Advice for Semifeasible Sets and the Complexity-Theoretic Cost(lessness) of Algebraic Properties*

Lane A. Hemaspaandra
Department of Computer Science
University of Rochester
Rochester, NY 14627 USA

Univ. of Rochester Comp. Sci. Dept. Technical Report TR-2004-843
July 17, 2004

Abstract

This paper provides a tutorial overview of the advice complexity of the semifeasible sets—informally put, the class of sets having a polynomial-time algorithm that, given as input any two strings of which at least one belongs to the set, will choose one that does belong to the set. No previous familiarity with either the semifeasible sets or advice complexity will assumed, and when we include proofs we will try to make the material as accessible as possible via providing intuitive, informal presentations.

Karp and Lipton introduced advice complexity about a quarter of a century ago [KL80]. Advice complexity asks, for a given power of interpreter, how many bits of "help" suffice to accept a given set. Thus, this is a notion that contains aspects both of informational complexity and of computational complexity. We will see that for some powers of interpreter the (worst-case) complexity of the semifeasible sets is known right down to the bit (and beyond), but that for the most central power of interpreter—deterministic polynomial time—the complexity is currently known only to be at least linear and at most quadratic.

While overviewing the advice complexity of the semifeasible sets, we will stress also the issue of whether the functions at the core of semifeasibility—so-called selector functions—can without cost be chosen to possess such algebraic properties as commutativity and associativity. We will see that this is relevant, in ways both potential and actual, to the study of the advice complexity of the semifeasible sets.

Keywords: advice complexity, linear advice, selector functions, semifeasible computation, P-selectivity, associativity, commutativity, algebraic properties, P/linear, NP/linear.

*This paper is based on the author's invited address at the DCFS '04 conference.

1 Introduction, Definitions, and Motivations

1.1 Semifeasible Sets

In this paper we present a tutorial overview of the advice complexity of the semifeasible sets. Let us start with a definition.

Definition 1.1 1. *[Sel79] A set A is said to be* P-selective *(or* semifeasible*) exactly if there is a polynomial-time computable, total, single-valued function f such that, for every pair of strings x and y, it holds that*

$$f(x,y) \in \{x,y\} \wedge (\{x,y\} \cap A \neq \emptyset \implies f(x,y) \in A).$$

The function f is said to be a P-selector function *for A.*

2. *A common notation that we will adopt is the following:* P-sel $= \{A \mid A$ *is a P-selective set*$\}$.

Since this notion, from the work of Selman that introduced the P-selective sets [Sel79, Sel81, Sel82], will sit at the center of this paper, it is worth pausing to consider what the definition means. The definition says that a set is P-selective if there is a 2-argument polynomial-time function that (a) always chooses and outputs one of its two inputs, and (b) if at least one of its input is in the set then it chooses and outputs one that is in the set. This is typically informally described via saying that the P-selector function, given two strings, makes a claim as to which is in the set, and if it can possibly be right it is.

To make this utterly explicit, let us consider all possible cases. If both inputs are out of the set, the P-selector function can legally output either input; it can't possibly win (i.e., output an element in the set), so the definition, quite reasonably, lets it off the hook. If both inputs are in the set, the P-selector function can legally output either input; it can't possibly lose as long as it makes a choice (and by definition it must make such a choice). So, the only critical case is what the function does when exactly one of its two inputs is in the set, and here the definition has very sharp teeth: In such cases, the P-selector function must always output the one input that is in the set.

The term *semifeasible set* generally applies to the P-selective sets, and sometimes is also taken as encompassing their close cousins (e.g., the NP-selective sets) that are selective (in a rigorous sense that we will not define in this paper, but see [HHN$^+$95, HNOS96b]) via selector functions from slightly more general complexity classes. In this overview, we will focus on the P-selective sets and their advice complexity.

Though we assume no prior familiarity with the P-selective sets, an introduction to them would fill a paper, so we here simply mention some pointers, properties, and motivations. The recent book by Hemaspaandra and Torenvliet [HT03] devoted to semifeasible algorithms—and most especially to the P-selective sets—provides a far more detailed introduction to the P-selective sets and the motivations for their study, and we commend that to readers interested in a fuller introduction/motivation of P-selectivity. Anyway, here are, briefly, some of the motivations. The most obvious one is that the class P simply isn't known to contain as much as anyone would like, and so there has been intense work in complexity theory to define and study generalizations of P. The transition from P (with its poly-time membership testing) to P-sel (with its poly-time test of, in some sense, relative membership likelihood) is one such generalization, and is an exact analog of the transition from the recursive sets (with their recursive membership testing) to Jockusch's [Joc68] "semirecursive sets" (with their recursive test of, in some sense, relative membership likelihood). Another motivation of the P-selective sets is that the P-selective sets in some sense capture the notion of being able to determine the better of two options—or at least, to choose one such that if either option is "acceptable" then the one that one has chosen will be acceptable. Note that in a search, if one faces two options, one often would be very happy to have a way of choosing the better one (even if in making such a choice one did not instantly know whether it was itself good enough... just that if it was, one would (ideally) eventually get to another choice, and another, and that by making the better choice at each point one could get to a "good" goal). Thus, very informally, selectivity can be motivated as trying to capture some notion of decent-quality decision-making regarding preference. Indeed, the fact that in the right framework P-selectivity can drive a search via pairwise choices, and the ability to put the problem into that framework and then ensure that the sequence of choices will end after at most a polynomial amount of work, is exactly the substance of Selman's classic proof that, unless P=NP, no NP-complete set can be P-selective [Sel79]. A third motivation for the P-selective sets is that they capture some nice objects. In particular, also from the seminal work of Selman ([Sel81], see also [Sel79, Ko82, Ko83]), we have that with respect to the standard formalization-as-a-set of the notion of the standard left cut of a real number, it turns out that for every real number (on the interval $[0,1)$) the left cut of the number is a P-selective set. This is not as deep as it sounds; it is just a fancy way of saying that if you want to choose one of two dyadic rationals that is going to be less than a certain number, choosing the smaller rational is never going to steer you wrong (unless both numbers were too large in the first place, in which case you had no good option from the get-go). A final motivation we mention for the study of semifeasible sets is that their study has yielded returns in unexpected directions. In particular, the study of (nondeterministic) selectivity has shown that NP lacks unique solutions unless the polynomial hierarchy collapses [HNOS96b]. We will briefly return to that issue in Section 3.

1.2 Advice Complexity

Since we will be overviewing the advice complexity of the semifeasible sets, we need to rigorously state the notion of advice, which is due to Karp and Lipton [KL80]. Let $\Sigma = \{0, 1\}$.

Definition 1.2 *[KL80]*

1. *Let $\mathcal{C}$ be any subset of 2^{Σ^*}, i.e., any collection of sets over the alphabet Σ. Let f be a function from $\mathbb{N}$ to $\mathbb{N}$. Then $\mathcal{C}/f(n)$ is defined to be the set $\{A \mid (\exists B \in \mathcal{C})\,(\exists g : \mathbb{N} \rightarrow \{0,1\}^*)\,[(\forall n)\,[f(n) = |g(n)|]$ and $(\forall x \in \{0,1\}^*)\,[x \in A \iff \langle x,\, g(|x|)\rangle \in B]\}$.*
2. *For any collection, $\mathcal{F}$, of functions from $\mathbb{N}$ to $\mathbb{N}$, we define $\mathcal{C}/\mathcal{F}$ to be $\{A \mid (\exists f \in \mathcal{F})\,[A \in \mathcal{C}/f]\}$.*

As is common, we will use the notations poly, quadratic, and linear to denote the class of all polynomially bounded, quadratically bounded, and linearly bounded functions from $\mathbb{N}$ to $\mathbb{N}$. (The bounds are in terms of values, not lengths. For example, the function $f(n) = 7n^2 + 2$ belongs to poly and quadratic but not to linear.) Among the classes of interest to us will be two subclasses of P/poly, namely P/quadratic and P/linear. The output of the function g above is often referred to as the "advice bits" or "advice string," and g is often referred to as the "advice function." The class P/poly is often spoken of as the class of sets having small circuits (since that is another equivalent characterization of this class). Meyer (see [BH77]) noted the nice alternative characterization

$$\mathrm{P/poly} = \{A \mid (\exists \text{ sparse } S)\,[A \in \mathrm{P}^S]\},$$

i.e., we have $\mathrm{P/poly} = \mathrm{P}^{\mathrm{SPARSE}}$ ($= \mathrm{P}^{\mathrm{TALLY}}$, too).

The above (classic, but) rather spooky definition, part 1 of Definition 1.2, is really quite innocuous and natural once one looks it over a bit. All it captures is the notion of how many bits one needs at each length for an interpreter from class $\mathcal{C}$ to be able to correctly determine the membership of each string from a given set.

Let us give a very simple example. Consider the following tally version of the halting problem (HP):

$$L_{halt} = \{1^{1 \cdot x} \mid x \in \mathrm{HP}\},$$

where $1 \cdot x$ denotes the integer that the string that one gets by concatenating the bits of x onto a 1 represents when viewed as a binary integer. Note that L_{halt} is not even a recursive set. Nonetheless, it clearly belongs to the class P/1. Namely, the one advice bit per length reveals whether that length's tally string belongs to L_{halt}, and the P set, given that advice bit and the input being asked about does this: If the input string is not of the form 1^* reject and otherwise accept exactly if that is what the advice bit says to do. Not all advice containments are this immediate. For example, via an amplification argument, all sets in one- and two-sided bounded-error probabilistic polynomial time (R and BPP) are known to belong to P/poly [Adl78,Sch86].

1.3 Algebraic Properties

Consider (total) functions mapping from $\Sigma^* \times \Sigma^*$ to Σ^*. P-selector functions are of this sort. As is standard, we say such a function is *commutative* exactly if

$$(\forall x, y \in \Sigma^*)[f(x,y) = f(y,x)],$$

and we say such a function is *associative* exactly if

$$(\forall x, y, z \in \Sigma^*)[f(x, f(y,z)) = f(f(x,y), z)].$$

This paper, which overviews the advice complexity of the P-selective sets, stresses at times how algebraic properties interact with that study. We mention here in passing a completely different context in which the costlessness of algebraic properties for functions has come up. Consider the theory of one-way functions—in particular, of worst-case one-way functions mapping from $\Sigma^* \times \Sigma^*$ to Σ^*. Work of Rivest and Sherman and of Rabi and Sherman (both reported on by Rabi and Sherman in [RS93,RS97]) proposed attractive protocols for secret-key agreement and for digital signatures. However, the protocols assume not merely that one-way functions exist but that one-way functions exists that in addition are commutative, associative, total (defined on all inputs), and *strongly* noninvertible (given not merely the output but also one of the inputs, one still cannot in general in polynomial time find the missing other input that with the given input maps to the given output). Note the strange situation we have: No one even knows that vanilla one-way functions exist, and there those guys go building protocols that depend on seemingly demanding strengthenings of one-way functions—in particular, ones that add on four properties, including the algebraic properties of associativity and commutativity.

Fortunately, this turns out to be a case where adding properties is costless. In particular, we have the following result of Hemaspaandra and Rothe ([HR99], see also [HPR01] for a surprising recent twist).

Theorem 1.3 *[HR99] One-way functions exist if and only if strong, total, commutative, associative one-way functions exist.*

That is, it is precisely as likely that the elaborate one-functions exist as it is that vanilla one-way functions exist—they stand or fall together.

2 Advice and Algebraic Properties for the Semifeasible Sets

In this section we will present an informal overview of the results known regarding the advice properties of the P-selective sets, and the cost of claiming algebraic properties for selector functions (and, when they occur, the relationships between these two tacks).

2.1 Commutativity Is Free

Let us look first at a result on requiring algebraic properties of selector functions. This result says that every set that has any P-selector function has a commutative P-selector function.

Theorem 2.1 *[Ko83] If A is a P-selective set, then there is a commutative, polynomial-time function that is a P-selector function for A.*

The proof of this is quite easy. Let A be any P-selective set. Let f by a polynomial-time function with respect to which A is P-selective (by the definition of P-selectivity, such a function must exist). The function $f'(x,y) = f(min(x,y), max(x,y))$ is clearly commutative, and yet it is easy to see that it also is a P-selector for A, thus proving the theorem. (The function $f''(x,y) = min(f(x,y), f(y,x))$ could also be used here, as could be the function $f'''(x,y) = max(f(x,y), f(y,x))$.)

2.2 Cashing in on Commutativity: The P-Selective Sets Are in P/quadratic

Let us make a brief digression. Consider a tournament on k nodes. A tournament is a digraph that has no self-loops, and such that between each pair of distinct nodes there is an edge in exactly one of the two directions. (Informally, imagine a clique, and then for each edge in the clique, make the edge be directed one way or the other—what one gets out is a tournament.) We will imagine each node as a player in a round-robin tennis tournament (that allows no ties), and if player a beats player b, then the node between a and b will point from b to a, i.e., arrows point from the loser to the winner. Now, here is a nice fact about tournaments. For any k-node tournament, there will be a superloser set of size at most about $\log(k)$ (all our logs will be base two). By a superloser set, we mean a set of nodes such that each node in the tournament is either in the superloser set or is pointed to by an edge from at least one node in the superloser set. So, for example, if 1000 people play a round-robin tournament, there will certainly be some superloser set of size at most 10, i.e., a group of (at most 10) people such that everyone in the tournament other than those ten people beats at least one of the ten people (though different ones may beat different ones from our ten). (Note in passing: The claim is also true regarding a superwinner set, and sounds more impressive in that context. There will be a group of at size at most 10 such that everyone else in the tournament lost to at least one of these 10 players!) Let us quickly give a proof that there will be a small superloser set. If $k = 0$ the empty set works fine as the superloser set, and we're done. If $k = 1$ put that one person into the superloser set, and we're done. So, suppose $k > 1$. Each of our k players played $k - 1$ games. Suppose every single player lost strictly more games than he or she won. That is impossible, since in each game, one person lost and one person won, so the total numbers of wins and losses over the entire tournament must match. Thus, at least one player lost at least half the games he/she played. Put that person into our superloser set. Now consider the tournament, except with our person removed and all people who beat our person removed. Call this new tournament T'. Note that T' has at most (in fact, strictly less than, if one looks carefully) $k/2$ nodes. In T' one repeats the above (focusing only on wins and losses *within* T'), adding one more node to the superloser set and creating a new tournament T''. And so on. This continues until all nodes are eliminated from consideration. It is easy to see that the process concludes in about $\log(k)$ steps. To be very exact about it, the superloser set will be of cardinality at most $\lfloor \log(k + 1) \rfloor$.

So, how will we use this information? Well, recall that in the previous section we noted that every P-selective set has a commutative selector function. We may use such a function to define a tournament regarding each possible length. That is, let A be an arbitrary P-selective set. Let f be a *commutative* P-selector for A (by Section 2.1 such a function exists). Our goal is to show that $A \in \mathrm{P}/\text{quadratic}$. Consider some arbitrary length n. (In this very informal presentation we will be casual about the uniformity of the interpreter with respect to n—but we mention in passing that it should actually be fixed outside of the scope of n, and that one can easily do so and we act as if that is implicitly being done.) We want to make sure a P set can, with only quadratically many advice bits, determine membership and nonmembership of any one of the 2^n strings of length n. Consider the following tournament. Its nodes will be all string in the set $A \cap \{x \mid n = |x|\}$, i.e., all length-$n$ elements of A. Given two such nodes, x and y, $x \neq y$, we draw an edge from x to y exactly if $f(x, y) = y$. Note that this *is* a tournament, and the reason we know that between two nodes there will be an arrow exactly one way is because f is commutative.

Now, let us invoke the tournament result from above to note that there is a small superloser set. Small in this case means, using the above information, at most of size $\lfloor \log(||A \cap \{x \mid n = |x|\}|| + 1) \rfloor$, and so certainly of cardinality-at-most-$\lfloor \log(2^n + 1) \rfloor = n + 1$. Each member of this cardinality at most $n+1$ set has exactly n bits. So, this set clearly can be represented using a quadratic number of bits.

Finally, note that given as our advice function this particular quadratic-sized information (namely, a superloser set in the above tournament), we can via a P interpreter easily decide the membership of any length-n input string. We do so as follows. Let a_n be the advice string for length n. And let y be the (length-n) input string our interpreter happens to be asked about. Then our P interpreter acts as follows. It decodes from its own input the values of a_n (since we are defining the set, all we know is we are given a purported a_n, and our interpreter must be a P interpreter overall and when given valid advice must do the right thing) and length-n string y. If a_n says that the superloser set is empty, then return "$y \notin A$." Otherwise, let $\{v_1, \ldots, v_j\}$ be the superloser set specified (in any nice, fixed, simple, space-efficient way) by a_n. If either (though actually the first disjunct is superfluous here) $y \in \{v_1, \ldots, v_j\}$ or $(\exists i : 1 \leq i \leq j)[f(v_i, y)] = y$, then output "$y \in A$" else output "$y \notin A$."

Note that, given the specified, quadratic-length advice, this interpreter always is correct regarding membership in A. Why? Consider again some fixed length n. If our interpreter accepts, that must be because either y is in the superloser set (and thus y is in A, since all the members of the superloser set are members of A) or it is the case that there is a member w of the superloser set that y beats with respect to f, the commutative P-selector for A fixed above. In the latter case, y must also be in A, since otherwise we have

$$f(w, y) = y \wedge w \in A \wedge y \notin A,$$

but that would contradict the claim that f is a P-selector for A—since P-selectors, when exactly one of the inputs is in, always pick the one that is in. Thus, if our interpreter accepts y, then $y \in A$. On the other hand, if $y \in A$, then our interpreter will accept y. This holds since by the choice of the superloser set either y belongs to the superloser set or beats at least one member of the superloser set (since every length-n element of A satisfies at least one of those two conditions). And so, by the construction given for the interpreter, in that case the interpreter will indeed accept the string.

The proof just outlined is due to the important early paper of Ko, and proves that each P-selective set belongs to the class P/quadratic. This was the first result ever obtained regarding the advice complexity of the P-selective sets.

Theorem 2.2 *[Ko83]* P-sel $\subseteq$ P/quadratic*.*

2.3 Linear Advice Suffices with Respect to NP Interpreters

In the previous section, we saw that all P-selective sets belong to P/quadratic. So quadratic advice suffices. Can we, by using greater power in the interpreter, get by with less advice—for example, with linear advice?

The answer is yes. In fact, Hemaspaandra, Naik, Ogihara, and Selman [HNOS96a] (without quite explicitly stating it in this form) showed that linear advice suffices if one boosts the power of the interpreter up to the class PP—unbounded-error probabilistic polynomial time.

Theorem 2.3 *[HNOS96a]* P-sel $\subseteq$ PP/linear*.*

Of course, PP is a very powerful class—given the flexibility of polynomial-time Turing reductions, it subsumes the entire polynomial hierarchy (Toda's Theorem [Tod91]). However, Hemaspaandra and Torenvliet soon showed that even NP interpreters suffice to accept all P-selective sets with linear advice.

Theorem 2.4 *[HT96]* P-sel $\subseteq$ NP/linear*. In fact,* P-sel $\subseteq$ NP/$n+1$ *(where* $n+1$ *is a shorthand for the function* $\lambda n.n+1$*).*

In the proof of Hemaspaandra and Torenvliet of this result, a very slightly superlinear number of nondeterministic guess moves (about $n \log^* n$) was used by the NP interpreters. However, Hemaspaandra, Nasipak, and Parkins [HNP98] soon noted that one could achieve P-sel $\subseteq$ NP$/n + 1$ even for NP interpreters limited to making at most a linear number of nondeterministic moves. (That is, we even have that P-sel $\subseteq$ NP$_1/n + 1$, where NP$_1$ is the "level one" of the limited nondeterminism hierarchy presented in the conference paper [KF77]—not the related journal paper that presents a different (namely, polylog nondeterminism) version of the limited nondeterminism hierarchy [KF80]—of Kintala and Fischer.)

We now sketch the proof that P-sel $\subseteq$ NP$_1/n+1$. Let A be an arbitrary P-selective set. Let f be a *commutative* (note how commutativity is again helping us, by soon allowing us to invoke tournament results) P-selector for A. Consider some arbitrary n. Build precisely the same tournament (let us call it T) as in Section 2.2 (the one having one node for each element of $A \cap \{x \mid n = |x|\}$). Before, we used the fact that this tournament has a logarithmic (in the number of nodes in the tournament) cardinality set of nodes from which the entire tournament is at distance at most one. We now will use a different fact, namely, that a tournament always (will either have 0 nodes, or if it has more than 0 nodes) will have a node from which the entire tournament is at distance at most two. Such a node is known as a "king" of the tournament. That is, given any tournament on k nodes, there will always be a node v such that for every node w in the tournament, there will be a directed path of length at most two (i.e., of length 0 or 1 or 2) from v to w. This fact was first noted in print at least as early as a 1953 article in the *Bulletin of Mathematical Biophysics* ([Lan53], but see also the historical comments therein regarding Hohn and Vaughan), which was about dominance among lions and the fact that there will always be a (lion) king. Let us quickly prove this, as it is an easy induction. The $k = 1$ case clearly holds. Consider by way of induction a tournament, U, on $m > 1$ nodes, and by way of induction suppose that we have established our claim for all tournaments on $m - 1$ nodes. Fix and delete any node, t, from U. By our inductive hypothesis, the remaining tournament (call it U') has a king, call it k. If the edge between t and k points to t, then k remains a king of U and we are done. If any node that k points to itself points to t, k remains a king and we are done. Suppose both these cases do not hold. So, t points to k, and every node that k points to is also pointed to by t. Then we also are done, since t will in this case be a king. Why? Well, it reaches itself at distance 0. It reaches k at distance 1. t reaches every node in U' that is distance 1 from k via distance one from itself. And, crucially, t

can reach every node that is distance 2 (and no less) from k via a path of distance 2 as follows: The distance-2 path from k to the node had some intermediate node, *but t points directly to every such node*, so t can intercept the path at its intermediate point, and then exploit the same second edge that k was exploiting. So, we now know that t reaches within distance 2 all points at distance at most 2 from k, but since k was a king of U', that means t reaches all of U' within distance 2, and since t also reaches itself via distance 0, t indeed is a king of U. Thus, now we know that all (nonempty) tournaments have a king. Let us return to the tournament T described above. So, it will either be empty, or will have a king (which will be an n-bit node name). Thus, in $n+1$ bits, we can easily encode whether the tournament is nonempty (i.e., A has at least one length-n string) and, if so, the name of a king. (Note that there are $2^n + 1$ possibilities—2^n potential kings plus the "empty tournament" case.) This $n + 1$-bit object will be our advice string. The NP interpreter (which gets as its input a pair that is purported-advice-for-length-n and length-n-string-to-decide y) will do exactly the obvious thing: If the advice says A has no strings at length n, reject y. Otherwise, let h be the length-n string the advice encodes. If $f(h, y) = y$ then accept, otherwise nondeterministically guess a length-n string a satisfying $h \neq a \neq y$, and if $f(h, a) = a \wedge f(a, y) = y$ then accept (on our current nondeterministic path). This scheme clearly shows membership in $\mathrm{NP}/n + 1$ for our arbitrary P-selective set—and even shows membership in $\mathrm{NP}_1/n + 1$, since the only guessing is the n-bit intermediate node a. Thus, we have proven the following.

Theorem 2.5 *[HNP98]* P-sel $\subseteq$ NP/linear. *In fact,* P-sel $\subseteq \mathrm{NP}_1/n + 1$ *(where $n + 1$ is a shorthand for the function $\lambda n.n + 1$).*

2.4 Optimality of *n+1* Bits of Advice

In the previous section, we saw that P-sel $\subseteq \mathrm{NP}/n + 1$. That is, $n + 1$ bits of advice suffice. Can one in general do better? The answer is a resounding NO.

Theorem 2.6 *[HNP98]* P-sel $\not\subseteq \mathrm{NP}/n$*, that is, there exists a P-selective set A such that $A \notin \mathrm{NP}/n$.*

In fact, the above result isn't complexity-theoretic. It is in effect information-theoretic: There is a multiplicity bottleneck at work here, which is exploited via a diagonalization. So in fact not only do n bits not suffice to let NP sets accept all P-selective sets, but the same claim holds for any reasonable class of sets, e.g., n bits do not suffice to let the recursively enumerable sets accept all P-selective sets!

2.5 Nonoptimality of *n+1* Bits of Advice: Bits Are Not the Last Word in Pinpointing Information Content

I lied. Section 2.4's title, from the fact that P-sel $\subseteq$ NP/$n+1$ and P-sel $\not\subseteq$ NP/n, concludes that $n+1$ bits is optimal. But is it really? A complete answer is a tiny bit (pardon the pun!) more nuanced than the title of Section 2.4 would indicate. In particular, if one cares about the content in terms of bits, then yes, n bits don't suffice in general (though for some specific P-selective sets, such as for example any P set, they easily do), but $n+1$ bits do always suffice.

But let us consider the nature of bits of information. 10-bit strings hold exactly 1024 possible values. 11-bit strings contain exactly 2048 possible values. Suppose some item has exactly 1025 possibilities? 10 bits won't cut it, but 11 bits is wild overkill; all we really need to hold the information is a 1025-ary token (a token that can take on the values $\{1, 2, \ldots, 1025\}$). The natural thing to do would be to classify information not by bits, but the richness of the tokens needed to hold them. For example, for the P-selective sets, note that within the proof given in Section 2.3 it is made clear that we need not $n+1$ bits, but rather just a 2^n+1-ary token. On the other hand, Theorem 2.6 implies that with respect to NP interpreters a 2^n-ary token does not suffice. Thus, for the P-selective sets with respect to NP interpreters (or for that matter EXP or RECURSIVE or RE or etc. interpreters), we actually know the information content right down to the richness of the token needed [HT96]—a much tighter understanding than just getting things down to how many bits are needed.

2.6 But What About P/linear?—Associativity is Free in the Context of Commutativity But Beyond that the Issue is Open

So we know that P-sel $\subseteq$ NP/linear and P-sel $\subseteq$ P/quadratic. Can we get the best of both worlds, by showing that P-sel $\subseteq$ P/linear? Well, no one knows. But a subclass of P-sel is known to fall into P/linear, and that brings us back to our interest in algebraic properties of P-selector functions.

In particular, let CA-P-sel denote all sets that are P-selective via some P-selector function that is both associate and commutative (similarly for C-P-sel and A-P-sel; note that Theorem 2.1 simply says P-sel = C-P-sel). One can show (due to the connection between commutative, associative selector functions and transitive tournaments—and in turn linear orders) the following result of Hemaspaandra, Hempel, and Nickelsen.

Theorem 2.7 *([HHN], see also [HHN01])* CA-P-sel $\subseteq$ P$/n+1$.

So, all we have to do is show that all P-selective sets have commutative, associate selector functions and we will have shown that P-sel $\subseteq$ P/linear. We saw earlier that adding commutativity was free. Is adding associativity free? No one has yet shown that it is, or that it is not. However, it is known that if one has an associative P-selector for a set, one can obtain for it a P-selector that is both commutative and associative. That is, we have the following result.

Theorem 2.8 *([HHN], see also [HHN01])* CA-P-sel = A-P-sel.

This result lets us restate Theorem 2.7 just as the following.

Theorem 2.9 *([HHN], see also [HHN01])* A-P-sel $\subseteq$ P$/n+1$.

So, what can one say about obtaining associative selector functions for all P-selective sets? Well, looking through the prism of relativization theory, Thakur has constructed an oracle world in which some P-selective sets lack linear advice with respect to all deterministic polynomial-time interpreters [Tha03]. This does not resolve the question in the real world, but suggests that relatively creative techniques would be needed to obtain any positive containment in the real world. Back in the real world, we here have come to an example where we cannot (currently) show costlessness for the addition of an algebraic property. And in light of Theorem 2.9 and the lower bound inherited from Theorem 2.6, this is particularly regrettable, since costlessness for this property would resolve totally the advice complexity of the P-selective sets with respect to P interpreters. Nonetheless, one can "buy" associativity for a complexity-theoretic price. In particular, if one is willing to go from P functions (P-selectors) to P^{NP} functions (P^{NP}-selectors), then one can achieve associativity.

Theorem 2.10 *([HHN], see also [HHN01]) If A is a P-selective set, then there is a* P^{NP} *function (i.e., a total function computable by a polynomial-time machine given an* NP *oracle—this is often denoted* $\mathrm{FP}^{\mathrm{NP}}_{\mathrm{total}}$ *or simply* $\mathrm{FP}^{\mathrm{NP}}$*) f that is a selector function for A and that is both commutative and associative.*

However, note that the price just paid to buy associativity is too high to help us regarding obtaining any new theorem for all the P-selective sets: Though P^{NP} functions via associativity can be used to show the claim P-sel $\subseteq \mathrm{P}^{\mathrm{NP}}/n+1$, Theorem 2.4 (P-sel $\subseteq$ NP/linear) is already a stronger claim than that. So Theorem 2.10 is interesting in that it gives the best current upper bound on the cost of achieving associativity; however, it is not helpful regarding advice. In contrast, Theorem 2.9 at least does identify an explicit class that has linear advice with respect to NP interpreters, namely, all sets that have associative P-selector functions.

3 The Fine Print, and Open Questions

The most important bit of fine print to mention is that in this paper I have at points chosen simplicity over strength and generality. (Readers wishing to see more complete presentations will wish to look at the original research papers cited here, and the citations therein; or, at an intermediate level, readers might wish to look first at the book chapter [HO02, Chapter 3], or at the book (most especially its second chapter) [HT03].)

To take a small example of having chosen simplicity over strength and generality, Theorem 2.9 indeed is true, but in the original research paper ([HHN], see also [HHN01]), it is proven not just for associativity, but even for a weakened version of associativity in which the rules of associativity must be respected in contests among strings of the same length, but may be violated in contests among strings of different lengths.

To take a larger example, throughout this paper I've focused only on the P-selective sets—sets selective via *total*, *deterministic* polynomial-time computable functions. In fact, all the issues discussed here—advice complexity and associativity—have been studied (typically either in the same papers cited here, or in the papers introducing nondeterministic selectivity [HHN$^+$95,HNOS96b]) in the setting of selector functions that are allowed to be partial and/or nondeterministic. The results tend to often be analogous to those that hold for P-selectors, but the proofs can be far more subtle and complex—especially in the case of partial functions.

We mention now just one issue related to nondeterministic analogs of selectivity—namely, the issue that was alluded to in the introduction. Consider an NP machine that on each rejecting path is considered to have no output, and that on each accepting path is viewed as having as an output whatever is on a special output tape. Since such a machine can have multiple accepting paths, it can have multiple outputs on a given input—such machines map from inputs to a set of outputs. Such machines define the NPMV (NP multivalued) functions [BLS84,BLS85]. Note that it is easy to make such a machine that when given a boolean formula f nondeterministically guesses an assignment and if it is a satisfying assignment then the machine on that path accepts and outputs that assignment. Thus, there are NPMV functions that output *all* satisfying assignments of (input, satisfiable) boolean formulas (and that of course output no assignments for unsatisfiable ones). But consider the following challenge: Are there NPMV functions that output (exactly) *one* satisfying assignment of (input, satisfiable) boolean formulas (and that of course output no assignments for unsatisfiable ones), i.e., when the input formula is satisfiable, at least one path is accepting and every accepting path outputs the same satisfying assignment? If not, then we have a problem for which finding all solutions is easier than finding one solution! One might find that paradoxical, but it is not. It is a side effect of the way nondeterminism works. One can't just say "get all solutions and then throw out all but the lexicographically smallest." That would be legal with deterministic computation, but for NPMV functions, a given path doesn't have a clue as to whether to throw itself out—it would seem to need NP = coNP to decide whether to do such throwing. The reason we mention this is that the nondeterministic, partial-functions variant of P-selectivity is how this issue was resolved. In particular (and let us not here define these terms or exactly what selectivity with respect to partial functions means), Hemaspaandra, Naik, Ogihara, and Selman [HNOS96b] showed that every NP set that is selective via a (possibly partial) single-valued NPMV function belongs to the advice class $(\mathrm{NP} \cap \mathrm{coNP})/\mathrm{poly}$, and from this concluded that if any NPMV function can output (exactly) *one* satisfying assignment of (input, satisfiable) boolean formulas (and have no output for unsatisfiable ones) then the polynomial hierarchy collapses. (A collapse to $\mathrm{ZPP}^{\mathrm{NP}}$, and thus certainly to $\mathrm{NP}^{\mathrm{NP}}$, is obtained by [HNOS96b], but due to the very recent strengthening by Cai et al. [CCHO03] of the consequences of the above-mentioned advice result, one can now even conclude a collapse to $\mathrm{S}_2^{\mathrm{NP} \cap \mathrm{coNP}}$—see [CCHO03] for further discussion.) What is so surprising about this result is that selectivity theory and the issue of unique solutions seemed previously to have no relationship at all. So this is an example where the study of advice and selectivity resulted in the resolution of a question from a completely different area. Also, what this says is that, unless the polynomial hierarchy collapses, finding all satisfying assignments really is easier than finding one satisfying assignment!

Selective sets have many cousins and generalizations, and for them too one can study issues of advice and so on. We do not discuss that here, but the book chapter [HT03, Chapter 6] provides some quick pointers toward many of these results, and also useful is the nice article of Nickelsen and Tantau [NT03].

To conclude, let us highlight the most glaring open issues regarding advice and the P-selective sets. Namely, does every P-selective set have an associative P-selector function? And, does every P-selective set belong to the class P/linear? A "yes" answer to the first would (by Theorem 2.9) imply a “yes” answer to the second, but it is not known that a “yes” answer to the second would imply a “yes” answer to the first. Stepping back a bit, the study of the P-selective sets has made tremendous progress in the quarter century since they were first defined in the seminal 1979 paper of Selman [Sel79]. Nonetheless, even after all that time, they provide a broad range of open issues, and we still don’t even know the worst-case advice complexity of the P-selective sets, with respect to P interpreters, more precisely than “quadratically many bits will suffice and n bits is impossible (but $n + 1$ bits potentially might suffice).” Clearly, there is much room here for inquisitive minds and new approaches. Please consider this paper your personal invitation!

Acknowledgments

I thank the DCFS ’04 team for the chance to write this paper and share this discussion, and I thank my coauthors in the papers cited here for the great pleasure of working with them on the study of semifeasible computation.

References

[Adl78] L. Adleman. Two theorems on random polynomial time. In *Proceedings of the 19th IEEE Symposium on Foundations of Computer Science*, pages 75–83. IEEE Computer Society, October 1978.

[BH77] L. Berman and J. Hartmanis. On isomorphisms and density of NP and other complete sets. *SIAM Journal on Computing*, 6(2):305–322, 1977.

[BLS84] R. Book, T. Long, and A. Selman. Quantitative relativizations of complexity classes. *SIAM Journal on Computing*, 13(3):461–487, 1984.

[BLS85] R. Book, T. Long, and A. Selman. Qualitative relativizations of complexity classes. *Journal of Computer and System Sciences*, 30(3):395–413, 1985.

[CCHO03] J. Cai, V. Chakaravarthy, L. Hemaspaandra, and M. Ogihara. Some Karp–Lipton-type theorems based on S_2. In *Proceedings of the 20th Annual Symposium on Theoretical Aspects of Computer Science*, pages 535–546. Springer-Verlag *Lecture Notes in Computer Science #2607*, February/March 2003.

[HHN] L. Hemaspaandra, H. Hempel, and A. Nickelsen. Algebraic properties for selector functions. *SIAM Journal on Computing.* To appear.

[HHN$^+$95] 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.

[HHN01] 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*, August 2001.

[HNOS96a] 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.

[HNOS96b] 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.

[HNP98] 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.

[HO02] L. Hemaspaandra and M. Ogihara. *The Complexity Theory Companion.* Springer-Verlag, 2002.

[HPR01] L. Hemaspaandra, K. Pasanen, and J. Rothe. If P $\neq$ NP then some strongly noninvertible functions are invertible. In *Proceedings of the 13th International Symposium on Fundamentals of Computation Theory*, pages 162–171. Springer-Verlag *Lecture Notes in Computer Science #2138*, August 2001.

[HR99] L. Hemaspaandra and J. Rothe. Creating strong, total, commutative, associative one-way functions from any one-way function in complexity theory. *Journal of Computer and System Sciences*, 58(3):648–659, 1999.

[HT96] L. Hemaspaandra and L. Torenvliet. Optimal advice. *Theoretical Computer Science*, 154(2):367–377, 1996.

[HT03] L. Hemaspaandra and L. Torenvliet. *Theory of Semi-Feasible Algorithms.* Springer-Verlag, 2003.

[Joc68] C. Jockusch. Semirecursive sets and positive reducibility. *Transactions of the AMS*, 131(2):420–436, 1968.

[KF77] C. Kintala and P. Fischer. Computations with a restricted number of nondeterministic steps. In *Proceedings of the 9th ACM Symposium on Theory of Computing*, pages 178–185. ACM Press, May 1977.

[KF80] C. Kintala and P. Fisher. Refining nondeterminism in relativized polynomial-time bounded computations. *SIAM Journal on Computing*, 9(1):46–53, 1980.

[KL80] R. Karp and R. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the 12th ACM Symposium on Theory of Computing*, pages 302–309. ACM Press, April 1980. An extended version has also appeared as: Turing machines that take advice, *L'Enseignement Mathématique*, 2nd series, 28:191–209, 1982.

[Ko82] K. Ko. The maximum value problem and NP real numbers. *Journal of Computer and System Sciences*, 24(1):15–35, 1982.

[Ko83] K. Ko. On self-reducibility and weak P-selectivity. *Journal of Computer and System Sciences*, 26(2):209–221, 1983.

[Lan53] 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.

[NT03] A. Nickelsen and T. Tantau. Partial information classes. *SIGACT News*, 34(1), 2003.

[RS93] M. Rabi and A. Sherman. Associative one-way functions: A new paradigm for secret-key agreement and digital signatures. Technical Report CS-TR-3183/UMIACS-TR-93-124, Department of Computer Science, University of Maryland, College Park, Maryland, 1993.

[RS97] M. Rabi and A. Sherman. An observation on associative one-way functions in complexity theory. *Information Processing Letters*, 64(5):239–244, 1997.

[Sch86] U. Schöning. *Complexity and Structure*. Springer Verlag *Lecture Notes in Computer Science #211*, 1986.

[Sel79] A. Selman. P-selective sets, tally languages, and the behavior of polynomial time reducibilities on NP. *Mathematical Systems Theory*, 13(1):55–65, 1979.

[Sel81] A. Selman. Some observations on NP real numbers and P-selective sets. *Journal of Computer and System Sciences*, 23(3):326–332, 1981.

[Sel82] A. Selman. Reductions on NP and P-selective sets. *Theoretical Computer Science*, 19(3):287–304, 1982.

[Tha03] M. Thakur. On optimal advice for P-selective sets. Technical Report TR-819, Department of Computer Science, University of Rochester, Rochester, NY, November 2003.

[Tod91] S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 20(5):865–877, 1991.