The Complexity of Finding Top-Toda-Equivalence-Class Members*

Lane A. Hemaspaandra,[1] Mitsunori Ogihara,[1] Mohammed J. Zaki,[2] and Marius Zimand[3]

[1] Department of Computer Science, University of Rochester, Rochester, NY 14627
[2] Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180
[3] Department of Computer and Information Sciences, Towson Univ., Baltimore, MD 21252

Abstract. We identify two properties that for P-selective sets are effectively computable. Namely we show that, for any P-selective set, finding a string that is in a given length's top Toda equivalence class (very informally put, a string from Σ^n that the set's P-selector function declares to be most likely to belong to the set) is $\mathrm{FP}^{\Sigma_2^p}$ computable, and we show that each P-selective set contains a weakly-$\mathrm{P}^{\Sigma_2^p}$-rankable subset.

1 Introduction

P-selectivity is a generalization of P. A set A is in P if there is a polynomial-time algorithm which, given any string x, determines whether x belongs to A. In contrast, a set A is P-selective if there is a polynomial-time algorithm (called a P-selector function) that, given any two strings x and y, outputs one of those strings, and such that the algorithm has the property that if at least one of x or y is in A, then the one the algorithm outputs belongs to A. Informally, it always places a bet on one of them being in the set, and it wins whenever winning such a bet is possible.

The book [18] provides a recent overview of the state of research regarding P-selectivity theory (see also the somewhat older article [2]). Nickelsen's thesis [24] and the recent survey article by Nickelsen and Tantau [25] are also very good starting points regarding the study of partial information classes such as the P-selective sets.

A key notion used in P-selectivity theory is the notion of a Toda equivalence class which, very loosely put, is a strongly connected component of the graph induced by a given P-selector function on the strings of a given length. This paper studies the complexity of finding a string from a given P-selector function's top Toda equivalence class—that is, a string from the unique strongly connected component that can be reached from no other strongly connected component.

P-selectivity theory has many features making it an interesting complexity-related research area. P-selectivity represents a natural generalization of feasible decidability, it has a well-studied analog in computability theory, namely, the semi-recursive sets [21]; the exploration of P-selectivity had a strong, unexpected impact on the study of NP

---

* Email: {lane,ogihara}@cs.rochester.edu, zaki@cs.rpi.edu, mzimand@towson.edu. Supported in part by NSF grants INT-9815095/DAAD-315-PPP-gü-ab, EIA-0080124, DUE-9980943, and EIA-0205061, and NIH grant P30-AG18254.

functions, solutions and ambiguity, namely, the nondeterministic version of selectivity is the central tool used to show that SAT has unique solutions only if the polynomial hierarchy collapses [12]; and, relatedly, selectivity has proven central in understanding more generally whether one can reduce the number of solutions of NP functions [12, 26,23,15].

Informally, P-selectivity captures the notion of sets for which there is a polynomial-time algorithm f telling which of any two given elements is "logically no less likely to be in the set" (see Definition 2.1). Such sets are called P-selective. P-selective sets can be arbitrarily complex: For every tally set A, there is a P-selective set that is Turing-equivalent to A [27,28]. In particular, some P-selective sets are uncomputable. Despite this, in the present paper we identify natural tasks that are computable for all P-selective sets. Indeed, these natural tasks are even computable within relatively low levels of the polynomial hierarchy.

The first such task is to produce, for an arbitrary P-selective set A (which, w.l.o.g., has at least one commutative P-selector function), at each length n, a string that is "most likely" to be in A. Let us explain a bit more what we mean by this. Each commutative polynomial-time P-selector function f for A will implicitly specify a structure of equivalence classes of strings (at a given length) that are equally likely (according to f) to be in A, and these classes can be ordered with respect to the order that one might informally call *no-less-likely-to-be-in-A* (we will explain how to define such equivalence classes, which themselves depend on only the P-selector function and not on A, in rigorous detail in Section 2 after introducing the tournament-graph model that is useful in their definition; in brief, two strings of the same length are said to be equivalent exactly if there are chains of applications of the P-selector function leading from each to the other). We will call those classes Toda equivalence classes, in light of Toda [33], and the related order will be called a Toda order. Informally put, a Toda equivalence class ζ, with respect to commutative P-selector function f, of length n strings has the property that for *every* set A for which f is a P-selector, either all the strings in ζ are in A or none of the strings in ζ are in A. And (restricting ourselves as we will globally do to just looking at strings all of the same length) each Toda equivalence class is a maximal set of strings for which this can be said.

The Toda-class approach's ordering implications play a central role in a wide range of results, ranging for example from the study of whether P-selective sets can be hard for standard complexity classes [33] to the study of associative P-selector functions [9, 10]. In this paper, we seek to better understand the Toda classes' own complexity. We show that finding an element in the top Toda class of a length (very informally and intuitively—and not quite correctly—put, a string that among the strings of that length is "most likely" to be in A) can be done with an $\mathrm{FP}^{\Sigma_2^p}$ computation.

The second task we study is that of weak-P-rankability. A function f *weakly ranks* a set A if, for any string x that is in A, $f(x)$ returns the rank of x in A; in other words, it says how many strings lexicographically less than or equal to x are in A. A set A is weakly-P-rankable if it has a function f that is computable in polynomial time and that weakly ranks A. The relation between P-selectivity and weak-P-rankability has been studied extensively by Hemaspaandra, Zaki, and Zimand in a paper [19] that focused on polynomial-time semi-rankable sets, i.e., sets that are simultaneously P-selective and weakly-P-rankable. It is shown there that there are P-selective sets that are not weakly-P-rankable. In partial contrast, in the present paper we show that any infinite P-selective set has an infinite subset that is weakly rankable by an $\mathrm{FP}^{\Sigma_2^p}$ function. We also obtain a result about the relationship between P-selectivity and weak-P-rankability. All the P-selective sets that have been considered in the literature are either standard left cuts or are structurally similar to a standard left cut (more precisely $\leq_m^p$-reducible to a standard left cut). We show that if a standard left cut is weakly-P-rankable, then it is in P. Regarding this section (Section 3), we particularly commend to the reader's attention the proof of Lemma 3.3 (omitted here; see the full version), which we feel to be a novel technique, and which is given prompt application in yielding the theorems that are stated immediately after it.

For space reasons, all proofs and much discussion/explanation is omitted here; please see the full version [16].

2 Instantiating the Top Toda Class

Let $\mathbb{N} = \{0, 1, 2, \ldots\}$ and let $\mathbb{N}^+ = \{1, 2, 3, \ldots\}$. Our alphabet will be $\Sigma = \{0, 1\}$. For any set A, $||A||$ denotes the cardinality of A. For any string $x \in \Sigma^*$, $|x|$ denotes the length of x. For any set A and any nonnegative integer n, $A^{=n}$ denotes $\{x \mid x \in A \wedge |x| = n\}$. For any set A and any string x, $A^{\leq x}$ denotes $\{y \mid y \in A \wedge y \leq_{lex} x\}$, where $\leq_{lex}$ denotes $\leq$ with respect to the standard lexicographical ordering. For the definitions of standard complexity classes such as P, NP, Σ_k^p, etc., we refer the reader to, for example, the handbook [14]. As is standard, FP denotes the class of all (total, single-valued) polynomial-time computable functions, $\Sigma_2^p = \mathrm{NP}^{\mathrm{NP}}$, $\Sigma_3^p = \mathrm{NP}^{\mathrm{NP}^{\mathrm{NP}}}$, $\mathrm{E} = \bigcup_{k \geq 0} \mathrm{DTIME}[2^{kn}]$, and $\mathrm{NE} = \bigcup_{k \geq 0} \mathrm{NTIME}[2^{kn}]$. Given classes $\mathcal{C}$ and $\mathcal{D}$, as is standard we say that $\mathcal{C}$ *is* $\mathcal{D}$*-immune* (equivalently, $\mathcal{C}$ *is immune to* $\mathcal{D}$) if there is an infinite set $A \in \mathcal{C}$ such that no infinite subset of A is a member of $\mathcal{D}$.

Definition 2.1. *[27,29] A set A is* P-selective *if there is a (total, single-valued) polynomial-time computable function f such that, for every x and y, it holds that*

1. *$f(x, y) = x$ or $f(x, y) = y$, and*
2. *$\{x, y\} \bigcap A \neq \emptyset \Rightarrow [(x \in A$ and $f(x, y) = x)$ or $(y \in A$ and $f(x, y) = y)]$.*

We use P-sel *to denote the class of all sets that are* P*-selective.*

The function f appearing in Definition 2.1 is called a *P-selector* or a *P-selector function*. A P-selector (function) f is *commutative* if it has the property that for all x and y in Σ^*, $f(x, y) = f(y, x)$. Each P-selective set A has a commutative P-selector function, because we can replace an arbitrary P-selector f for A with $f'(x, y) = f(\min(x, y), \max(x, y))$. It is easy to see that f' is a commutative P-selector for A. Since all P-selective sets have commutative P-selector functions, it is very common in the literature to focus on commutative P-selector functions, and we do so here.

A tournament graph $G = (V_G, E_G)$ is a complete oriented graph, i.e., a directed graph having the property that, for every two (possibly equal) nodes a and b, $||\{(a, b), (b, a)\} \cap E_G|| = 1$. A commutative P-selector induces a tournament graph

G_f on Σ^*. The nodes are the strings in Σ^* and there is an edge (x, y) (also denoted $x \geq_f y$) exactly if $f(x, y) = x$. When speaking in plain text, we will use synonymously with this the terms x *beats* y, x *wins at* y, and y *loses to* x. (Note that each x both wins at itself and loses to itself, and our tournament graphs have self-loops at each node.) We say $x >_f y$ if $x \geq_f y$ and $x \neq y$.

Let $G_{f,n}$ be the subgraph of G_f induced by the nodes in Σ^n. Two nodes $x, y \in \Sigma^n$ are *Toda-equivalent*, notated $x \equiv_{Toda} y$, if $G_{f,n}$ contains a path from x to y and a path from y to x. The relation $\equiv_{Toda}$ is an equivalence relation. We will denote the equivalence class of $x \in \Sigma^n$ by $[x]_f$. For strings x and y of the same length, we order their equivalence classes as follows: $[x]_f \geq [y]_f$ holds exactly if $x \geq_f y$. For strings x and y of the same length, we say that $[x]_f > [y]_f$ exactly if $[x]_f \geq [y]_f$ and $[x]_f \neq [y]_f$.

(It is easy to see that these relations are consistent when applied only among strings all of the same length. Note that we neither define nor ever use $[w]_f > [z]_f$ or $[w]_f \geq [z]_f$ for the case where $|w| \neq |z|$. Our focus will always be on collections of same-length strings. However, if one wanted to compare different-length strings, for the purposes of this paper let us say that equivalence classes of different lengths are always, by definition, incomparable, and so viewed as being over all of Σ^* our classes form a partial rather than a total order. Since our focus is within a length and there our order is never undefined, we will for simplicity simply use the term "order.")

Of course, $[y]_f < [x]_f$ means the same as $[x]_f > [y]_f$, and $[y]_f \leq [x]_f$ means the same as $[x]_f \geq [y]_f$.

The following fact holds.

Fact 2.2.

1. *If A is a* P*-selective set having commutative function f as a* P*-selector then, for all $x \in \Sigma^n$, $A \cap [x]_f = [x]_f$ or $A \cap [x]_f = \emptyset$.*
2. *Let A be a* P*-selective set having commutative function f as a* P*-selector. If $A \cap [x]_f = [x]_f$ then for all y such that $|y| = |x|$ and $[y]_f \geq [x]_f$, it holds that $A \cap [y]_f = [y]_f$. (In fact, it even holds that if $A \cap [x]_f = [x]_f$ then for* all *y (regardless of $|y|$) such that $f(x, y) = y$ it holds that $A \cap [y]_f = [y]_f$.)*

These equivalence classes (related to strings of length n) form a partition of Σ^n. In particular, there are strings $x_1^n, x_2^n, \ldots, x_k^n \in \Sigma^n$ such that the equivalence classes $[x_1^n]_f, \ldots, [x_k^n]_f$ form a partition of Σ^n and $[x_1^n]_f > [x_2^n]_f > \ldots > [x_k^n]_f$. The set $[x_1^n]_f$ is called the *top Toda class* (at length n with respect to f).

Definition 2.3. *Let f be a commutative* P*-selector function. A function g is a* BestAtLength function (for f) *if, for each n, $g(1^n) \in [x_1^n]_f$, i.e., it outputs an element of the top Toda class at length n with respect to f.*

Each string that belongs to the top Toda class (at length n with respect to f) will be called a *top Toda element* (at length n with respect to f).

Note that Definition 2.3 does not mention what set f is a P-selector for, since each BestAtLength function will work equally well for each of the potentially uncountably many sets for which f is a P-selector.

Theorem 2.4. *Every commutative* P*-selector f has a BestAtLength function computable in* $\mathrm{FP}^{\Sigma_2^p}$.

It is natural to ask if there is a more efficient BestAtLength function. The next result shows that it is unlikely that the BestAtLength function of every commutative P-selector is in FP because this would imply $\mathrm{E} = \mathrm{NE}$. The proof is relativizable, so it follows that there is an oracle relative to which the BestAtLength function for some commutative P-selector is not in FP. The issue of whether the statement of Theorem 2.4 can be improved to $\mathrm{FP}^{\mathrm{NP}}$ remains open.

Theorem 2.5. *1. If* $\mathrm{P} = \mathrm{NP}$*, then every commutative* P*-selector* f *has a BestAtLength function in* FP.

2. *If every commutative* P*-selector* f *has a BestAtLength function in* FP *then* $\mathrm{E} = \mathrm{NE}$.

Since there are relativized worlds in which E and NE differ, Theorem 2.5 (in its relativized version, which also holds) yields the following corollary.

Corollary 2.6. *There is an oracle relative to which there is a commutative* P*-selector* f *such that no BestAtLength function for* f *is in* FP.

Theorem 2.5 relativizes. That is, for each set B it holds that: If every commutative P^B-selector f has a BestAtLength function in FP^B then $\mathrm{E}^B = \mathrm{NE}^B$. Though neither $\mathrm{NP} \cap \mathrm{coNP}$ nor $\mathrm{coNE} \cap \mathrm{NE}$ is currently known to have complete sets (see [31,4,5] regarding $\mathrm{NP} \cap \mathrm{coNP}$), one nonetheless can (by a "set by set" argument—[11, Corollary 6] for example employs a set-by-set argument in the completely different setting of Karp–Lipton-type results), in light of the facts that $\mathrm{E}^{\mathrm{NP}\cap\mathrm{coNP}} = \mathrm{coNE} \cap \mathrm{NE}$ and $\mathrm{NE} = \mathrm{coNE} \cap \mathrm{NE} \iff \mathrm{NE} = \mathrm{coNE}$, as an application of this see that the following holds.

Corollary 2.7. *If every commutative* $\mathrm{FP}^{\mathrm{NP}\cap\mathrm{coNP}}$*-selector* f *has a BestAtLength function in* $\mathrm{FP}^{\mathrm{NP}\cap\mathrm{coNP}}$ *then* $\mathrm{coNE} = \mathrm{NE}$.

The reader may naturally wonder whether the $\mathrm{E} = \mathrm{NE}$ conclusion of part 2 of Theorem 2.5 can be strengthened to a $\mathrm{P} = \mathrm{NP}$ conclusion. We do not have a definitive answer; the fact that BestAtLength functions have tally inputs seems a difficult impediment to proving this. We do note that it is the only impediment. That is, if we make a new notion of BestAtLength function, let us call it BestBelowUsAtLength, that (for a fixed commutative P-selector function f) takes as its input $\langle 1^n, z\rangle$ and (i) when $|z| \neq n$ outputs "illegal z"; and (ii) when $|z| = n$ and $[z]_f$ is not in the bottom Toda equivalence class (with respect to f at length n) outputs a string w, $|w| = n$, such that $[w]_f$ is the topmost Toda equivalence class (with respect to f at length n) that is less than $[z]_f$ (i.e., such that $[z]_f > [w]_f$ yet, for each $v \in \Sigma^n$, it holds that $[w]_f < [v]_f \Rightarrow [v]_f \geq [z]_f$). (When $|z| = n$ and $[z]_f$ is in the bottom Toda equivalence class (with respect to f at length n), a BestBelowUsAtLength function can output whatever lie it likes.) Under this definition, it is easy to see that a $\mathrm{P} = \mathrm{NP}$ conclusion holds.

Proposition 2.8. *If every commutative* P*-selector* f *has a BestBelowUsAtLength function in* FP *then* $\mathrm{P} = \mathrm{NP}$.

An element x in the top Toda class at length n has the property that for every string $y \in \Sigma^n$ there is a path from x to y in the induced tournament graph. We next investigate a related notion. It is known (this was noted as early as the 1950s [22]) that in a tournament graph there is at least one node v such that for every other node w there is a path of length at most 2 from v to w. This property has played an important role in improving from quasilinear to linear the amount of nondeterminism used to accept the P-selective sets with optimal nonuniform advice ([13], see also [17]) and in understanding the complexity of the reachability problem in tournaments [32]. A node v with the above property is called a *king*. A function g is a Find-a-King function for a commutative P-selector f if for all n, on input 1^n, f outputs a king of the graph $G_{f,n}$.

Proposition 2.9. *Every commutative* P*-selector function* f *has a Find-A-King function computable in* $\mathrm{FP}^{\Sigma_3^p}$.

It is interesting to note that building a top Toda element seems to be easier than building a king (as indicated by the previous theorems), yet recognizing a top Toda element seems (given our current stage of knowledge) to be a more difficult task than recognizing a king. This holds because a string $x \in \Sigma^n$ is a top Toda element at length n exactly if for any string $y \in \Sigma^n$ there is a path from x to y; checking this condition is in PSPACE. One can observe that this problem is also in the advice class $\mathrm{PP}/$ linear (that is it can be done in PP given an advice string of size $O(n)$). Indeed, for an arbitrary P-selector f, let us consider the Toda-equivalence classes for the strings in Σ^n sorted according to the order relation defined above: $[x_1]_f > [x_2]_f > \ldots > [x_k]_f$. Then any element x in $[x_1]_f$ beats at least the elements in $\{x\} \cup (\bigcup_{2\leq i\leq k}[x_i]_f)$ and thus has outdegree at least $1 + \sum_{2\leq i\leq k}||[x_i]_f||$. Also note that any element (of Σ^n) that is not in $[x_1]_f$ cannot beat any element in $[x_1]_f$ and thus its outdegree is at most $\sum_{2\leq i\leq k}||[x_i]_f||$. Thus if we use as the advice string the binary encoding of $1 + \sum_{2\leq i\leq k}||[x_i]_f||$ and if we check versus a threshold the outdegree of a node via a PP computation, we can check whether a string is in the top Toda class.

On the other hand, a string x in $G_{n,f}$ is a king node if and only if $(\forall y_1 : |y_1| = |x|)(\exists y_2 : |y_2| = |x|)[(x \text{ beats } y_1) \vee ((x \text{ beats } y_2) \wedge (y_2 \text{ beats } y_1))]$. Thus checking whether a string x is a king in $G_{n,f}$ can be done with a Π_2^p computation. We note that this fact, and also Proposition 2.9, could be shown indirectly using Tantau's recent work on the complexity of succinct tournament reachability [32].

Though all P-selective sets are known to be in the class $\mathrm{NP}/$ linear [17], which was an improvement from even earlier work [8] that implicitly placed them in $\mathrm{PP}/$ linear, it is an open question whether the $\mathrm{PP}/$ linear result for top Toda element recognition given above can itself be strengthened to $\mathrm{NP}/$ linear. We conjecture that it cannot. Also, we note in passing that in the profoundly different model in which the tournament—far from being uniformly specified via a P-selector function—can be explored only via queries to a black box, and our input additionally includes the set of nodes inducing via that black box a tournament over which a king (or a certain sequence of kings) is sought, bounds on the necessary and sufficient numbers of queries to find such have been studied in, for example, [30].

3 P-Selectivity and Ranking

While P-selectivity is an extension of polynomial-time decidability, the notion of polynomial-time weak rankability goes in the opposite direction. It describes sets that are so simple that there exists a polynomial-time algorithm that, on inputs that are elements in the set, outputs the number of elements in the set up to that element. Weak-P-rankability can be an attribute of extremely complex sets. One further generalization is to allow the ranking functions to belong to some broader family of functions (e.g., having some complexity bound less stringent than polynomial time). There have been many papers studying the issue of which sets can be ranked [3,7,1,20].

Definition 3.1. *For any set B and any string x, define $rank_B(x) = ||B^{\leq x}||$.*

1. *[7] A set A is* strongly-P-rankable *if there is a polynomial-time computable function f such that $(\forall x \in \Sigma^*)\,[f(x) = rank_A(x)]$. We also use* strongly-P-rankable *to denote the class of all sets that are strongly-*P*-rankable.*
2. *[3] A set A is* P-rankable *if there is a polynomial-time computable function f such that (a) $(\forall x \in A)\,[f(x) = rank_A(x)]$ and (b) $(\forall x \notin A)\,[f(x) =$ "not in A"$]$. We also use* P-rankable *to denote the class of all sets that are* P*-rankable.*
3. *[7] A set A is* weakly-P-rankable *if there is a polynomial-time computable function f such that $(\forall x \in A)\,[f(x) = rank_A(x)]$. We also use* weakly-P-rankable *to denote the class of all sets that are weakly-*P*-rankable.*
4. *Let $\mathcal{F}$ be a family of functions mapping strings into natural numbers. A set A is* weakly-$\mathcal{F}$-rankable *if there is a function $f \in \mathcal{F}$ such that $(\forall x \in A)\,[f(x) = rank_A(x)]$. We also use* weakly-$\mathcal{F}$-rankable *to denote the class of all sets that are weakly-$\mathcal{F}$-rankable.*

Note that, immediately from the definitions, strongly-P-rankable $\subseteq$ P-rankable $\subseteq$ weakly-P-rankable. (The first inclusion is easy to see in light of the fact that, for each $x \neq \epsilon$, $x \in A \iff rank_A(x) > rank_A(x-1)$, where $x - 1$ denotes the immediate lexicographic predecessor of x.) Also note that for $x \notin A$, the definition of weakly-P-rankable sets puts no constraint on the behavior of f on input x other than that f must run in polynomial time. This is a point of similarity with P-selectivity useful for the following refinement of P-selectivity, which has been introduced in [19]. The refinement adds the requirement that when at least one of the inputs belongs to the set, we output not merely an input that is in the set, but also output its correct ranking information.

Definition 3.2. *[19] A set A is* polynomial-time semi-rankable *if there is a (total, single-valued) function f such that, for every x and y,*

1. *$(\exists n)\,[f(x,y) = \langle x, n\rangle$ or $f(x,y) = \langle y, n\rangle]$, and*
2. *$\{x,y\} \bigcap A \neq \emptyset \Rightarrow [(x \in A$ and $f(x,y) = \langle x, rank_A(x)\rangle)$ or $(y \in A$ and $f(x,y) = \langle y, rank_A(y)\rangle)]$.*

In such a case, we say that f is a semi-ranking function for A. We use P-sr *to denote the class of sets that are polynomial-time semi-rankable.*

As noted in [19], P-sr $=$ P-sel $\cap$ weakly-P-rankable. Among other results, it is shown in [19] that P-sr is a proper set of P-sel, i.e., there are P-selective sets that are not weakly-P-rankable (there are also sets that are weakly-P-rankable but not P-selective).

In the full version of this paper, we provide a new and short argument showing this fact, and even that there are P-selective sets that are not weakly-rankable by any function in, for example, the arithmetical hierarchy.

It is shown in [19] that P-sr has structural properties different from those of P-sel. For example, unlike P-sel, P-sr is not closed under complementation, union with P sets, or join with P sets. It is natural to ponder whether P-sel can be separated from the class of weakly-P-rankable sets or from P-sr in a stronger sense, namely with immunity. We note first that the above approach is useless because $L(0) = \emptyset$ and any standard left cut $L(\gamma)$, $0 < \gamma < 1$, contains the subset $\{0^j \mid j \in \mathbb{N}\}$, which belongs to P-sr. Perhaps somewhat surprisingly it turns out, as we will show as Theorem 3.5, that the statement "P-sel is weakly-P-rankable immune" implies $\mathrm{P} \neq \Sigma_2^p$ (equivalently, $\mathrm{P} \neq \mathrm{NP}$) and thus it seems beyond reach at this time.

A set S if P-printable [6] if there exists a polynomial-time algorithm such that, for each $n \in \mathbb{N}$, on input 0^n the algorithm outputs exactly the members of S having length at most n. Note that every P-printable set is sparse and belongs to P. The above definition can be relativized in the standard way.

Lemma 3.3. *Let Q be any set. If A is* P*-selective, S is* P^Q*-printable, and $A \cap S$ is infinite, then $A \cap S$ (and thus also S, and most particularly also A) has an infinite weakly-*FP^Q*-rankable subset.*

The following results immediately follow from Lemma 3.3.

Theorem 3.4. P-sel *is not bi-immune to the class of weakly-*P*-rankable sets.*

Theorem 3.5. *Any infinite* P*-selective set A has an infinite weakly-*$\mathrm{FP}^{\Sigma_2^p}$*-rankable subset.*

Corollary 3.6. *If* $\mathrm{P} = \mathrm{NP}$*, then* P-sel *is not immune to the class of weakly-*P*-rankable sets.*

The reader may wish to compare Theorem 3.5 with the following result—neither of which seems to imply the other—from [9] (see also [10]) regarding printability [6]: Each infinite P-selective set B has an infinite $\mathrm{FP}^{B \oplus \Sigma_2^p}$-printable subset.

An important subclass of P-sel is the class of sets that are standard left cuts, a class that we have already used. Recall that for each real $0 \leq \gamma < 1$ the standard left cut of γ is the set $L(\gamma) = \{\beta_1\beta_2 \cdots \beta_z \mid z \in \mathbb{N} \wedge (\forall j : 1 \leq j \leq z)[\beta_j \in \{0,1\}] \wedge \sum_{1 \leq i \leq z} \frac{\beta_i}{2^i} < \gamma\}$. All the P-selective sets that have been constructed in the literature are either standard left cuts or are $\leq_m^p$-equivalent to a standard left cut and, in fact, proving that there is a P-selective set that is not $\leq_m^p$-equivalent to a standard left cut is known to be as hard as showing $\mathrm{P} \neq \mathrm{PP}$ [8]. In contrast, we observe that standard left cuts that are weakly-rankable are always in P.

Theorem 3.7. *If A is a standard left cut, then A is weakly-*P*-rankable if and only if A is strongly-*P*-rankable.*

Since strongly-P-rankable $\subseteq$ P-rankable $=$ $\mathrm{P} \cap$ weakly-P-rankable, we have the following immediate corollary.

Corollary 3.8. *Each weakly-*P*-rankable standard left cut belongs to* P.

Acknowledgments: We are grateful to A. Kaplan and B. Serog for helpful conversations, and to an anonymous LATIN '04 referee for helpful comments.

References

1. A. Bertoni, M. Goldwurm, and N. Sabadini. The complexity of computing the number of strings of given length in context-free languages. *Theoretical Computer Science*, 86(2):325–342, 1991.
2. D. Denny-Brown, Y. Han, L. Hemaspaandra, and L. Torenvliet. Semi-membership algorithms: Some recent advances. *SIGACT News*, 25(3):12–23, 1994.
3. A. Goldberg and M. Sipser. Compression and ranking. *SIAM Journal on Computing*, 20(3):524–536, 1991.
4. Y. Gurevich. Algebras of feasible functions. In *Proceedings of the 24th IEEE Symposium on Foundations of Computer Science*, pages 210–214. IEEE Computer Society Press, Nov. 1983.
5. J. Hartmanis and N. Immerman. On complete problems for $\mathrm{NP} \bigcap \mathrm{coNP}$. In *Proceedings of the 12th International Colloquium on Automata, Languages, and Programming*, pages 250–259. Springer-Verlag *Lecture Notes in Computer Science #194*, July 1985.
6. J. Hartmanis and Y. Yesha. Computation times of NP sets of different densities. *Theoretical Computer Science*, 34(1–2):17–32, 1984.
7. L. Hemachandra and S. Rudich. On the complexity of ranking. *Journal of Computer and System Sciences*, 41(2):251–271, 1990.
8. E. Hemaspaandra, A. Naik, M. Ogihara, and A. Selman. P-selective sets and reducing search to decision vs. self-reducibility. *Journal of Computer and System Sciences*, 53(2):194–209, 1996.
9. L. Hemaspaandra, H. Hempel, and A. Nickelsen. Algebraic properties for deterministic selectivity. In *Proceedings of the 4th Annual International Computing and Combinatorics Conference*, pages 49–58. Springer-Verlag *Lecture Notes in Computer Science #2108*, Aug. 2001.
10. L. Hemaspaandra, H. Hempel, and A. Nickelsen. Algebraic properties for deterministic and nondeterministic selectivity. Technical Report TR-778, Department of Computer Science, University of Rochester, Rochester, NY, May 2002. Revised, May 2003.
11. L. Hemaspaandra, A. Hoene, A. Naik, M. Ogiwara, A. Selman, T. Thierauf, and J. Wang. Nondeterministically selective sets. *International Journal of Foundations of Computer Science*, 6(4):403–416, 1995.
12. L. Hemaspaandra, A. Naik, M. Ogihara, and A. Selman. Computing solutions uniquely collapses the polynomial hierarchy. *SIAM Journal on Computing*, 25(4):697–708, 1996.
13. L. Hemaspaandra, C. Nasipak, and K. Parkins. A note on linear-nondeterminism, linear-sized, Karp–Lipton advice for the P-selective sets. *Journal of Universal Computer Science*, 4(8):670–674, 1998.
14. L. Hemaspaandra and M. Ogihara. *The Complexity Theory Companion*. Springer-Verlag, 2002.
15. L. Hemaspaandra, M. Ogihara, and G. Wechsung. Reducing the number of solutions of NP functions. *Journal of Computer and System Sciences*, 64(2):311–328, 2002.

16. L. Hemaspaandra, M. Ogihara, M. Zaki, and M. Zimand. The complexity of finding top-Toda-equivalence-class members. Technical Report TR-808, Department of Computer Science, University of Rochester, Rochester, NY, Aug. 2003.
17. L. Hemaspaandra and L. Torenvliet. Optimal advice. *Theoretical Computer Science*, 154(2):367–377, 1996.
18. L. Hemaspaandra and L. Torenvliet. *Theory of Semi-Feasible Algorithms*. Springer-Verlag, 2003.
19. L. Hemaspaandra, M. Zaki, and M. Zimand. Polynomial-time semi-rankable sets. In *Journal of Computing and Information*, 2(1), Special Issue: *Proceedings of the 8th International Conference on Computing and Information*, pages 50–67, 1996. CD-ROM ISSN 1201-8511/V2/#1.
20. D. Huynh. The complexity of ranking simple languages. *Mathematical Systems Theory*, 23(1):1–20, 1990.
21. C. Jockusch. Semirecursive sets and positive reducibility. *Transactions of the AMS*, 131(2):420–436, 1968.
22. H. Landau. On dominance relations and the structure of animal societies, III: The condition for score structure. *Bulletin of Mathematical Biophysics*, 15(2):143–148, 1953.
23. A. Naik, J. Rogers, J. Royer, and A. Selman. A hierarchy based on output multiplicity. *Theoretical Computer Science*, 207(1):131–157, 1998.
24. A. Nickelsen. Polynomial-time partial information classes. Wissenschaft und Technik Verlag, 2001. Also Ph.D. thesis, Technische Universität Berlin, Berlin, Germany, 1999.
25. A. Nickelsen and T. Tantau. Partial information classes. *SIGACT News*, 34(1), 2003.
26. M. Ogihara. Functions computable with limited access to NP. *Information Processing Letters*, 58(1):35–38, 1996.
27. A. Selman. P-selective sets, tally languages, and the behavior of polynomial time reducibilities on NP. *Mathematical Systems Theory*, 13(1):55–65, 1979.
28. A. Selman. Analogues of semirecursive sets and effective reducibilities to the study of NP complexity. *Information and Control*, 52(1):36–51, 1982.
29. A. Selman. Reductions on NP and P-selective sets. *Theoretical Computer Science*, 19(3):287–304, 1982.
30. J. Shen, L. Sheng, and J. Wu. Searching for sorted sequences in tournaments. *SIAM Journal on Computing*, 32(5):1201–1209, 2003.
31. M. Sipser. On relativization and the existence of complete sets. In *Proceedings of the 9th International Colloquium on Automata, Languages, and Programming*, pages 523–531. Springer-Verlag *Lecture Notes in Computer Science #140*, July 1982.
32. T. Tantau. A note on the complexity of the reachability problem for tournaments. Technical Report TR01-092, Electronic Colloquium on Computational Complexity, http://www.eccc.uni-trier.de/eccc/, Nov. 2001.
33. S. Toda. On polynomial-time truth-table reducibilities of intractable sets to P-selective sets. *Mathematical Systems Theory*, 24(2):69–82, 1991.