \Leftrightarrow g(x) \in B$.

To prove Part 2, observe that A is PF_t^L-selective. By hypothesis, A is Turing self-reducible, and thus, by relativizing the Buhrman, van Helden, and Torenvliet theorem,[11] it follows that $A \in \mathrm{P}^L$. Thus $A \leq_T^{\mathrm{P}} L$, which proves the lemma. □

Longpré and Selman showed that if a set A is NP-complete under disjunctive reductions, then PP-A is Turing-hard for NP. We prove this consequence under the assumption that A is $\leq_{pos}^{\mathrm{P}}$-hard for NP.

Theorem 12 *If A is $\leq^{\mathrm{P}}_{pos}$-hard for* NP, *then* PP-A *is Turing-hard for* NP.

Proof. If A is $\leq^{\mathrm{P}}_{pos}$ hard for NP, then SAT $\leq^{\mathrm{P}}_{pos}$ A. The theorem now follows from Part 2 of Lemma 7. □

Acknowledgements

We are grateful to Joel Seiferas for helpful discussions.

References

1. M. Agrawal and V. Arvind. Polynomial time truth-table reductions to P-selective sets. In *Proceedings of 9th Annual IEEE Conference on Structure in Complexity Theory*, pages 24–30, 1994.
2. M. Abadi, J. Feigenbaum, and J. Kilian. On hiding information from an oracle. *Journal of Computer and System Sciences*, 39:21–50, 1989.
3. E. Allender and L. Hemachandra. Lower bounds for the low hierarchy. *JACM*, 39(1):234–250, 1992.
4. L. Adleman and K. Manders. Reducibility, randomness, and intractability. In *Proceedings of 9th ACM Symposium Theory of Computing*, pages 151–163, 1977.
5. J. Balcázar, R. Book, and U. Schöning. Sparse sets, lowness, and highness. *SIAM Journal on Computing*, 15:739–747, 1986.
6. N. Bshouty, R. Cleve, S. Kannan, and C. Tamon. Oracles and queries that are sufficient for exact learning. In *Proceedings of 7th ACM Conference on Computational Learning Theory*, pages 249–254, 1994.
7. R. Beigel, M. Kummer, and F. Stephan. Approximable sets. In *Proceedings of 9th Annual IEEE Conference on Structure in Complexity Theory*, pages 12–23, 1994.
8. R. Book, T. Long, and A. Selman. Quantitative relativizations of complexity classes. *SIAM Journal on Computing*, 13:461–487, 1984.
9. H. Buhrman and L. Torenvliet. On the cutting edge of relativization: The resource-bounded injury method. In *Proceedings of 21st International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science. Springer-Verlag, 1994.
10. H. Buhrman, L. Torenvliet, and P. van Emde Boas. Twenty Questions to a P-selector. *Information Processing Letters*, 48(4), 1994.
11. H. Buhrman, P. van Helden, and L. Torenvliet. Selective self-reducible sets: A new characterization of P. In *Proceedings of 8th Annual Conference on Structure in Complexity Theory*, pages 45–51, 1993.
12. J. Cai, A. Naik, and A. Selman. A note on p-selective sets and on adaptive versus nonadaptive queries to NP. Technical Report 94-02, State University of New York at Buffalo, 1994.
13. D. Denny-Brown, Y. Han, L. Hemaspaandra, and L. Torenvliet. Semi-membership algorithms: Some recent advances. *SIGACT News*, 25(3):12–23, September 1994.
14. S. Even, A. Selman, and Y. Yacobi. The complexity of promise problems with applications to public-key cryptography. *Information And Control*, 61(2):159–173, May 1984.
15. J. Gill. Computational complexity of probabilistic Turing machines. *SIAM Journal of Computing*, 6(4):675–695, Dec. 1977.
16. H. Heller. *Relativized polynomial hierarchies extending two levels.* PhD thesis,

Universität München, 1981.

17. J. Hartmanis, N. Immerman, and V. Sewelson. Sparse sets in NP-P: EXPTIME versus NEXPTIME. *Information and Control*, 65:158–181, 1985.

18. E. Hemaspaandra, A. Naik, M Ogiwara, and A. Selman. P-selective sets, and reducing search to decision versus self-reducibility. Technical Report 93-21, SUNY at Buffalo, Buffalo, NY 14260, 1993. To appear in JCSS.

19. L. Hemaspaandra, A. Naik, M. Ogiwara, and A. Selman. Computing unique solutions collapses the polynomial hierarchy. In D.Z. Du and X. S. Zhang, editors, *Proccedings of 5th International Symposium on Algorithms and Computation*, Lecture Notes in Computer Science, pages 56–64. Springer Verlag, 1994. To appear in SIAM J. of Comput.

20. C. Jockusch, Jr. Semirecursive sets and positive reducibility. *Trans. of the Amer. Math. Soc.*, 131(2):420–436, 1968.

21. J. Kämper. Non-uniform proof systems: a new framework to describe non-uniform and probabilistic complexity classes. *Theoretical Computer Science*, 85(2):305–331, 1991.

22. R. Karp and R. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of 12th ACM Symposium on Theory of Computing*, pages 302–309, 1980. An extended version has also appeared as: Turing machines that take advice, *L'Enseignement Mathématique*, 2nd series 28, 1982, pages 191–209.

23. K. Ko. On self-reducibility and weak P-selectivity. *Journal of Computer and System Sciences*, 26:209–211, 1983.

24. J. Köbler. Locating P/poly optimally in the extended low hierarchy. In *Proceedings of the 10th Annual Symposium on Theoretical Aspects of Computer Science*, pages 28–37. Springer-Verlag, 1993.

25. J. Köbler. On the structure of low sets. In *Proceedings of 10th Annual IEEE Structure in Complexity Theory Conference*, pages 246–261, 1995.

26. J. Köbler and O. Watanabe. New collapse consequences of NP having small circuits. Technical Report 94-11, Universität Ulm, November 1994. To appear in ICALP 95.

27. R. Ladner, N. Lynch, and A. Selman. A comparison of polynomial time reducibilities. *Theoretical Computer Science*, 1:103–123, 1975.

28. T. Long. Strong nondeterministic polynomial-time reducibilities. *Theoretical Computer Science*, 21:1–25, 1982.

29. L. Longpré and A. Selman. Hard promise problems and nonuniform complexity. *Theoretical Computer Science*, 115(3):277–290, 1993.

30. S. Mocas. Using Bounded time classes to separate classes in the Exponential Time Hierarchy from the classes in PH. In *Proceedings of 9th Annual Conference on Structure in Complexity Theory*, pages 53–58, 1994.

31. A. Meyer and M. Paterson. With what frequency are apparently intractable problems difficult? Technical Report MIT/LCS/TM-126, M.I.T., 1979.

32. M. Ogihara. Polynomial-time membership comparable sets. In *Proceedings of 9th Annual IEEE Conference on Structure in Complexity Theory*, pages 2–11, 1994.

33. U. Schöning. A low and a high hierarchy within NP. *Journal of Computer and System Sciences*, 27:14–28, 1983.

34. A. Selman. P-selective sets, tally languages, and the behavior of polynomial time reducibilities on NP. *Mathematical Systems Theory*, 13:55–65, 1979.

35. A. Selman. Reductions on NP and P-selective sets. *Theoretical Computer Science*,

19:287–304, 1982.

36. A. Selman. Promise problems complete for complexity classes. *Information and Computation*, 78:87–98, 1988.

37. A. Selman. A taxonomy of complexity classes of functions. *Journal of Computer and System Sciences*, 48(2):357–381, 1994.

38. L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3:1–22, 1976.

39. S. Toda. On polynomial-time truth-table reducibilities of intractable sets to P-selective sets. *Mathematical Systems Theory*, 24:69–82, 1991.

40. T. Thierauf, S. Toda, and O. Watanabe. On sets bounded truth-table reducible to p-selective sets. In *Proceedings of 11th Annual Symposium on Theoretical Aspects of Computer Science*, pages 427–438, 1994.

41. J. Wang. Selectivity and Self-reducibility: New Characterizations of Complexity Classes. *Journal of Information and Computation*, 1(1):371–384, 1994. Special Issue: Proceedings of 6th Annual International Conference on Computing and Information, May 1994, Canada. To appear in *Information Processing Letters*.

42. C. Wrathall. Complete sets and the polynomial hierarchy. *Theoretical Computer Science*, 3:23–33, 1976.