Lower Bounds for the Low Hierarchy[1]

Eric Allender[2]   and   Lane A. Hemachandra[3]

Department of Computer Science   Department of Computer Science

Rutgers University   University of Rochester

New Brunswick, NJ 08903   Rochester, NY 14627

SUMMARY

The low hierarchy in NP [Sc-83] and the extended low hierarchy [BBS-86] have been useful in characterizing the complexity of certain interesting classes of sets. However, until now, there have been no results establishing whether a given lowness result is the best possible.

We prove absolute lower bounds on the location of classes in the extended low hierarchy, and relativized lower bounds on the location of classes in the low hierarchy in NP. In some cases, we are able to show that the classes are lower in the hierarchies than was known previously. In almost all cases, we are able to prove that our results are essentially optimal.

We also examine the interrelationships among the levels of the low hierarchies and the classes of sets reducible to or equivalent to sparse and tally sets under different notions of reducibility. We feel that these results clarify the structure underlying the low hierarchies.

1 Introduction

The low hierarchy within NP (with levels $\hat{L}_1$, L_1, $\hat{L}_2$, L_2, ...) was defined and studied by Schöning [Sc-83] as a way to classify the complexity of sets within NP that seem to encode less information than NP-complete sets. In essence, Schöning proves in [Sc-83] that no set in the low hierarchy can be complete under any reasonable notion of completeness, unless the polynomial hierarchy collapses. A number of researchers have examined the low hierarchy, and many interesting complexity classes have been shown to be low. Results of this sort will be surveyed later in this paper.

The low hierarchy, as defined in [Sc-83], can only be used to classify the complexity of sets in NP. In order to talk about related sets that are not in NP, the extended low hierarchy was introduced in [BBS-86]. (The levels of this hierarchy are labeled $\hat{EL}_2$, EL_2, $\hat{EL}_3$, EL_3, ...) Again, many interesting complexity classes have been shown to reside inside

the extended low hierarchy. (Definitions of the low and extended low hierarchies are given in Section 2.)

For example, it is shown in [BBS-86] that every sparse set is in EL_3, and it is shown in [KS-85] that every sparse set in NP is in $\hat{L}_2$. Also, it is shown in [KS-85] that every co-sparse set in NP is in $\hat{L}_3$. The question naturally arises as to whether these results are optimal; i.e., is there any real reason why the co-sparse sets in NP have not been shown to be as "low" as the sparse sets in NP? Do their complexities in the extended low hierarchy also differ?

Taking questions such as these as our starting point, we systematically examine many results concerning the low and extended low hierarchies. In some cases, we are able to improve known lowness results.

More importantly, we present the first theorems showing that various lowness results *cannot* be improved. For example, we show the existence of a sparse set that is not in EL_2, thus answering a question posed in [BB-86].

Although we are able to construct sets that are not in certain levels of the extended low hierarchy, we cannot hope to prove similar results for the low hierarchy within NP, unless we are first able to prove $P \neq NP$. Given that that is beyond current techniques, the best that we can do is to construct oracles relative to which certain classes are not contained in given levels of the low hierarchy.

Our results are summarized in Tables 1 and 2. As one can see, in most cases, we are able to present lowness results that are best possible (at least using relativizable techniques). The results in these tables show, for example, that the sparse sets in NP do seem to have different lowness properties from the co-sparse sets in NP, and that it is very unlikely that every set in P/poly can be shown to have the same lowness properties as the sets in RP have.

We find that, in trying to understand the structure of the low hierarchies, it is helpful to study the lowness properties of the classes of sets reducible to or equivalent to sparse and tally sets under various notions of reducibility. These results are summarized in Tables 3 and 4.

Finally, we note that all oracles constructed in this paper are recursive.

2 Definitions

We assume that the reader is familiar with fundamental notions such as NP and the polynomial hierarchy.

Following [KS-85], for all $k \geq 0$ we define L_k to be the class of sets L in NP such

that $\Sigma_k^{p,L} = \Sigma_k^p$, and $\hat{L}_k$ is the class of sets L such that $\Delta_k^{p,L} = \Delta_k^p$. Thus the sets in the low hierarchy are those sets in NP that provide no additional power to some level of the polynomial hierarchy, when given as an oracle. Some basic properties of this hierarchy are that $\hat{L}_0 = L_0 = \hat{L}_1 = P$, and $L_1 = NP \cap coNP$. Also, for all k, $L_k \subseteq \hat{L}_{k+1} \subseteq L_{k+1}$.

Only sets in NP can appear in the low hierarchy. To help classify sets that are not in NP, the extended low hierarchy was defined in [BBS-86] as follows: For all $k \geq 1$, EL_k is the class of sets L such that $\Sigma_k^{p,L} \subseteq \Sigma_{k-1}^{p,L \oplus SAT}$, where $A \oplus B = \{0x \mid x \in A\} \cup \{1x \mid x \in B\}$. Similarly, one can define intermediate levels (although to our knowledge this has not been done in earlier work on the extended low hierarchy). Let $\hat{EL}_k$ denote the class of sets such that $\Delta_k^{p,L} \subseteq \Delta_{k-1}^{p,L \oplus SAT}$. This gives rise to the following observations:

Proposition 1

1. Every subset of $\{0,1\}^*$ is in $\hat{EL}_1$.

2. $EL_1 = \hat{EL}_2$.

3. For all $k \geq 2$, $EL_k \cap NP = L_k$[BBS-86].

4. For all $k \geq 2$, $\hat{EL}_k \cap NP = \hat{L}_k$.

Proof: To prove the first statement, it suffices to note that, for every set L, $\Delta_1^{p,L} = P^L \subseteq P^{L \oplus SAT} = \Delta_{1-1}^{p,L \oplus SAT}$.

To prove the second statement, observe that $L \in EL_1 \iff NP^L \subseteq P^{L \oplus SAT} \iff P^{NP^L} \subseteq P^{L \oplus SAT} \iff L \in \hat{EL}_2$.

Statement four follows since, if L is in NP and $k \geq 2$, then $L \in \hat{EL}_k \iff P^{\Sigma_{k-1}^{p,L}} \subseteq P^{\Sigma_{k-2}^{p,L \oplus SAT}} \iff P^{\Sigma_{k-1}^{p,L}} \subseteq P^{\Sigma_{k-2}^{p,SAT}} \iff \Delta_k^{p,L} = \Delta_k^p \iff L \in \hat{L}_k$. ∎

Because the classes EL_1 and $\hat{EL}_1$ behave anomalously, we will consider for the rest of this paper that the extended low hierarchy begins at level $\hat{EL}_2$.

The following theorem illustrates some of the closure properties of the levels of the low hierarchies. Let us say that a class $\mathcal{C}$ is *closed* under relation R if for all sets A and B, (ARB and $B \in \mathcal{C} \implies A \in \mathcal{C}$).

Theorem 2

1. Every level of the extended low hierarchy is closed under $\equiv_T^p$.

2. $\hat{EL}_2$ and EL_2 are not closed under $\leq_m^p$.

3. If the polynomial hierarchy is infinite, then no level of the extended low hierarchy is closed under $\leq_m^p$.

4. Every level of the low hierarchy is closed under $\leq_m^p$.

3

5. If $A \leq^p_T B$ and $A \in \mathrm{NP}$ and $B \in \mathrm{L}_k$ ($\hat{\mathrm{L}}_k$), then $A \in \mathrm{L}_k$ ($\hat{\mathrm{L}}_k$). (Thus every level of the low hierarchy is closed under $\leq^p_T$, when only sets in NP are considered.)

Proof: Part 1: Let L be in EL_k and let $A \equiv^p_T L$. Then $\Sigma^{p,A}_k \subseteq \Sigma^{p,L}_k \subseteq \Sigma^{p,L\oplus\mathrm{SAT}}_{k-1} \subseteq \Sigma^{p,A\oplus\mathrm{SAT}}_{k-1}$; thus EL_k is closed under $\equiv^p_T$. The proof that $\hat{\mathrm{EL}}_k$ is also closed under $\equiv^p_T$ is very similar.

Part 2: From the proof of Theorem 4, later in this paper, one can see that there is a sparse set S in $\mathrm{DSPACE}(2^n)$ that is not in EL_2. On the other hand, note that if A is any set that is $\leq^p_m$-complete for $\mathrm{DSPACE}(2^{n^{O(1)}})$, then $\mathrm{NP}^A = \mathrm{DSPACE}(2^{n^{O(1)}})$, and thus $\mathrm{P}^{\mathrm{NP}^A} = \mathrm{DSPACE}(2^{n^{O(1)}}) = \mathrm{P}^{A\oplus\mathrm{SAT}}$, and thus A is in $\hat{\mathrm{EL}}_2$. Part 2 follows since $S \leq^p_m A$, $A \in \hat{\mathrm{EL}}_2$, and $S \notin \mathrm{EL}_2$.

Part 3: It was observed in [BBS-86] that any set $\leq^p_m$-complete for PSPACE (such as QBF) is in $\hat{\mathrm{EL}}_2$, and thus QBF is in every level of the extended low hierarchy. If some level EL_k of the extended low hierarchy is closed under $\leq^p_m$, then SAT is in EL_k (since SAT $\leq^p_m$ QBF $\in \mathrm{EL}_k$) and thus SAT $\in \mathrm{EL}_k \cap \mathrm{NP} = \mathrm{L}_k$ and hence the polynomial hierarchy collapses, by [Sc-83].

Note that part 5 implies part 4, so we will prove only part 5. Let $A \leq^p_T B$ and $A \in \mathrm{NP}$ and $B \in \mathrm{L}_k$. Then $\Sigma^{p,A}_k \subseteq \Sigma^{p,B}_k = \Sigma^p_k$, and thus $A \in \mathrm{L}_k$. The proof for $\hat{\mathrm{L}}_k$ is equally trivial. ∎

It would be desirable to prove that no level of the extended low hierarchy is closed under $\leq^p_m$. The techniques of [Ko-89a, Ko-89b] may help in proving this.

A set S is a *tally set* if $S \subseteq 0^*$ (see [Bo-74]). A set S is *sparse* if there is a polynomial $p(\cdot)$ such that for all n, there are at most $p(n)$ strings in S of length at most n (see [Ma-86]).

Corollary 3 Every set in NP that is $\leq^p_m$ reducible to a sparse set is in $\hat{\mathrm{L}}_2$.

Proof: Let A be a set in NP such that $A \leq^p_m S_1$ for some sparse set S_1. Let f be the many-one reduction that witnesses $A \leq^p_m S_1$. We can assume, without loss of generality, that f is "honest," i.e., there is some polynomial p such that, for all x, $p(|f(x)|) \geq |x|$. Let $S_2 = \{y : \exists x \in A\ f(x) = y\}$. It is easy to check that S_2 is sparse and in NP and $A \leq^p_m S_2$. The result now follows from Theorem 2 and the fact that all sparse sets in NP are in $\hat{\mathrm{L}}_2$ [KS-85]. ∎

The structure of the low hierarchies can be better understood by examining the lowness properties of classes of sets equivalent to or reducible to sparse or tally sets under various sorts of reducibility. The rest of this section presents definitions that are necessary in order to do this in a systematic way.

Let r be a type of reducibility, such as $\leq^p_m$, $\leq^p_T$, $\leq^p_{btt}$, etc. (For definitions of these notions of reducibility, see [LLS-75].) Following the definitions of [BK-88], define $P_r(SPARSE)$ to

be the class of sets L such that there is some sparse set S such that $L \leq_r^p S$. Similarly, define $P_r(TALLY)$ to be the class of sets L such that there is some tally set T such that $S \leq_r^p T$. The properties of the classes $P_r(TALLY)$ and $P_r(SPARSE)$ are studied in [BK-88, Ko-88]. One motivation for studying classes of this sort comes from interest in P/poly, which is the class of sets that have polynomial-sized circuits. It is shown in [BK-88, Sc-85] that P/poly $= P_T(SPARSE) = P_{tt}(SPARSE) = P_T(TALLY) = P_{tt}(TALLY)$.

Somewhat different structures emerge when one considers the classes of sets that are *equivalent* to sparse or tally sets under these notions of reduction. Such a study was begun in [TB-88]. Following their definitions, define $E_r(SPARSE)$ ($E_r(TALLY)$) to be the class of sets L such that there exists some sparse (tally) set S such that $L \leq_r^p S$ and $S \leq_r^p L$. In addition to [TB-88], results concerning these classes may be found in [AW-88].

Our paper makes brief mention of the classes APT ("almost polynomial time") [MP-79] and P-close [Sc-86]. The only facts we will need concerning these classes are (1) every sparse set is in APT $\cap$ P-close, and (2) every set in APT $\cup$ P-close is $\leq_{1-tt}^p$-reducible to a sparse set.

Our proofs in this paper will make use of a standard enumeration of clocked nondeterministic polynomial-time oracle machines $M_1, M_2, \ldots$, where the running time of M_j is no more than $n^j + j$. Similarly, we will use $P_1, P_2, \ldots$ to refer to a standard enumeration of clocked deterministic polynomial-time oracle machines.

3 Results on Extended Lowness

Theorem 4 There is a sparse set that is not in EL_2.

Proof: Our goal is to construct a sparse set S such that $L = \{x : |x| = n, \exists y \in \Sigma^n \forall z \in \Sigma^n \ 0xyz \notin S$ and $1xy0^n \in S\}$ is not in $NP^{S \oplus SAT}$ (note that L *is* in NP^{NP^S}); thus S is not in EL_2.

S is constructed according to a stage construction. At state j, we will guarantee that the jth machine M_j does not accept L with oracle $S \oplus SAT$.

To begin the construction, let $S_0 = \emptyset$, and set $n_0 = 1$. At stage j, let S_{j-1} be the part of oracle S that has been constructed in earlier stages. (At each stage j, S_{j-1} will be finite.) Let $n = n_j = n_{j-1}^j + j$. (This will guarantee that n is large enough to avoid interference with earlier stages.) Run M_j on input 0^n with oracle $S_{j-1} \oplus SAT$.

If there is an accepting computation, then let $S_j = S_{j-1}$: note that in this case 0^n is not in L, but is accepted by M_j.

If, on the other hand, there is not an accepting computation, then if there is a y such that M_j rejects 0^n with oracle $(S_{j-1} \cup \{10^n y 0^n\}) \oplus SAT$, then let $S_j = S_{j-1} \cup \{10^n y 0^n\}$.

5

Note that in this case 0^n is in L, but is rejected by M_j.

If there is no such y, then choose some accepting path of M_j on input 0^n with oracle $(S_{j-1} \cup \{10^n0^n0^n\}) \oplus SAT$. Along this path, there is some z such that 00^n0^nz is not queried. Let $S_j = S_{j-1} \cup \{10^n0^n0^n, 00^n0^nz\}$. In this case, 0^n is not in L, but is accepted by M_j.

That completes the construction of stage j. The set S is $\bigcup_j S_j$. Note that, at each stage, at most 2 strings are placed into S, and thus S has at most two strings of any length n. Thus S is sparse. ∎

Corollary 5

- co-SPARSE $\not\subseteq$ EL$_2$.

- APT $\not\subseteq$ EL$_2$.

- P-close $\not\subseteq$ EL$_2$.

- P/poly $\not\subseteq$ EL$_2$.

- $E_r(SPARSE) \not\subseteq$ EL$_2$, for any standard reducibility $\leq_r^p$.

- $P_r(SPARSE) \not\subseteq$ EL$_2$, for any standard reducibility $\leq_r^p$.

In trying to prove best possible results on lowness and extended lowness, we were led to make improvements on previously known lowness results concerning sparse sets and left cuts of real numbers. The rest of this section deals with these improvements.

It was shown in [BBS-86] that every sparse set is in EL$_3$; we improve this a half-level and show that SPARSE $\subseteq$ $\hat{\text{EL}}_3$. (The half-levels of the extended low hierarchy were not considered by [BBS-86].)

Theorem 6 $P_{1-tt}(SPARSE) \subseteq \hat{\text{EL}}_3$.

Proof: Let $L \in P_{1-tt}(SPARSE)$. To show that $L \in \hat{\text{EL}}_3$, it will suffice to show that $\text{NP}^{\text{NP}^L} \subseteq \text{P}^{\text{NP}^{L \oplus SAT}}$.

We may assume without loss of generality that $L \leq_{1-tt}^p S$ for some sparse set S via a $\leq_{1-tt}^p$ reduction f having the property that, for all x, $f(x)$ is either ACCEPT or REJECT (indicating that x should be accepted or rejected) or $f(x)$ is of the form $\langle y, b \rangle$, where $b \in \{T, F\}$, $|y| > |x|$, and x is accepted iff $b = $ the truth value of the statement "$y \in S$".

Let $S' = \{y \in S : (\exists x \in L \ f(x) = \langle y, T \rangle) \vee (\exists x \in \overline{L} \ f(x) = \langle y, F \rangle)\}$; intuitively, S' contains those elements of S that are actually tested by f. Then $L \leq_{1-tt}^p S'$ and $S' \in \text{NP}^L$. Note also that the set $pref(S') = \{\langle z, 0^n \rangle : \exists w \ |zw| = n \wedge zw \in S'\}$ is in NP^L.

Let $A \in \text{NP}^{\text{NP}^L}$. That is, there exists a set B and there exist i, j such that A is the language accepted by M_i^B and B is the language accepted by M_j^L. Let $A' = \{\langle x, t \rangle : t$ is

6

the encoding of a finite set and x is accepted by M_i with oracle the set accepted by M_j^t}. Then $A' \in NP^{NP}$, and if t is an encoding of the set of strings in S of length $\leq (|x|^i + i)^j + j$, then $x \in A \iff \langle x, t \rangle \in A'$.

Our $P^{NP^{L \oplus SAT}}$ algorithm for recognizing A is as follows. On input x, use repeated calls to the oracle for $pref(S')(\in NP^L \subseteq NP^{L \oplus SAT})$ to construct an encoding t of strings in A of length $\leq (|x|^i + i)^j + j$, and then use the oracle for $A'(\in NP^{NP} \subseteq NP^{L \oplus SAT})$ to determine if $\langle x, t \rangle \in A'$ ($\iff x \in A$). ∎

Corollary 7

- $P_m(SPARSE) \subseteq \hat{EL}_3$.

- $E_T(SPARSE) \subseteq \hat{EL}_3$.

- SPARSE $\subseteq \hat{EL}_3$.

- co-SPARSE $\subseteq \hat{EL}_3$.

- APT $\subseteq \hat{EL}_3$.

- P-close $\subseteq \hat{EL}_3$.

- $E_T(SPARSE) \cap NP \subseteq \hat{L}_3$.

- $P_{1-tt}(SPARSE) \cap NP \subseteq \hat{L}_3$.

- co-SPARSE$\cap$NP $\subseteq \hat{L}_3$. [KS-85]

- P-close$\cap$NP $\subseteq \hat{L}_3$. [Sc-85]

The following result should be compared with Theorem 6.

Theorem 8 $P_{btt}(TALLY)$ is contained in $\hat{EL}_2$.

Proof: It was shown in [BK-88] that $P_{btt}(TALLY) = P_m(TALLY)$, so it will suffice to show that if there is a tally set T and a $\leq_m^p$ reduction f such that $f : L \leq_m^p T$, then $NP^L \in P^{SAT \oplus L}$.

We may assume without loss of generality that f is an honest reduction. Let $T' = \{0^n : \exists x \in L \ f(x) = 0^n\}$. Then $L \leq_m^p T'$ and $T' \in P^{SAT \oplus L}$.

Let $A \in NP^L$; so there is a machine M_i that accepts A with oracle L. Let $A' = \{\langle x, t \rangle : t$ is the encoding of a finite tally set, and M_i accepts x with oracle $t\}$. Note that if t is an encoding of the elements of T of length $\leq |x|^i + i$, then $x \in A \iff \langle x, t \rangle \in A'$. Note also that $A' \in NP$.

7

Our $P^{SAT \oplus L}$ for recognizing A is as follows. On input x, construct t—an encoding of the strings in A of length $\leq |x|^i + i$—by determining if $0^j \in T'$ for $0 \leq j \leq |x|^i + i$. Then accept $x \iff \langle x, t \rangle \in A'$ (which can be determined by one call to SAT). ∎

Corollary 9 $E_T(TALLY) \subseteq \hat{E}L_2$. [BB-86]

Left cuts of real numbers were introduced into complexity theory by Ko [Ko-82]. Ko defined *standard* left cuts and *general* left cuts. Lowness results are known for both. Selman showed that every Standard Left Cut is in $E_T(TALLY)$ [Se-81]; we improve this below to show that every General Left Cut is in $E_T(TALLY)$. It was shown in [KS-85] that every (General) Left Cut in NP is in L_2. (Ko and Schöning *did* consider half-levels of the low hierarchy in NP.) Using Corollary 9 we improve the result of [KS-85] to show that every General Left Cut in NP is in $\hat{L}_2$.

We will present only the definition for general left cuts. For the purposes of this paper, it suffices to note that every standard left cut is also a general left cut.

Definition: Let $E = (+|-)(0|1)^+.(0|1)^*$. To every string $x \in E$ we associate the rational number (which we also denote by x) whose binary encoding is given by x. (For more detail, see [Ko-86].) For any real number y, the *general left cut* of y is the set of all strings $x \in E$ such that $x - y \leq 2^{|z|}$, where z is the suffix of x appearing after the decimal point.

Theorem 10 If L is a General Left Cut, then L is in $E_T(TALLY)$.

Proof: Let L be a general left cut. Note that for all lengths n there is a string (which we shall call max_n) such that for all $x.z \in (\Sigma^*.\Sigma^n \cap E), x.z \in L \iff x.z \leq max_n$.

Let T be the set $\{1^n 0^i :$ the ith bit of the fractional part of max_n is $1\}$. T is easily seen to be in $E_T(TALLY)$. It is also easy to see that $T \equiv_T^p L$. ∎

Corollary 11

- Every General Left Cut is in $\hat{E}L_2$.

- Every Standard Left Cut is in $\hat{E}L_2$. [Se-81]

- Every Standard Left Cut in NP is in $\hat{L}_2$.

- Every General Left Cut in NP is in $\hat{L}_2$.

4 Relativizations

In this section we present oracle constructions showing that known lowness results concerning sets in NP cannot be improved using any proof techniques that relativize; hence the known lowness results are optimal in some sense.

Baker, Gill, and Solovay [BGS-75] presented an oracle relative to which there is a tally set in NP−coNP. Since L_1 = NP∩coNP, an immediate corollary is that there is an oracle relative to which there is a tally set (hence also a sparse set in APT) in NP that is not in L_1. Since Ko showed in [Ko-83] that if there is a tally set in NP−coNP then there is a standard left cut in NP−coNP, it follows that the [BGS-75] relativization also shows the lowness results for Left Cuts presented in the previous section cannot be improved.

In order to present our other relativization results, we must say a few words on how the levels of the relativized low hierarchy are defined. We define L_k^A to be the class of sets L in NP^A such that $\Sigma_k^{p,L \oplus A} = \Sigma_k^{p,A}$, and $\hat{L}_k^A$ is the class of sets L such that $\Delta_k^{p,L \oplus A} = \Delta_k^{p,A}$.

All sets in the probabilistic complexity class RP are in L_2 [KS-85]; thus, the following result is essentially optimal with respect to relativizable techniques.

Theorem 12 There is an oracle A relative to which RP is not contained in $\hat{L}_2$.

Proof: We present an oracle A such that the set $L = \{x : \exists y \; |y| = 2|x| \text{ and } xy \in A\}$ is in RP^A, and such that the set $B = \{0^n : \exists x \in \Sigma^n \; x \notin L\}$ is not contained in P^{NP^A}. (Note that B is clearly in NP^L; thus L is not in $\hat{L}_2^A$, since $P^{NP^{L \oplus A}}$ is not contained in P^{NP^A}.)

We use a stage construction. In stage $\langle i, j \rangle$ we guarantee that for all deterministic machines P_i and for all nondeterministic machines M_j, $P_i^{M_j^A}$ does not recognize B, while maintaining the condition that $L \in RP^A$.

Begin the construction in stage 0 with $A = \emptyset$ and $n_0 = 1$. The following lines explain the construction of stage l, where $l = \langle i, j \rangle$. In this proof, and in all later "stage" constructions in this paper, we always use the current version of our evolving oracle A in simulations, and always refer to the final oracle A when speaking of reserving strings in A or $\bar{A}$.

Stage $l = \langle i, j \rangle$: Choose n so that $n > (n_{l-1} + i + j)^{i+j}$ and $2^n > (n^i + i)(n + i + j)^{i+j}$, and set $n_l \leftarrow n$.

Begin a simulation of P_i on input 0^n with oracle M_j^A. Let w_1 be the first query asked by P_i in this computation. If there is a set S of strings of length at least n such that $M_j^{A \cup S}$ accepts w_1, then let S be such a set of minimal size. (Note that S has size at most $|w_1|^j + j$, since that is the largest number of queries that can be asked along any single computation path of M_j.) Choose one accepting path p of M_j on input w_1 with oracle $A \cup S$. Place all the strings in S into A, and reserve for $\bar{A}$ each string that is queried on path p and not in S

and not already in A. Note that at most $|w_1|^j + j \le (n^i + i)^j + j \le (n + i + j)^{i+j}$ strings are reserved in this way. (We will call this process *reserving an accepting path*. Note that as long as we add no strings to A that have been reserved for $\overline{A}$, M_j will accept w_1 with oracle A.) Also note that if there is no such S, then M_j will reject w_1 with oracle A no matter how we modify oracle A, as long as we place in A no strings that have been reserved for $\overline{A}$.

Now continue the simulation of P_i until another query w_2 to M_j^A is encountered, and repeat the process of the preceding paragraph with string w_2. Continue this process for all queries asked by P_i on input 0^n. Note that membership for at most $(n^i + i)(n + i + j)^{i+j}$ strings is decided in this way.

When the simulation of P_i finally halts, there are two cases:

Case 1: $P_i^{M_j^A}$ accepts 0^n. In this case place into A all strings of length $3n$ that have not been reserved for $\overline{A}$. It is easy to verify that for all $x \in \Sigma^n$ there are at least 2^{2n-1} strings $y \in \Sigma^{2n}$ such that $xy \in A$. Thus every string in Σ^n is in L, and thus $0^n \notin B$, which was the condition that needed to be satisfied. Furthermore, we have maintained the straightforward RP^A membership test for L.

Case 2: $P_i^{M_j^A}$ rejects 0^n. In this case, let $X = \{x_1, x_2, \ldots, x_r\}$ be the elements x of Σ^n for which there is some y of length $2|x|$ such that xy has been placed into A. For each $x \in X$, there are at least $2^{2|x|} - (n^i + i)(n + i + j)^{i+j} > 2^{2|x|-1}$ strings $y \in \Sigma^{2|x|}$ such that xy has *not* been reserved for $\overline{A}$. Place each such string xy into A. Thus we have maintained the straightforward RP^A membership test for L. Furthermore, since $r < (n^i + i)(n + i + j)^{i+j} < 2^n$, there are strings $x \in \Sigma^n - X$ and for such strings x there is no $y \in \Sigma^{2|x|}$ such that xy is in A. Thus $0^n \in B$, which guarantees that $P_i^{M_j^A}$ does not recognize B. ∎

Corollary 13

- There is an oracle relative to which $BPP \cap NP$ is not contained in $\hat{L}_2$.

- There is an oracle relative to which $coAM \cap NP$ is not contained in $\hat{L}_2$.

The next two results concern P-selective and P-cheatable sets. P-selective sets were defined by Selman in [Se-79] and have been studied since then in [Se-81, KS-85] and numerous other papers. It was shown in [KS-85] that every P-selective set in NP is in L_2, and in [ABG-89] it was shown that all P-selective sets are in EL_2.

P-cheatable sets have been studied by several authors in the last few years (see [Be-87, AG-87, GJY-87, ABG-89]). In [ABG-89], it was shown that the P-cheatable sets are in EL_2.

We do not know if there are any P-cheatable or P-selective sets that are not in $E\hat{L}_2$. However, the following two theorems show that relativizable proof techniques cannot show

that every set that is P-selective or P-cheatable is in $\hat{\mathrm{EL}}_2$. First, we present the definitions for P-selective and P-cheatable sets; since we need to refer to these notions relative to oracles, we will define P^A-selectivity and P^A-cheatability for any oracle A.

Definitions: A set L is P^A-*selective* if there is a function f computable in polynomial time relative to A such that for all x and y, $f(\langle x,y\rangle) \in \{x,y\}$ and $\{x,y\} \cap L \neq \emptyset \implies f(\langle x,y\rangle) \in L$. L is P^A-*1-cheatable* if there is a machine running in polynomial time relative to oracle $A \oplus L$ that, on input $\langle x,y\rangle$, outputs $\langle \chi_L(x), \chi_L(y)\rangle$, and makes at most one query to the L part of its oracle. (Here, χ_L denotes the characteristic function of L.)

Any set that is P^A-1-cheatable is called P^A-*cheatable*, as are sets satisfying certain more general notions of cheatability (see [ABG-89, GJY-87]). In this paper we need nothing beyond 1-cheatability.

Theorem 14 There is an oracle A relative to which there is a set in NP that is P-selective and P-cheatable and is not in $\hat{\mathrm{L}}_2$.

Proof: In some ways this oracle construction is similar to the preceding one, except that we must build extra information into the oracle A in order to ensure the set L is P^A-selective and P^A-1-cheatable.

Let us say that A is *legal for* L if A and L satisfy the following five requirements:

- $x \in L \iff \exists y \; |y| = |x|$ and $0xy \in A$ (1)

- for all x and y such that $x < y$, exactly one of the strings

$$11\langle x,y,x\rangle, 11\langle x,y,y\rangle, 10\langle x,y,x\rangle, 10\langle x,y,y\rangle$$

  is in A (2)

- for all x,y,u,v such that $x < y$ and $\{x,y\} = \{u,v\}$, if $1b\langle x,y,u\rangle$ is in A, then

  - $(u \in L \implies v \in L)$ (3)
  - $(u \notin L \implies (v \in L \iff b = 1))$ (4)

- If $|x| < |y|$, then $10\langle x,y,x\rangle \notin A$ (5)

We will say that A is *legal* if there exists a set L such that A is legal for L. Note that for *any* set L there exist infinitely many sets A such that A is legal for L. Note also that if A is legal for L, then L is in NP^A, by condition (1). Conditions (2), (3), and (4) guarantee P-selectivity and P-cheatability, as we show below. Condition (5) is a technical condition that simplifies the proof.

First we claim that if A is legal for L, then L is P^A-1-cheatable. To see this, consider the following routine:

On input $\langle w, z \rangle$, let $x = \min(w, z)$ and $y = \max(w, z)$.

If $w = z$ then $\chi_L(w) = \chi_L(z)$, so one query to L suffices. Otherwise:

Make at most four queries to A to find for which $u \in \{x, y\}$ and $b \in \{0, 1\}$ is $1b\langle x, y, u \rangle$ in A.

Let v denote the string in $\{x, y\} - \{u\}$.

If $u \in L$ (which can be determined by making a query to L),

> Then $\chi_L(u) = 1$ and $\chi_L(v) = 1$
>
> Else $\chi_L(u) = 0$ and $\chi_L(v) = b$

Next we claim that if A is legal for L, then L is P^A-selective. To see this, consider the function f computed by the following routine:

On input $\langle w, z \rangle$, let $x = \min(w, z)$ and $y = \max(w, z)$.

If $w = z$ then output w. Otherwise:

Make at most four queries to A to find for which $u \in \{x, y\}$ and $b \in \{0, 1\}$ is $1b\langle x, y, u \rangle$ in A.

Let v denote the string in $\{x, y\} - \{u\}$.

Output v.

It is easy to verify that f satisfies the conditions required for L to be P^A-selective.

Let $B = \{0^n : \forall x \in \Sigma^n \; x \in L\}$. Clearly, B is in $\mathrm{coNP}^L \subseteq \mathrm{P}^{\mathrm{NP}^{L \oplus A}}$. We will build A in such a way that B is not in $\mathrm{P}^{\mathrm{NP}^A}$, thus showing that the P^A-selective and P^A-1-cheatable set L is not in $\hat{\mathrm{L}}_2^A$.

As before, in stage $\langle i, j \rangle$ we guarantee that for all deterministic machines P_i and for all nondeterministic machines M_j, $\mathrm{P}_i^{\mathrm{M}_j^A}$ does not recognize B, while maintaining the condition that A is legal for L.

At the beginning of each stage, there will be finite sets of strings that are reserved for A and for $\overline{A}$, respectively. As each stage progresses, more strings will be reserved for A and $\overline{A}$. Reserving a string for A or $\overline{A}$ may force other strings to be reserved for A or $\overline{A}$ (and thus, indirectly, for L) in order to maintain legality. For example, if we reserve $0xy$ for A, where $|x| = |y|$, then x is in L. (We will say in this case that x is *reserved for L*.) Also, if there is some string $z > x$ such that $10\langle x, z, x \rangle$ is reserved for A, then z is also reserved for L; this means that some string $0zv$ must be reserved for A, for some v of the same length as z. The consequences of reserving a string can be complex. In order to minimize this complexity, we will take pains to ensure that we avoid reserving strings for $\overline{L}$.

Begin the construction in stage 0 with $A = \emptyset$ and $n_0 = 1$. (We will maintain the condition that no string of length greater than n_l is reserved by stage l.) The following lines

12

explain the construction of stage l, where $l = \langle i, j \rangle$.

Stage $l = \langle i, j \rangle$: Choose n so that $n > (n_{l-1} + i + j)^{i+j}$ and $2^n > (n^i + i)(n + i + j)^{i+j}$, and set $n_l \leftarrow (n + i + j)^{i+j}$.

The first step of this stage is to reserve for A all strings in $0\Sigma^r$ such that $n_{l-1} \leq r < n_l$ and $r \neq 2n$. In order to maintain the legality of A, the following strings must also be reserved:

> for all $x < y$ where either $\chi_L(x)$ or $\chi_L(y)$ is determined (that is, for all $x < y$ with $2|y| + 1 \leq n_l$ such that it is *not* the case that $|x| = |y| = n$), and such that $\chi_L(y)$ was not determined in stage $l - 1$:
>> (Note that the last condition implies $\chi_L(y) = 1$.)
>> reserve $11\langle x, y, x \rangle$ for A, and
>> reserve $11\langle x, y, y \rangle, 10\langle x, y, x \rangle, 10\langle x, y, y \rangle$ for $\overline{A}$.

Next, reserve for A all strings w of the form $1b\langle x, y, z \rangle$ where either $z \notin \{x, y\}$ or $x \geq y$, such that $|w| \leq n_l$.

It may now be verified that the only strings of length $\leq n_l$ that are *not* reserved are strings of the form $0xy$ where $|x| = |y| = n$, and strings of the form $1b\langle x, y, z \rangle$ where $x < y$ and $z \in \{x, y\}$ and ($|x| = n$ or $|y| = n$).

Furthermore, A has the property that for every subset S of Σ^n, there a set L with $L \cap \Sigma^n = S$ such that there is an extension of A that is legal for L.

Next, begin a simulation of P_i on input 0^n with oracle M_j^A. Let w_1 be the first query asked by P_i in this computation. If there is a set S that is a legal extension of A such that M_j^S accepts w_1, then choose one accepting path p of M_j on input w_1 with oracle S, and let Q be the set of strings that are queried along this path. Reserve all of the strings in $Q \cap S$ for A, and reserve for $\overline{A}$ all of the strings in $Q \cap \overline{S}$; in order to maintain the legality of A, it may be necessary to reserve other strings at this time too. Note that the size of Q is at most $|w_1|^j + j \leq (n + i + j)^{i+j}$. (If there is no such legal extension S, reserve no strings for A or $\overline{A}$ at this step.)

Now continue the simulation of P_i until another query w_2 to M_j^A is encountered, and repeat the process of the preceding paragraph with string w_2. Continue this process for all queries asked by P_i on input 0^n. Note that at most $(n^i + i)(n + i + j)^{i+j}$ queries to the oracle A are processed in this way.

In order to proceed, we will need the following lemma:

Lemma 15 During the simulation phase of stage $\langle i, j \rangle$, if r queries to oracle A are processed, then at most r strings in $0\Sigma^{2n}$ are reserved for A or $\overline{A}$, and at most r strings are reserved for L, and no string is reserved for $\overline{L}$ during stage $\langle i, j \rangle$.

Let us assume for now that the lemma is true, and present the rest of the construction. When the simulation of P_i on input 0^n is finally completed, there are two cases:

Case 1: $P_i^{M_j^A}$ accepts 0^n. In this case let x be a string of length n that has not been reserved for L. (Such a string x must exist, by Lemma 15, since $2^n > (n^i + i)(n + i + j)^{i+j}$.) Since x is not reserved for L, that means that there is some legal extension of A that keeps x out of L. Extend A in this way; note that no string of length greater than n_l needs to be reserved for A or for $\overline{A}$ to guarantee that $x \notin L$. Now observe that $0^n \notin B$, and thus $P_i^{M_j^A}$ does not accept B.

Case 2: $P_i^{M_j^A}$ rejects 0^n. In this case, since no string of length n is reserved for $\overline{L}$, there is a legal extension of A such that L contains every string in Σ^n. Extend A in this way; as before, no string of length greater than n_l needs to be reserved for A or for $\overline{A}$ to guarantee this. Now $0^n \in B$, and thus $P_i^{M_j^A}$ does not accept B.

Now it remains only to prove Lemma 15. We will prove, by induction on r, $0 \leq r \leq (n^i + i)(n + i + j)^{i+j}$, that after the rth query to oracle A is processed, at most r strings in $0\Sigma^{2n}$ are reserved for A or $\overline{A}$, and at most r strings are reserved for L, and no string is reserved for $\overline{L}$ during the simulation.

The proof will proceed by showing that each reserved string can be "charged" to some query; we will show that no individual query is charged for too many reserved strings.

The basis $r = 0$ is trivially true; thus consider what happens when the rth string w is reserved. There are several cases:

Case 1: w is reserved for A, and $w = 0xy$ where $|x| = |y|$. This reserves x for L; charge x to w.

Note that $|x| = n$. For each string z that is reserved for L, let $u = \min(x, z)$ and $v = \max(x, z)$. Reserve $11\langle u, v, v \rangle$ for A, and reserve $11\langle u, v, u \rangle, 10\langle u, v, v \rangle, 10\langle u, v, u \rangle$ for $\overline{A}$.

For each string z that is reserved for $\overline{L}$ (any such z must have length less than n_{l-1}), reserve $11\langle z, x, z \rangle$ for A, and reserve $11\langle z, x, x \rangle, 10\langle z, x, z \rangle, 10\langle z, x, x \rangle$ for $\overline{A}$.

If there is any string v such that v is not currently reserved for L, and there is some string u of the form $1b\langle u, x, x \rangle$ or $1b\langle x, u, x \rangle$ that is reserved for A, then in order to maintain legality, u must also be reserved for L. However, note that u must have been reserved during the simulation phase, and no string was charged to u when u was reserved. Thus, we can charge the string v to u. Also, in order to reserve v for L, it is necessary to find some string s in Σ^n and reserve $0vs$ for A. This string $0vs$ may also be charged to u. Now follow Case 1 recursively.

14

Case 2: w is reserved for $\overline{A}$, and $w = 0xy$ where $|x| = |y|$. Again, $|x| = n$. Since at most $r < 2^n$ strings in $0\Sigma^{2n}$ have been reserved for $\overline{A}$, it is not necessary to reserve x for $\overline{L}$.

Case 3: w is reserved for A, and $w = 11\langle x, y, x\rangle$, with $x < y$. Reserve $11\langle x, y, y\rangle, 10\langle x, y, x\rangle$, $10\langle x, y, y\rangle$ for $\overline{A}$.

If x is reserved (either for L or for $\overline{L}$), and y is not yet reserved for L, then this reserves y for L; charge y to w. (Note in this case that $|y| = n$.) Also, some $z \in \Sigma^n$ must be found such that $0yz$ is not yet reserved for $\overline{A}$. (Such a z exists by the inductive hypothesis.) Reserve $0yz$ for A, and charge $0yz$ to w. Now continue as in Case 1.

Case 4: w is reserved for A, and $w = 11\langle x, y, y\rangle$, with $x < y$. Reserve $11\langle x, y, x\rangle, 10\langle x, y, x\rangle$, $10\langle x, y, y\rangle$ for $\overline{A}$.

This case is symmetric to Case 3; in this case it is x that must be reserved for L if y is already reserved for L.

Case 5: w is reserved for A, and $w = 10\langle x, y, x\rangle$, with $x < y$. Reserve $11\langle x, y, y\rangle, 11\langle x, y, x\rangle$, $10\langle x, y, y\rangle$ for $\overline{A}$.

Note that by condition (5), if $10\langle x, y, x\rangle$ is in a legal set A, it implies that $|x| = |y| = n$. Since no string in Σ^n has been reserved for $\overline{L}$, this does not cause y to be reserved for $\overline{L}$.

If x has been reserved for L, then this causes y to be reserved for L; charge y to w. Also, some $z \in \Sigma^n$ must be found such that $0yz$ can be reserved for A; charge $0yz$ to w. Now continue as in Case 1.

Case 6: w is reserved for A, and $w = 10\langle x, y, y\rangle$, with $x < y$. Reserve $11\langle x, y, y\rangle, 10\langle x, y, x\rangle$, $11\langle x, y, y\rangle$ for $\overline{A}$.

If y has been reserved for L, then this causes x to be reserved for L; charge x to w. Also, some $z \in \Sigma^n$ must be found such that $0xz$ can be reserved for A; charge $0xz$ to w. Now continue as in Case 1.

Case 7: w is reserved for $\overline{A}$, where w is in the set $S_{x,y} = \{1b\langle x, y, v\rangle : 1 \in \{0,1\}, v \in \{x, y\}\}$, for some $x < y$. If three elements of $S_{x,y}$ have been reserved for $\overline{A}$, this is equivalent to reserving the fourth element of $S_{x,y}$ for A. Continue as in the appropriate case above.

Case 8: None of the previous cases apply. In this case nothing else needs to be reserved.

An examination of these cases shows that no query is charged with reserving more than one string in $0\Sigma^{2n}$ for A or $\overline{A}$, and no query is charged with reserving more than one string for L. In addition, in no case is any string reserved for $\overline{L}$.

15

That completes the proof of the lemma and of the theorem. ∎

Corollary 16

- There is an oracle relative to which there is a P-selective set that is not in $\hat{EL}_2$.

- There is an oracle relative to which there is a P-cheatable set that is not in $\hat{EL}_2$.

We do not know if Corollary 16 is true for *all* oracles, or if there are also oracles relative to which every P-selective and P-cheatable set is in $\hat{EL}_2$. All known constructions of P-selective and P-cheatable sets involve tally languages; it is not currently known how to give an explicit construction of a P-selective or P-cheatable set that is provably not in $E_T(TALLY)$.

All co-sparse sets are in $\hat{L}_3$ [KS-85]: the following theorem shows that this result is essentially optimal with result to relativizable proof techniques.

Theorem 17 There is an oracle A relative to which there is a co-sparse set in NP that is not in L_2.

Proof: We will present an oracle A such that the set $L = \{u : \exists v \in \Sigma^{|u|}\ uv \notin A\}$ is co-sparse, and such that $NP^L \not\subseteq coNP^{NP^A}$. Note that L is clearly in NP^A, and note also that the conditions on L and A imply that $coNP^L$ is not contained in NP^{NP^A}, and hence L is not in L_2^A.

Let $B = \{0^n : \exists u \in \Sigma^n\ u \notin L\}(= \{0^n : \exists u \in \Sigma^n\ \forall v \in \Sigma^n\ uv \in A\})$. B is clearly in NP^L. Our goal is to build A so that B is not in $coNP^{NP^A}$. Note that B is in $coNP^{NP^A}$ iff there is some polynomial q_i and some deterministic polynomial-time machine P_k such that for all n, the condition

$$\exists u \in \Sigma^n\ \forall v \in \Sigma^n\ uv \in A \tag{6}$$

is equivalent to the condition

$$\forall y \in \Sigma^{\leq q_i(n)}\ \exists z \in \Sigma^{\leq q_i(n)}(P_k^A(0^n, y, z) = 1). \tag{7}$$

Thus our stage construction, at stage $\langle i, k \rangle$, will construct $A \cap \Sigma^n$ so that conditions (6) and (7) are not equivalent, while maintaining the condition that L is co-sparse. (Actually, we will maintain the condition that, for every n, either L contains all of Σ^n, or L contains all but one string in Σ^n.)

Begin the construction in stage 0 with $A = \{00, 01, 10, 11\}$, and set $n_0 = 2$.

Stage $l = \langle i, k \rangle$: Let p and r be polynomials such that for all m the number of strings in $\Sigma^{\leq q_i(m)} \leq 2^{p(m)}$ and such that for all y and z in $\Sigma^{\leq q_i(m)}$, the running time of P_i on input

$(0^m, y, z)$ is bounded by $r(m)$. Choose n so that $n > n_{l-1}$, and such that $p(n) < 2^{n/2}$ and $r(n) < 2^{n/2}$. Let $n_l \leftarrow 2r(n)$.

First put into $\overline{A}$ all strings of length m for all m such that $n_{l-1} < m < 2n$ or $2n < m \leq n_l$. This guarantees that the initial segment of L is co-sparse, except that at this point there are still no strings of length n in L.

For each string $u \in \Sigma^n$, let $\tilde{u}$ denote the set $\{uv : v \in \Sigma^n\}$. Note that $0^n \in B \iff \tilde{u} \subseteq A$ for some $u \in \Sigma^n$.

Let a set S be called a *sample set* if S contains exactly one element of $\tilde{u}$ for each $u \in \Sigma^n$. Note that there are 2^{n2^n} sample sets, $S_1, S_2, \ldots S_{2^{n2^n}}$. Also note that $0^n \notin B$ iff $\overline{A}$ contains some sample set.

Case 1: There is some sample set S_j so that with oracle $A \cup (\Sigma^{2n} - S_j)$, condition (7) evaluates to TRUE. In this case, add to A the strings in $\Sigma^{2n} - S_j$. Note that this makes condition (6) false and condition (7) true, and that P_i touches only strings of length less than n_l. Also, in this case, L contains every string in Σ^n.

Case 2: There is some u such that with oracle $A \cup (\Sigma^{2n} - \tilde{u})$, condition (7) evaluates to FALSE. In this case, add to A the strings in $\Sigma^{2n} - \tilde{u}$. Note that this makes condition (6) true and condition (7) false. Also, in this case, L contains every string in Σ^n except u.

Case 3: Neither of the previous two cases apply. But this is precisely the situation that was shown to be impossible by Baker and Selman in the proof of Lemma 2.4 of [BS-79]. ∎

Corollary 18

- There is an oracle relative to which P-close∩NP is not contained in L_2.

- There is an oracle relative to which P/poly∩NP is not contained in L_2.

5   Conclusions

The study of the low hierarchy is motivated by the desire to have evidence that certain classes of sets contain no NP-complete sets. The lower in the low hierarchy a class resides, the stronger the evidence is that its sets are not NP-complete. In order to have the strongest evidence possible, it is natural to try to improve on known lowness results.

Thus, it is important to know what sorts of lowness results are impossible given the current state of the art, and which improvements might reasonably be sought; this paper is the first contribution in this direction.

We have examined the current state of knowledge concerning the lowness properties of certain classes of sets. In many cases, we have been able to present lowness results that

are provably optimal, and in most cases we have been able to present lowness results that cannot be improved using any relativizable proof technique.

In addition, we have studied the lowness properties of the classes of sets equivalent to or reducible to sparse and tally sets. These results help clarify the structure underlying the low hierarchies.

As the tables indicate, further progress on placing sets at the correct level of the low hierarchies may hinge on resolving the question of whether $P/poly = E_T(SPARSE)$. This question, mentioned both in [TB-88] and in [AW-88], is a special case of the more general question of whether it is the case for *any* standard reducibility $\leq_r^p$ that $P_r(SPARSE) = E_r(SPARSE)$. Recent results by the authors suggest that these questions may be quite difficult, since, for example, one can show that $P=NP \implies P_m(SPARSE) = E_m(SPARSE)$, and the question of whether $P_m(TALLY) = E_m(TALLY)$ is equivalent to some basic open questions in complexity theory [AH-89b]. At this point, however, nothing is known about the $P/poly =? E_T(SPARSE)$ question.

One other avenue for further work involves the notion of self-reducibility. It can be shown that for some classes C, the self-reducible elements of C have lowness properties not shared by other elements of C. For example, it is shown in [KS-85] that the self-reducible sets in NP that have polynomial-size circuits are in L_2. What relativized lower bounds can be proved on the lowness of this class?

In many cases, we have claimed that lowness results are "optimal." We should caution, however, that these results are optimal only with respect to the low hierarchies *as they have been defined*. However, just as one can define (infinitely) many intermediate levels of the polynomial-time hierarchy, so can one define corresponding levels of the low hierarchies. Using a finer granularity in defining the low hierarchies, it may be possible to characterize more precisely the sort of "lowness" results which can be proved for various classes; that was not our intent in this paper.

Finally, we mention that lowness results for various classes have recently been presented in [Ko-87, Sc-88, Kä-88]. We leave as an open problem the investigation of the optimality of these results.

6   Acknowledgment

References

[AH-89a]   E. Allender and L. Hemachandra, *Lower Bounds for the Low Hierarchy*, Proceedings, 16th International Colloquium on Automata, Languages, and Programming, Lecture Notes in Computer Science 372, pp. 31–45.

[AH-89b]   E. Allender and L. Hemachandra, in preparation.

[AW-88]   E. Allender and O. Watanabe, *Kolmogorov complexity and degrees of tally sets*, To appear in Information and Control. A preliminary version of this work appeared in Proc. 3rd Structure in Complexity Theory Conference, pp. 102–111.

[ABG-89]   A. Amir. R. Beigel, and W. Gasarch, *Cheatable, P-terse, and P-superterse Sets*, in preparation. A preliminary version appeared as Technical Report TR-2090. Department of Computer Science, University of Maryland, 1988.

[AG-87]   A. Amir and W. Gasarch. *Polynomial Terse Sets*, Information and Computation 77. 37–56.

[BGS-75]   T. Baker, J. Gill, and R. Solovay, *Relativizations of the P=?NP question*, SIAM J. Comput. 4, 431–444.

[BS-79]   T. Baker and A. Selman, *A second step toward the polynomial hierarchy*, Theoretical Computer Science 8, 177–187.

[BB-86]   J. Balcázar and R. Book, *Sets with small generalized Kolmogorov complexity*, Acta Informatica 23, 679–688.

[BBS-86]   J. Balcázar. R. Book, and U. Schöning. *Sparse sets, lowness, and highness*, SIAM J. Comput. 15, 739–747.

[BDG-88]   J. Balcázar, J. Díaz. and J. Gabarró, Structural Complexity I, Springer-Verlag, Berlin/New York.

[Be-87]   R. Beigel, *Query-Limited Reducibilities*, Stanford Ph.D. thesis, Palo Alto, CA.

[Be-89]   R. Beigel, *Bounded queries to SAT and the Boolean hierarchy*, to appear in Theoretical Computer Science. Also available as Technical Report 7, Johns Hopkins University, Dept. of Computer Science, 1987.

[BK-88]   R. Book and K. Ko, *On sets reducible to sparse sets*, SIAM J. Comput. 17, 903–919.

[Bo-74]   R. Book, *Tally languages and complexity classes*, Information and Control 26, 186–193.

[GJY-87] J. Goldsmith, D. Joseph, and P. Young, *Self-reducible, P-selective, near-testable, and P-cheatable sets: the effect of internal structure on the complexity of a set*, Proc. 2nd Structure in Complexity Theory Conference, pp. 50–59.

[Kä-88] J. Kämper, *Non-uniform proof systems: a new framework to describe non-uniform and probabilistic complexity classes*, Proceedings, Eighth Conference on Foundations of Software Technology and Theoretical Computer Science, Lecture Notes in Computer Science 338, pp. 193–210.

[Ko-82] K. Ko, *The maximum value problem and NP real numbers*, Journal of Computer and System Sciences 24, 15–35.

[Ko-83] K. Ko, *On the definition of some complexity classes of real numbers*, Mathematical Systems Theory 16, 95–109.

[Ko-86] K. Ko, *Applying Techniques of Discrete Complexity Theory to Numerical Computation*, in Studies in Complexity Theory. R. Book, ed., Wiley and Sons, New York, NY, pp. 1–62.

[Ko-87] K. Ko, *On helping by robust oracle machines*, Theoretical Computer Science 52, 15–36.

[Ko-88] K. Ko, *Distinguishing Bounded Reducibilities by Sparse Sets*, Proc. 3rd Structure in Complexity Theory Conference, pp. 181–193.

[Ko-89a] K. Ko, *Separating the low and high hierarchies by oracles*, submitted for publication.

[Ko-89b] K. Ko, *Constructing oracles by lower bound techniques for circuits*. to appear in Proc. International Symposium on Combinatorial Optimization, China. 1988 (Ding-Zhu Du, editor).

[KS-85] K. Ko and U. Schöning, *On circuit-size complexity and the low hierarchy in NP*, SIAM J. Comput. 14, 41–51.

[LLS-75] R. Ladner, N. Lynch, and A. Selman, *A comparison of polynomial-time reducibilities*. Theoretical Computer Science 1, 103–123.

[Ma-86] S. Mahaney, *Sparse sets and reducibilities*, in Studies in Complexity Theory, R. Book. ed., Wiley and Sons, New York, NY, pp. 63–118.

[MP-79] A. Meyer and M. Paterson, *With what frequency are apparently intractable problems difficult?*. Technical Report MIT/LCS/TM-126, Lab. for Computer Science, MIT.

[Sc-83] U. Schöning, *A low and a high hierarchy within NP*, Journal of Computer and System Sciences 27, 14–28.

[Sc-85] U. Schöning, Complexity and Structure, Lecture Notes in Computer Science 211.

[Sc-86] U. Schöning, *Complete sets and closeness to complexity classes*, Mathematical Systems Theory 19, 29–41.

[Sc-87] U. Schöning, *Graph Isomorphism is in the Low Hierarchy*, Journal of Computer and System Sciences 37, 312–323.

[Sc-88] U. Schöning, *Robust oracle machines*, Proceedings, 13th Symposium on Mathematical Foundations of Computer Science, Lecture Notes in Computer Science 324, pp. 93–106.

[Se-79] A. Selman, *P-selective sets, tally languages, and the behavior of polynomial time reducibilities on NP*, Math. Systems Theory 13, 55–65.

[Se-81] A. Selman, *Some observations on NP real numbers and P-selective sets*, Journal of Computer and System Sciences 23, 326–332.

[TB-88] S. Tang and R. Book, *Separating polynomial-time Turing and truth-table degrees of tally sets*, Proc. 15th International Colloquium on Automata, Languages, and Programming, Lecture Notes in Computer Science 317, pp. 591–599.

Class C	$C \subseteq$	$C \not\subseteq$	optimal?
TALLY	$\widehat{EL}_2$ [BB-86]	N/A	YES
Standard Left Cuts	$\widehat{EL}_2$[Se-81,BB-86]	N/A	YES
General Left Cuts	$\widehat{EL}_2$	N/A	YES
BPP	EL_2[Sc-85]	$\widehat{EL}_2$ †	YES‡
P-selective	EL_2[ABG-89]	$\widehat{EL}_2$ †	YES‡
weakly P-selective	EL_2[ABG-89]	$\widehat{EL}_2$†	YES‡
P-cheatable	EL_2	$\widehat{EL}_2$†	YES‡
SPARSE	$\widehat{EL}_3$	EL_2	YES
co-SPARSE	$\widehat{EL}_3$	EL_2	YES
APT	$\widehat{EL}_3$	EL_2	YES
P-close	$\widehat{EL}_3$	EL_2	YES
P/poly	EL_3[BBS-86]	EL_2	*

Table 1: Inclusions and non-inclusions for the Extended Low Hierarchy.
For example, every sparse set is in $\widehat{EL}_3$, there are sparse sets that are not in EL_2, and thus the given lowness result is optimal. References given refer to the papers where the stated result is first proved. For example, it is shown in [BBS-86] that P/poly is contained in EL_3. If no reference is given, the result is due to this paper. Since the lowest level in the extended low hierarchy is $\widehat{EL}_2$, the third column is not applicable for the first three rows in the table.
†This is a relativization result; we present an oracle relative to which this class is not contained in $\widehat{EL}_2$.
‡This result cannot be improved using any relativizable proof technique.
* If this class is not contained in $\widehat{EL}_3$, then P/poly $\neq E_T(SPARSE)$.

Class C	$C \cap NP \subseteq$	$C \cap NP \nsubseteq$	optimal?
TALLY	$\hat{L}_2$ [KS-85]	L_1[BGS-75]	YES
Standard Left Cuts	$\hat{L}_2$	L_1[BGS-75]	YES
General Left Cuts	$\hat{L}_2$	L_1[BGS-75]	YES
SPARSE	$\hat{L}_2$[KS-85]	L_1[BGS-75]	YES
APT	$\hat{L}_2$[KS-85]	L_1[BGS-75]	YES
P-selective	L_2[KS-85]	$\hat{L}_2$	YES
Weakly P-selective	L_2[KS-85]	$\hat{L}_2$	YES
P-cheatable	L_2[ABG-89]	$\hat{L}_2$	YES
RP	L_2[KS-85]	$\hat{L}_2$	YES
BPP	L_2[Sc-85]	$\hat{L}_2$	YES
coAM	L_2[Sc-87]	$\hat{L}_2$	YES
co-SPARSE	$\hat{L}_3$ [KS-85]	L_2	YES
P-close	$\hat{L}_3$[Sc-85]	L_2	YES
P/poly	L_3[KS-85]	L_2	‡‡

Table 2: Inclusions and relativized non-inclusions for the Low Hierarchy in NP.

For example. every sparse set in NP is in $\hat{L}_2$, there is an oracle relative to which there are sparse sets in NP that are not in L_1, and thus the given lowness result is optimal.

‡‡If this class is not contained in $\hat{L}_3$. then there is an oracle relative to which P/poly $\neq$ $E_T(SPARSE)$.

Class C	$C \subseteq$	$C \not\subseteq$	optimal?
$E_T(TALLY)$	$\hat{EL}_2$ [BB-86]	N/A	YES
$P_m(TALLY)$	$\hat{EL}_2$	N/A	YES
$(=P_{btt}(TALLY))$[BK-88]			
$E_T(SPARSE)$	$\hat{EL}_3$	EL_2	YES
$P_m(SPARSE)$	$\hat{EL}_3$	EL_2	YES
$P_{1-tt}(SPARSE)$	$\hat{EL}_3$	EL_2	YES
$P_{2-tt}(SPARSE)$	EL_3[BBS-86]	EL_2	**
$P_{tt}(TALLY)$	EL_3[BBS-86]	EL_2	**
$(=P_T(SPARSE)$			
$=P/poly)$[BK-88. Sc-85]			

Table 3: Extended Low Hierarchy results for sets reducible to sparse and tally sets.
** If this class is not contained in $\hat{EL}_3$, then P/poly $\neq$ $E_T(SPARSE)$. (Also, [AH-89b] shows that if P=NP, then $P_{2-tt}(SPARSE) \subseteq E_{tt}(SPARSE) \subseteq \hat{EL}_3$.)

Class C	$C \cap NP \subseteq$	$C \cap NP \not\subseteq$	optimal?
$E_T(TALLY)$	$\hat{L}_2$ [BB-86]	L_1[BGS-75]	YES
$P_m(TALLY)$	$\hat{L}_2$	L_1[BGS-75]	YES
$(=P_{btt}(TALLY))$[BK-88]			
$P_m(SPARSE)$	$\hat{L}_2$	L_1[BGS-75]	YES
$E_{1-tt}(SPARSE)$	$\hat{L}_3$	L_2	YES
$E_T(SPARSE)$	$\hat{L}_3$	L_2	YES
$P_{1-tt}(SPARSE)$	$\hat{L}_3$	L_2	YES
$P_{2-tt}(SPARSE)$	L_3[KS-85]	L_2	††
$P_{tt}(TALLY)$	L_3[KS-85]	L_2	††
$(=P_T(SPARSE)$			
$=P/poly)$[BK-88, Sc-85]			

Table 4: Low Hierarchy results for sets reducible to sparse and tally sets.
The non-inclusion results in column three are relativization results: for example, there is an oracle relative to which $E_T(SPARSE)$ is not in L_2.
††If this class is not contained in $\hat{L}_3$, then there is an oracle relative to which P/poly $\neq$ $E_T(SPARSE)$.