P-selective Self-reducible sets: A New Characterization of P

Harry Buhrman   Leen Torenvliet

Department of Mathematics and Computer Science
University of Amsterdam
Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands

Abstract

We show that any p-selective and self-reducible set is in P. As the converse is also true, we obtain a new characterization of the class P. A generalization and several consequences of this theorem are discussed. Among other consequences, we show that under reasonable assumptions auto-reducibility and self-reducibility differ on NP, and that there are non-p-T-mitotic sets in NP.

1 Introduction

Separating complexity classes is a very popular, but rarely won game in complexity theory. Frustrated by misfortune, computer scientists have often turned to attempts of characterizing complexity classes in a different way. The hopes are, that the new characterization of the complexity class may provide new insights and a 'handle' to force the separation where earlier attempts have failed. Well-known examples of this are the many ways to define the class of sets for which there exist small circuits [Pip79], and the identification of various forms of interactive proof systems with standard complexity classes as $PSPACE$, EXP and NP [Sha90, BFL90, ALM$^+$92]. Also, the classification of complexity classes by various logical theories is a rapidly growing field of interest [Imm84, Imm87].

The class P, of polynomial time decidable sets, was first described by Edmonds [Edm65] as the class of problems for which feasible algorithms exist. Unfortunately, many problems of interest are not known to be in P. Therefore, interest has shifted from P to classes 'near' P, and classes of sets as 'near-testable' [GHJY91], 'p-selective' [Sel79], 'self-reducible' [MP79] and such have been defined. For many of these, characterizations in terms of standard complexity classes are more or less known. For instance it is known that all self-reducible sets are in $PSPACE$, all p-cheatable self-reducible sets are in P [GJY93], all p-cheatable and near testable sets are in P [GJY93], and that all disjunctively self-reducible sets are in NP [Ko83]. Furthermore it is known that disjunctively self-reducible sets that are in addition p-selective are all in P [Sel79].

In this paper, we give a generalization of that theorem. We show that if a set is both p-selective and Turing self-reducible, then it is in P. As the converse is trivially true, this gives a new characterization of P as the class of sets that are both p-selective and self-reducible. The theorem is proved by first showing that a p-selective set that is truth-table self-reducible is in fact many-one self-reducible. It is known [Ko83] that every many-one self-reducible set is in P. Next, we generalize the theorem by showing that we can construct a truth-table self-reduction from a given Turing self-reduction. The two theorems hold, not only for self-reducible sets, but also for auto-reducible sets. Unfortunately, a many-one auto-reducible set may have non-trivial complexity. We show that even within a class as NP, auto-reducibility and self-reducibility probably differ.

The rest of this paper is organized as follows. In Section 2, we discuss some of the preliminaries and definitions. In Section 3, we state our main theorems. In Section 4, we discuss some of the consequences of the theorems, a generalization, and limits to obtaining further generalizations.

2 Preliminaries

Let $\Sigma = \{0, 1\}$. Strings are elements of Σ^*, and are denoted by small letters $x, y, u, v, \ldots$. For any string x, the length of x is denoted by $|x|$. Languages are subsets of Σ^*, and are denoted by capital letters $A, B, C, S, \ldots$. For a set A, the function χ_A will denote the characteristic function of A, i.e. $\chi_A(x) = 1$ if $x \in A$, and $\chi_A(x) = 0$ otherwise. For any set S, the cardinality of S is denoted by $\|S\|$. We assume that the reader is familiar with the standard Turing machine model and other standard notions of complexity theory, as can be found in [BDG88]. Nevertheless, some of the definitions that we feel may not be common knowledge are cited below.

An *oracle* machine is a multi-tape Turing machine with an input tape, an output tape, work tapes, and a *query* tape. Oracle machines have three distinguished states QUERY, YES and NO, which are explained as follows: at some stage(s) in the computation the machine may enter the state QUERY, and then goes to the state YES, or goes to the state NO, depending on the membership of the string currently written on the query tape in a fixed *oracle* set.

Oracle machines appear in the paper in two flavors: adaptive and non-adaptive. For a non-adaptive machine, queries may not be interdependent, whereas an adaptive machine may compute the next query depending on the answer to previous queries. If a Turing machine M accepts (rejects) a string x, we will sometimes write $M(x) = 1$ ($M(x) = 0$). We use the same notation for oracle machines ($M^A(x) = 0/1$). The set of strings recognized by a Turing (oracle) machine (with oracle A), is denoted by $L(M)$ ($L(M, A)$).

We use polynomial time bounded adaptive oracle machines to model Turing reductions ($\leq_T^P$), and non-adaptive machines to model *truth-table* reductions ($\leq_{tt}^P$). For polynomial time bounded machines, this yields definitions equivalent to the standard definitions of reducibilities in [LLS75]. If the number of queries in a truth-table reduction is fixed by some constant, we call such a reduction a *bounded truth-table reduction.* If we can identify this constant, i.e. no more than k queries are generated on any input, then we call such a reduction a k-truth-table reduction. A special case is $k = 1$, which appears in one of our theorems, the $\leq_{1-tt}^P$-reduction. If $k = 1$ and moreover, the machine reducing A to B accepts x iff the (single) query generated is in B, then we speak of a *many-one* reduction, or $\leq_m^P$-reduction. A many-one reduction that sometimes accepts or rejects without producing a query is called $\leq_{\hat{m}}^P$-reduction. Such a reduction can easily be translated into a many-one reduction for oracle sets that are neither $\emptyset$ nor Σ^*. Therefore, we will make no significant difference between these two types of reductions. A reduction (of any of the above types) is called *positive* iff for any two oracles $A \subseteq B$ it holds that $L(M, A) \subseteq L(M, B)$ [Sel82].

The set of queries generated on input x by oracle machine M is denoted $Q_M(x)$. For adaptive machines, this set may be oracle dependent, and is therefore denoted $Q_M^A(x)$, if A is the oracle set. The (possibly exponential size) set of all *possible* queries generated by adaptive machine M on input x—also called the *query tree* of M on input x—is denoted $QT_M(x)$.

Meyer and Paterson [MP79] introduced self-reducible sets. A more accessible definition can be found e.g. in [Ko83]. We first copy Ko's definition of a polynomially well-founded and length related ordering.

Definition 1: A partial ordering $<$ on Σ^* is *polynomially well-founded* *and length related* if there is a polynomial p such that

1. $x \lessdot y$ can be decided in $p(|x| + |y|)$ steps
2. $x \lessdot y$ implies $|x| \leq p(|y|)$ for all $x, y \in \Sigma^*$ and
3. the length of a $\lessdot$-decreasing chain is shorter than p of its $\lessdot$-maximal element.

Relative to such an order, we can define self-reducibility.

Definition 2: A set A is (polynomial time) *self-reducible* iff there exists a polynomially well-founded length related order $\lessdot$ on Σ^*, such that $A \leq_T^P A$ via an oracle machine M that on input x queries only strings y for which $y \lessdot x$.

Of course, we may replace the adaptive self-reducing machine by a non-adaptive machine and obtain *non-adaptive* or *truth-table* self-reducibility. Self-reducibility is a special case of *auto-reducibility*, for which it is just required that M, on input x, queries only strings y for which $y \neq x$. Self-reducibility is strongly related to a property that is shared by some sets in *NP*. This notion was introduced by Borodin and Demers [BD76] as *functional self-reducibility*. A set A is called functional self-reducible if a proof for membership in A for a string x, can be generated in polynomial time using A as an oracle. We will adopt modern terminology for this property, and say that such a set has Search Reduces to Decision (SRTD).

Definition 3: [BD76, NOS93, BBFG91] Let L be in *NP*. $x \in L$ can be defined as $\exists y[|y| \leq p(|x|) \wedge R_L(x,y)]$ for some polynomial p and some polynomial time computable relation R_L. We say that R_L and p define L. Let $\textsf{witness}_{L,R}(x) = \{y | R_L(x,y)\}$. We say that *Search Reduces to Decision* for L iff there exists a partial function $f^L \in PF^L$ such that for all $x \in \Sigma^* : x \in L \leftrightarrow f^L(x) \in \textsf{witness}_{L,R}(x)$, for some relation R_L defining L. The oracle access of f can be defined both adaptively and non-adaptively. SRTD is sometimes equipped with the prefix '(non)adaptive' accordingly.

Selman introduced p-selective sets in [Sel79], which are a direct translation of the *semi-recursive* sets introduced, in the context of an attempted solution to Post's program, by Jockusch [Joc68].

Definition 4: A set A is called *p-selective* iff there exists a polynomial time computable function $f : \Sigma^* \times \Sigma^* \mapsto \Sigma^*$, called a *p-selector*, such that for any $x, y \in \Sigma^*$

1. $f(x,y) \in \{x,y\}$ and
2. $\chi_A(f(x,y)) = \max\{\chi_A(x), \chi_A(y)\}$

3 Main results

Selman showed in [Sel79] that a set L is disjunctively self-reducible *and* p-selective if and only if L is in P. In [NOS93], a similar characterization of P is obtained, which shows that only the sets in P can be both p-selective and have search nonadaptively reducing to decision

We obtain another characterization of P here, which is a direct generalization of the result in [Sel79] and a direct improvement upon [Ko83]. We show for general (Turing) self-reducible sets, that p-selectivity of such a set means that it is in P. The best known result before our theorem was due to Ko [Ko83], who proved that p-selective self-reducible sets are all in Σ_2^P. Before we prove our theorem, let us make a useful statement on p-selective sets.

Lemma 1 *Let B be a p-selective set and $V \subseteq \Sigma^*$. The p-selector f for B induces a total quasi order $\preceq_f$ on V, such that:*

1. *$\forall x, y \in V : x \preceq_f y \rightarrow [x \in B \rightarrow y \in B]$.*
2. *any finite set $V \subseteq \Sigma^*$ can be ordered in time polynomial in $\|V\| \times \max\{|x| \mid x \in V\}$ as $V = \{x_1, \ldots, x_n\}$ such that $x_i \preceq_f x_{i+1}$.*

Proof. We define $x \preceq_f y$ iff there exists a sequence $z_0 = x, \ldots, z_n = y$ such that $f(z_i, z_{i+1}) = z_{i+1}$. Then this order is total, since $f(x,y) \in \{x,y\}$. It remains to show that $x \in B \rightarrow y \in B$ can be concluded from $x \preceq_f y$. Suppose $x \preceq_f y$, that $z_0 = x, \ldots, z_n = y$, where $f(z_i, z_{i+1}) = z_{i+1}$, and $x \in B$. Then $\chi_B(x) = 1$ and $\chi_B(z_{i+1}) \geq \chi_B(z_i)$ for all i. Hence $\chi_B(y) = 1$.

For 2 we will use induction on the size of V. For any two strings x_i and x_j in V, either $x_i \preceq_f x_j$ or $x_j \preceq_f x_i$ or both, and in time polynomial in $|x_i| + |x_j|$ we can decide which of the first two cases hold. Next we play a *knock-out tournament*(cf. [Moo68] page 48) among the n strings in V, where we say that x beats y if $x \preceq_f y$. (If $x \preceq_f y$ is established, then $y \preceq_f x$ is not examined, so a draw may end in an arbitrary winner, which is o.k., since in that case $x \in B$ iff $y \in B$.) Let x_0 be the winner of the tournament. If $x_0 \in B$, then $x_i \in B$ for all i, and if there is an i such that $x_i \notin B$, then $x_0 \notin B$. By induction hypothesis, the set of $n-1$ strings $V - \{x_0\}$ can be ordered into a chain $\{x_1, \ldots, x_{n-1}\}$, such that $x_i \preceq_f x_{i+1}$. Then $x_0 \preceq_f x_1 \preceq_f \ldots \preceq_f x_{n-1}$ is the chain searched for. $\square$

We note here that although for any two strings x and y and p-selector f one of $x \preceq_f y$ or $y \preceq_f x$ can be decided in time polynomial in $|x| + |y|$, but the question whether both hold (i.e. if a chain can be found that connects x and y in the other direction) may be undecidable for certain f.

The chain order of finite sets of strings is all we need to prove the first theorem, which states that p-selective truth-table auto-reducible sets are in fact many-one auto-reducible. On input x, we let the nonadaptive machine generate a list of queries, order these, and conclude that one query is enough to determine membership of x in the set.

Theorem 2 *A set A that is both p-selective and auto-reducible via a non-adaptive machine M is many-one auto-reducible via M' such that $Q_{M'}(x) \subseteq Q_M(x)$.*

Proof. Let M_a witness the non-adaptive auto-reduction for A and let f be the p-selector. We will show that there exist an $\leq_{\hat{m}}^P$-reduction from A to A, in which the string produced (if any) is one of the queries in the truth-table reduction. We assume that M_a generates at least one query on input x, otherwise the $\leq_{\hat{m}}^P$-auto-reduction is straightforward. We simulate $M_a(x)$ to generate the queries $q_1, \ldots, q_{p(|x|)}$. Now we order the set consisting of these queries *and* x into a chain $z_1, \ldots, z_{p(|x|)+1}$, according to Lemma 3 (i.e. such that $\chi_A(z_i) \leq \chi_A(z_{i+1})$). We claim that the query q with the property that $x \in A \leftrightarrow q \in A$, is immediately adjacent to x in this chain. Without loss of generality, we can assume that the queries in the truth table $q_1, \ldots, q_{p(n)}$ are (re)numbered in the same order as these queries appear in the sequence $z_1, \ldots, z_{p(n)+1}$, i.e. we assume that $\chi_A(q_i) \leq \chi_A(q_{i+1})$. Then, of the $2^{p(n)}$ possible values for the string $\chi_A(q_1) \ldots \chi_A(q_{p(n)})$ only the $p(n)+1$ strings $0\ldots0, 0\ldots01, 0\ldots011, \ldots, 01\ldots1, 1\ldots1$ remain possible. The corresponding values for $\chi_A(x)$ can easily be derived from the program M_a, by simulating $M_a(x)$ for each of these strings in turn, deciding a next state for each query state on the corresponding bit of the string, rather than querying the oracle.

We conclude that the truth-table degenerates to

$\chi_A(q_1)$	$\chi_A(q_2)$	$\dots$	$\chi_A(q_{p(n-1)})$	$\chi_A(q_{p(n)})$	$\chi_A(x)$
0	0	$\dots$	0	0	b_1
0	0	$\dots$	0	1	b_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
0	1	$\dots$	1	1	$b_{p(n)}$
1	1	$\dots$	1	1	$b_{p(n)+1}$

where $b_i = 0$ or $b_i = 1$, depending on whether the simulation to obtain this value described above, ends in reject or accept.

The string x can appear in three possible positions in the sequence $q_1, \dots, q_p(n)$.

1. $x \preceq_f q_1$. Then $\chi_A(q_1) \geq \chi_A(x)$. On the other hand, the only row in which $\chi_A(q_1) \neq 0$ is the $(p(n)+1)$-th row of the truth table. Hence we can conclude that either $x \notin A$ (in case $b_{p(n)+1} = 0$), or that $x \in A \leftrightarrow q_1 \in A$.

2. $q_{p(n)} \preceq_f x$. Then $\chi_A(q_n) \leq \chi_A(x)$. On the other hand, the only row in which $\chi_A(q_n) = 0$ is the first row. So we can conclude that either $x \in A$ (in case $b_1 = 1$), or that $x \in A \leftrightarrow q_n \in A$.

3. $\exists i : q_i \preceq_f x \preceq_f q_{i+1}$. In this case $\chi_A(q_i) = 1 \rightarrow x \in A$ and $\chi_A(q_{i+1}) = 0 \rightarrow x \notin A$. If $\chi_A(q_i) = 0$ and $\chi_A(q_{i+1}) = 1$, then $\chi_A(q_j) = 0$ for all $j \leq i$ and $\chi_A(q_j) = 1$ for all $j \geq i+1$. Therefore $\chi_A(x) = b_{p(n)-i+1}$. Depending on the value of $b_{p(n)-i+1}$, there remain two possible truth tables.

$\chi_A(q_i)$	$\chi_A(q_{i+1})$	$\chi_A(x)$	$\chi_A(q_i)$	$\chi_A(q_{i+1})$	$\chi_A(x)$
0	0	0	0	0	0
0	1	0	0	1	1
1	1	1	1	1	1

So either $\chi_A(x) = \chi_A(q_i)$ or $\chi_A(x) = \chi_A(q_{i+1})$ and we can decide which is the case, without querying the oracle.

$\square$

Next we show, that a Turing auto-reduction for a p-selective set can be transformed into a truth-table auto-reduction and therefore, with the help of Theorem 2, into a many-one auto-reduction.

Theorem 3 *A set A that is both p-selective and auto-reducible via machine M, is auto-reducible via a non-adaptive machine that queries a polynomial subset of $QT_M(x)$*

Proof. Let M_a be an adaptive auto-reduction for A, and let f be a p-selector for A. We will show how to construct a non-adaptive auto-reduction $M_{a'}$ for A. We start by simulating M_a on input x. If M_a accepts x or rejects x without ever reaching the query state then we're done, so we may assume that M_a, on input x, reaches the state QUERY at least once. We simulate the computation of M_a. Each time that M_a reaches the query state, we continue the computation either in the YES state or the NO state, not by querying the oracle, but by the outcome of the computation of $f(x,q)$, where q is the query presently written on the oracle tape. By this process we reach either an accepting or a rejecting state, and identify on the way a polynomial number of queries $q_1, \ldots, q_{p(n)}$. We claim that the truth-table can be built from these queries, the accept or reject state reached, and the outcome of the selector computations on the way. We will first however describe the simulation. Let $q_1, \ldots, q_{k-1}$ be a sequence of queries generated by the simulation so far, and assume that the simulation has reached the state QUERY with q_k written on the oracle tape.

If $f(x, q_k) = q_k$ then $q_k \notin A \to x \notin A$, so we continue the simulation in the YES state. Otherwise, we know that $q_k \in A \to x \in A$. In this case, we continue the simulation in the NO state.

CLAIM 4 *If $q_1, \ldots, q_k$ is the sequence of queries generated by the simulation above, then there exists a nonadaptive oracle machine that generates $q_1, \ldots, q_k$ and computes $\chi_A(x)$ correctly on the basis of the answers to these queries.*

Proof. The simulation doesn't query the oracle, so a non-adaptive machine $M_{a'}$ that generates these queries exists. Now having received the answers to the queries and hence knowing $\chi_A(q_1), \ldots, \chi(q_k)$, machine $M_{a'}$ correctly computes $\chi_A(x)$ as follows. If for all i, $\chi_A(q_i) = 0 \leftrightarrow q_i \preceq_f x$, then the simulation has assumed the right answer to the queries and the computation path followed is correct. We can accept or reject according to the simulation. Otherwise, we can identify a q_i, such that either $\chi_A(q_i) = 0$ and $x \preceq_f q_i$, in which case we can reject, or $q_i \preceq x$ and $\chi_A(q_i) = 1$, in which case we can accept. □

□

Close inspection of the proof of Theorem 3 shows that the truth-table reduction produced is in fact a positive truth-table. Hence we might refer to Selman [Sel82], who showed that a set that is positive truth-table self-reducible and P-selective is in P, instead of to Theorem 2, to obtain the following corollary for self-reducible sets. Recall that self-reducibility is a special case of auto-reducibility, and that a many-one self-reducible set is in P.

Corollary 5 *A set A is (Turing) self-reducible and p-selective if and only if A is in P*

4 Some consequences and questions

In this (last) section, we will discuss some corollaries to the theorems, and attempt a generalization. We will indicate why further generalizations are probably hard to obtain (if they can be obtained at all).

4.1 Consequences of the theorems

From the assumption $E \neq NE$, it follows that there exists a tally set in $NP - P$ [Boo74]. Selman showed in [Sel82], that for any tally language T, there exists a p-selective set A such that $T \leq_T^P A$ and $A \leq_{pos-tt}^P T$. From this, it follows that $E \neq NE$ implies the existence of a p-selective set in $NP - P$. Our theorems imply that such a set cannot be self-reducible. Since any self-reducible tally set is in P, the existence of a non-self-reducible set in $NP - P$ follows directly from [Boo74]. Not every tally set however, is p-selective.

Corollary 6 *There exists a p-selective set in NP that is not self-reducible unless $NE = E$.*

Naik, Ogihara and Selman [NOS93] show, under the assumption $NE \cap \text{co}-NE \neq E$, that there exists a p-selective set in $NP - P$ for which search reduces to decision. We claim that 'search reduces to decision' implies auto-reducibility.

Proposition 7 *If A is a set for which search reduces to decision, then A is auto-reducible.*

Proof. Let M_{SRTD} witness the fact that A has SRTD, for a relation R_A and a polynomial p defining A. We are going to define M_{auto} that is an auto-reduction for A. M_{auto} works as follows. On input x, simulate M_{SRTD}. M_{auto} records each bit written on the query tape on a special tape. M_{auto} follows the actions taken by M_{SRTD}, unless M_{SRTD} enters the query state *and* the string written on the query tape is x. In that case, M_{auto} continues the simulation in the YES state. Let z be the output of the computation. Accept iff $R_A(x,z)$. First we note, that indeed the input string x is never queried by M_{auto}. To see that $M^A_{auto}(x) = \chi_A(x)$, observe that if $x \in A$ then the assumed answers to the query x (if it appears) are all correct. Hence M_{SRTD} outputs a string such that $R_A(x,z)$. On the other hand, if $x \notin A$, then a string z such that $R(x,z)$ *does not exist*. Hence such a string can not be output of the computation. □

It is known that all sets in NP that are complete under $\leq^P_T$ reductions have SRTD [Sel92]. Together with proposition 7, this gives the following corollary.

Corollary 8 *[BF92] If A is $\leq^P_T$-complete for NP then A is auto-reducible*

This corollary is also interesting seen in the light that such a fact is plainly *not* true for r.e. sets, since Ladner [Lad73] has demonstrated $\leq_T$-complete r.e. sets that are not mitotic and hence not auto-reducible (for recursive reductions of course).

The p-selective set in $NP - P$ cited before [NOS93] is clearly not self-reducible. As it does have SRTD, and hence is auto-reducible, we conclude:

Corollary 9 *There exists a set in NP that is auto-reducible but not self-reducible unless $NE \cap \text{co}-NE = E$.*

Beigel et al. [BBFG91] have shown the existence of a set (under the assumption that $NEE \neq EE$) for which search does not reduce to decision. It is straightforward to see that this set is also not auto-reducible. If $NEE \neq EE$ then $NP - P$ has a so-called log-sparse set T, i.e. a tally set for which the difference in length between two elements is exponential. Naik [Nai92] observed, in answer to an earlier question, that $T \oplus T$ is a set that *is* auto-reducible, but for which still search does not reduce to decision. To sum up, we find that for NP (under appropriate assumptions):

1. Any set for which search reduces to decision is auto-reducible

2. Any set that is self-reducible is also auto-reducible
3. There exists a set that is auto-reducible, for which search does not reduce to decision.
4. There exists a set that is auto-reducible, which is not self-reducible.
5. There exists a set for which search reduces to decision, which is not self-reducible.

This leaves open only the question: Is there a set in NP that is self-reducible, for which search does not reduce to decision? As for the interesting question whether every self-reducible set in NP is disjunctively self-reducible, we note that Naik [Nai92] has shown under the assumption $UE \neq \text{co}-UE$, that there exists a set in NP which is conjunctively self-reducible yet not disjunctively self-reducible.

Ambos-Spies [AS84] has shown, that any set that is p-T-mitotic is also p-T-auto-reducible. It follows immediately from this observation that

Corollary 10 *There exists a non p-T-mitotic set in NP unless $NEE = EE$.*

4.2 A first generalization

The theorems in the previous section state results about sets that are both self-reducible and p-selective. We could weaken the assumption of the theorems by assuming that a self-reducible set A is not itself p-selective, but *reducible* to some p-selective set B, and see what this implies for the complexity of A for different reducibilities. Of course, if $A \leq_m^P B$, then A is p-selective via the selector function for B, so $A \in P$ follows. It is known however [HHO+92], that there are sets that are not p-selective, but $\leq_{1-tt}^P$-reducible to a p-selective set. Therefore, a proof that a self-reducible set A $\leq_{1-tt}^P$-reducible to a set B is in P, is a result that does not follow directly from the theorems in the previous section.

Theorem 11 *Let A be auto-reducible via M and B be p-selective. Suppose $A \leq_{1-tt}^P B$. Then A is $\leq_{1-tt}^P$-auto-reducible via M', where $Q_{M'}(x) \subseteq QT_M(x)$*

Proof. First we show how to transform the adaptive auto-reduction into a non-adaptive auto-reduction with the help of the $\leq_{1-tt}^P$-reduction from A to B, using the p-selector f for B. Although A may not be p-selective, we can still order any two strings in a certain way. Let M_1 witness the fact that $A \leq^P_{1-tt} B$ and let z_1 and z_2 be strings. Let z'_1 be the query generated on input z_1 by M_1 and z'_2 the query generated on input z_2. The following four situations (omitting degenerate cases) can occur:

1. $z_1 \in A \leftrightarrow z'_1 \in B$ and $z_2 \in A \leftrightarrow z'_2 \in B$
2. $z_1 \in A \leftrightarrow z'_1 \notin B$ and $z_2 \in A \leftrightarrow z'_2 \notin B$
3. $z_1 \in A \leftrightarrow z'_1 \notin B$ and $z_2 \in A \leftrightarrow z'_2 \in B$
4. $z_1 \in A \leftrightarrow z'_1 \in B$ and $z_2 \in A \leftrightarrow z'_2 \notin B$

We wish to perform a simulation of the Turing reduction as in the proof of Theorem 3, and produce a polynomial number of queries from which the truth-table can be constructed. We do not have a p-selector for A, but we can, given x and the query q_k, produce the queries x' and q'_k, by partially simulating the $\leq^P_{1-tt}$-reduction M_1. We treat the four cases above separately. Let x play the role of z_1, and let q_k play the role of z_2. The queries generated by M_1 on input x and q_k are then z'_1 and z'_2, respectively. Compute $f(z'_1, z'_2)$. The two possible outcomes of this, and the four possible cases above, give eight possible continuation decisions in the simulation. Coarsely, these eight cases reduce to two: If we can conclude $\chi_A(x)$ from the assumption $q_k \in A$, then we continue in the NO state, and if we can conclude $\chi_A(x)$ from the assumption $q_k \notin A$, then we continue in the YES state, so that the path defined produces a truth-table as in the proof of Theorem 2. The eight cases are spelled out here to show that we always land in one of the two cases. The reader may skip to the next paragraph without missing crucial lines in the proof.

i. case 1 and $z'_1 \preceq_f z'_2$. In this case we know that $q_k \notin A \rightarrow z'_2 \notin B \rightarrow z'_1 \notin B \rightarrow x \notin A$. Hence the simulation is continued in the YES state.

ii. case 1 and $z'_2 \preceq_f z'_1$. In this case we know that $q_k \in A \rightarrow z'_2 \in B \rightarrow z'_1 \in B \rightarrow x \in A$. Hence the simulation is continued in the NO state.

iii. case 2 and $z'_1 \preceq_f z'_2$. In this case we know that $q_k \in A \rightarrow z'_2 \notin B \rightarrow z'_1 \notin B \rightarrow x \in A$. Hence the simulation is contiued in the NO state.

iv. case 2 and $z'_2 \preceq_f z'_1$. In this case we know that $q_k \notin A \rightarrow z'_2 \in B \rightarrow z'_1 \in B \rightarrow x \notin A$. Hence the simulation is contiued in the YES state.

v. case 3 and $z_1' \preceq_f z_2'$. In this case we know that $q_k \notin A \rightarrow z_2' \notin B \rightarrow z_1' \notin B \rightarrow x \in A$. Hence the simulation is contiued in the YES state.

vi. case 3 and $z_2' \preceq_f z_1'$. In this case we know that $q_k \in A \rightarrow z_2' \in B \rightarrow z_1' \in B \rightarrow x \notin A$. Hence the simulation is contiued in the NO state.

vii. case 4 and $z_1' \preceq_f z_2'$. In this case we know that $q_k \in A \rightarrow z_2' \notin B \rightarrow z_1' \notin B \rightarrow x \notin A$. Hence the simulation is contiued in the NO state.

viii. case 4 and $z_2' \preceq_f z_1'$. In this case we know that $q_k \notin A \rightarrow z_2' \in B \rightarrow z_1' \in B \rightarrow x \in A$. Hence the simulation is contiued in the YES state.

The rest of this part of the proof runs along the lines of Claim 4. Either the assumed answers to all queries produced by the simulation are corroborated by the oracle, or at least one is inconsistent with the oracle, and a conclusion about membership of x in A can immediately be drawn.

We now continue to show that the non-adaptive auto-reduction for A, say M_{auto} can, again with the help of the p-selector, be transformed into a single-query auto-reduction. Let $q_1, \ldots, q_{p(n)}$ be the queries produced by M_{auto} on input x. Let M_1 produce query x' on input x, and q_i' on input q_i. Since B is p-selective, we can produce a chain order from $q_1', \ldots, q_{p(n)}'$ and x', and prove, like in the proof of Theorem 3, that $\chi_A(x)$ can be concluded from $\chi_A(q_i)$, where q_i' is one of the (at most two) strings adjacent to x' in the chain order. The argument is only slightly more elaborate than the argument in Theorem 3, probably because the reduction is not necessarily positive. First we assume that $M_1^{\emptyset}(x) \neq M_1^{\{x'\}}(x)$. Otherwise, we can compute $\chi_A(x)$ without querying the oracle. Likewise, we may assume for each i that $M_1^{\emptyset}(q_i) \neq M_1^{\{q_i'\}}(q_i)$. Otherwise, there exists an equivalent truth-table auto-reduction where such a q_i is not a query. This time, we assume the queries q_i' (re)numbered in chain order and distinguish three cases:

1. $x' \preceq_f q_1'$. In this case $q_1' \notin B \rightarrow x' \notin B \rightarrow \chi_A(x) = M_1^{\emptyset}(x)$. On the other hand, if we know that $q_1' \in B$ then $\forall i(q_i' \in B)$. Then we can compute $\chi_A(q_1), \ldots, \chi_A(q_{p(n)})$ from this information and M_1, and if we know $\chi_A(q_1), \ldots, \chi_A(q_{p(n)})$ then we can compute $\chi_A(x)$. $\chi_B(q_1')$ in turn, can be computed from $\chi_A(q_1)$.

2. $q_{p(n)}' \preceq_f x'$. In this case $q_{p(n)}' \in B \rightarrow x' \in B \rightarrow \chi_A(x) = M_1^{\{x'\}}(x)$. On the other hand, if we know that $q_{p(n)}' \notin B$ then $\forall i(q_i' \notin B)$. Then we can compute $\chi_A(q_1), \ldots, \chi_A(q_{p(n)})$ and $\chi_A(x)$ can be computed as in

the previous case. We infer that in this case, $\chi_A(x)$ is derivable from $\chi_A(q_n)$.

3. $\exists i[q'_i \preceq x' \preceq q'_i]$. In this case $\chi_B(q'_i) = 1 \rightarrow x' \in B$, and $\chi_B(q'_{i+1}) = 0 \rightarrow x' \notin B$. Both cases provide enough information to compute $\chi_A(x)$. The only case left to examine, is the case where $\chi_B(q'_i) = 0$ and $\chi_B(q'_{i+1}) = 1$. In this case $\chi_B(q'_j) = 0$ for all $j \leq i$, and $\chi(q'_j) = 1$ for all $j \geq i+1$. Therefore, this assumption also fixes $\chi_A(x)$. We conclude, that we can certainly compute $\chi_A(x)$ in this case from $\chi_B(q'_i)$ and $\chi_B(q'_{i+1})$, and therefore from $\chi_A(q_i)$ and $\chi_A(q_{i+1})$. It remains to show that we can compute $\chi_A(x)$ from only one of these values. There remain four possible 'truth tables' on $\chi(q'_i)$ and $\chi_B(q'_{i+1})$ and outcome $\chi_A(x)$. We show, that all four of these degenerate to one query 'truth-tables' (and of course the value of the characteristic function for the remaining query can be computed from M_1 and the value of the corresponding string.) The four tables are as follows.

$\chi(q'_i)$	$\chi(q'_{i+1})$	$\chi(x)$	$\chi(q'_i)$	$\chi(q'_{i+1})$	$\chi(x)$
0	0	1	0	0	1
0	1	0	0	1	1
1	1	0	1	1	0

$\chi(q'_i)$	$\chi(q'_{i+1})$	$\chi(x)$	$\chi(q'_i)$	$\chi(q'_{i+1})$	$\chi(x)$
0	0	0	0	0	0
0	1	0	0	1	1
1	1	1	1	1	1

In each of the cases, for z equal to either q'_i or q'_{i+1}, either $x \in A$ iff $z \in B$, or $x \in A$ iff $z \notin B$.

This conclude the proof of the theorem. $\square$

Since $\leq^P_{1-tt}$-self-reducibility (as $\leq^P_m$-self-reducibility) for a set A implies membership of A in P, we derive the following corollary.

Corollary 12 *If A is self-reducible, B is p-selective and $A \leq^P_{1-tt} B$, then $A \in P$.*

The following theorem shows that Theorem 11 can probably not be generalized to Turing reductions.

Theorem 13 *If $E \neq UE$, then there exists sets D and B in $NP - P$ such that:*

1. *D is disjunctively self-reducible,*
2. *B is p-selective, and*
3. *$D \leq_T^P B$.*

Proof. The assumption $E \neq UE$ implies the existence of a tally set T in $UP - P$. Let R_T and $q(n)$ define T in the sense of Definition 3. Since $T \in UP - P$, there is for each $0^n \in T$ exactly one string w such that $R_T(0^n, w)$. We call this string *the* witness for 0^n.

Let $D = \text{PREFIX}(T) = \{<0^n, y> \mid y \text{ is a prefix of } w \text{ and } R_T(0^n, w)\}$. It is not hard to see that D is disjunctively self-reducible (cf. Selman [Sel88]). Also, $T \leq_m^P D$ since $0^n \in T$ iff $<0^n, \lambda> \in D$. Now as $T \in NP - P$ it follows that also $D \in NP - P$. Furthermore D is sparse since for each $0^n \in T$ there is only one witness of size $q(n)$ and there are only $q(n)$ prefixes of w. Since D is sparse there is a tally set T' in NP such that $D \leq_T^P T'$ (cf. [HIS85]). From this and the earlier cited result by Selman [Sel82] we infer the existence of a p-selective set B in $NP - P$ such that $T \leq_m^P D \leq_T^P T' \leq_T^P B$ □

Very recently, Beigel et al. [BKS94] have improved upon this Theorem, by showing that there exists a relativized world such that there exists a disjunctive self-reducible set A in $NP - P$ such that A is $\leq_{2-tt}^P$ -reducible to some p-selective set.

On the other hand, Buhrman et al. [BTB93] proved that any set that is positively Turing reducible to a p-selective set is itself p-selective. Therefore it follows directly that Theorem 11 also holds with 1-tt reductions replaced by positive Turing reducitons. Finally, we can infer from Theorem 11 the following corollaries.

Corollary 14 *EXP does not have $\leq_{1-tt}^P$-hard sets that are p-selective*

Proof. Suppose for a contradiction that EXP *does* have $\leq_{1-tt}^P$-hard sets that are p-selective then EXP has Turing hard sparse sets, so $EXP \subseteq P/poly$ and therefore $EXP \subseteq \Sigma_2^P$. On the other hand $SAT \in EXP$, so SAT is then $\leq_{1-tt}^P$ -reducible to some p-selective set, and then from Theorem 11 it follows that $P = NP$, the hierarchy collapses and $EXP \subseteq P$ which contradicts the time-hierarchy theorem. □

Corollary 15 *Let C be any of the following classes: $NP, PP, PSPACE$. Then $C = P$ if and only if there exists a $\leq_{1-tt}^P$-hard set for C that is p-selective.*

References

[ALM$^+$92] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedi. Proof verification and hardness of approximation problems. In *Proc. 33rd IEEE Symposium Foundations of Computer Science*, 1992.

[AS84] K. Ambos-Spies. p-mitotic sets. In E. Börger, G. Hasenjäger, and D. Roding, editors, *Logic and Machines, Lecture Notes in Computer Science 177*, pages 1–23. Springer-Verlag, 1984.

[BBFG91] R. Beigel, M. Bellare, J. Feigenbaum, and S. Goldwasser. Languages that are easier to verify than their proofs. In *Proc. 32nd IEEE Symposium on Foundations of Computer Science*, pages 19–28, 1991.

[BD76] A. Borodin and A. Demers. Some comments on functional self-reducibility and the NP hierarchy. Technical Report TR76-284, Cornell University, Department of Computer Science, Upson Hall, Ithaca, NY 14853, 1976.

[BDG88] J. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity I.* Springer-Verlag, 1988.

[BF92] R. Beigel and J. Feigenbaum. On being incoherent without being very hard. *Computational Complexity*, 2(1):1–17, October 1992.

[BFL90] L. Babai, L. Fortnow, and C. Lund. Non-deterministic exponential time has two-prover interactive protocols. In *Proc. 31st IEEE Symposium Foundations of Computer Science*, pages 16–25, 1990.

[BKS94] R. Beigel, M. Kummer, and F. Stephan. Approximable sets. In *Proc. Structure in Complexity Theory 9th annual conference*, pages 12–23, Amsterdam, Holland, 1994. IEEE computer society press.

[Boo74] R. Book. Tally languages and complexity classes. *Information and Control*, 26:186–193, 1974.

[BTB93] H. Buhrman, L. Torenvliet, and P. Van Emde Boas. Twenty questions to a p-selector. *Information Processing Letters*, 48(4):201–204, December 1993.

[Edm65] J. Edmonds. Paths, trees and flowers. *Canad. J. Math.*, 17:449–467, 1965.

[GHJY91] J. Goldsmith, L. Hemachandra, D. Joseph, and P. Young. Near-testable sets. *SIAM J. Comput.*, 20(3):506–523, 1991.

[GJY93] J. Goldsmith, D. Joseph, and P. Young. Using self-reducibilities to characterize polynomial time. *Information and Computation*, 104(2):288–308, 1993.

[HHO+92] L.A. Hemachandra, A. Hoene, M. Ogiwara, A.L. Selman, T. Thierauf, and J. Wang. Selectivity. Manuscript, 1992.

[HIS85] J. Hartmanis, N. Immerman, and V. Sewelson. Sparse sets in NP-P: EXPTIME versus NEXPTIME. *Information and Control*, 65(2/3):158–181, May/June 1985.

[Imm84] N. Immerman. Languages that capture complexity classes. *SIAM J. Comput.*, 16:760–778, 1984.

[Imm87] N. Immerman. Expressibility as a complexity measure: results and directions. In *Proc. Structure in Complexity Theory 2nd annual conference*, pages 194–202, Ithaca, NY, 1987. IEEE Computer Society Press.

[Joc68] C.G. Jockusch. Semirecursive sets and positive reducibility. *Trans. of the Amer. Math. Soc.*, 131:420–436, 1968.

[Ko83] K. Ko. On self-reducibility and weak P-selectivity. *J. Comput. System Sci.*, 26:209–211, 1983.

[Lad73] R. Ladner. Mitotic recursively enumerable sets. *J. Symbolic Logic*, 38(2):199–211, 1973.

[LLS75] R. Ladner, N. Lynch, and A. Selman. A comparison of polynomial time reducibilities. *Theoretical Computer Science*, 1:103–123, 1975.

[Moo68] J.W. Moon. *Topics on Tournaments.* Selected Topics in Mathematics. Holt, Rinehart and Winston, New York, 1968.

[MP79] A. Meyer and M. Paterson. With what frequency are apparently intractable problems difficult? Technical Report MIT/LCS/TM-126, M.I.T., 1979.

[Nai92] A.V. Naik. Personal communication. e-mail, 1992.

[NOS93] A.V. Naik, M. Ogihara, and A.L. Selman. P-selective sets, and reducing search to decision vs. self-reducibility. In *Proc. Structure in Complexity Theory 8th annual conference*, pages 52–64, San Diego, California, 1993. IEEE computer society press.

[Pip79] N. Pippenger. On simultaneous resource bounds. In *Proc. 20th IEEE Symposium on Foundations of Computer Science*, pages 307–311, 1979.

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

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

[Sel88] A. L. Selman. Promise problems complete for complexity classes. *Information and Computation*, 78(2):87–97, August 1988.

[Sel92] A.L. Selman. Personal communication. e-mail, 1992.

[Sha90] A. Shamir. IP=PSPACE. In *Proc. 31st IEEE Symposium Foundations of Computer Science*, pages 11–15, 1990.