Graph Isomorphism is Low for ZPP\textsuperscript{NP} and other
Lowness results

V. Arvind\textsuperscript{1} and J. Köbler\textsuperscript{2}

\textsuperscript{1} Institute of Mathematical Sciences, C.I.T Campus, Chennai 600113, India
\textsuperscript{2} Humboldt-Universität zu Berlin, Institut für Informatik, D-10099 Berlin, Germany

Abstract. We show the following new lowness results for the probabilistic class ZPP\textsuperscript{NP}.
- The class AM \cap \text{coAM} is low for ZPP\textsuperscript{NP}. As a consequence it follows that Graph Isomorphism and several group-theoretic problems known to be in AM \cap \text{coAM} are low for ZPP\textsuperscript{NP}.
- The class IP[P/poly], consisting of sets that have interactive proof systems with honest provers in P/poly, is also low for ZPP\textsuperscript{NP}.

We consider lowness properties of nonuniform function classes, namely, NPMV/poly, NPSV/poly, NPMV\textsubscript{1}/poly, and NPSV\textsubscript{1}/poly. Specifically, we show that
- Sets whose characteristic functions are in NPSV/poly and that have program checkers (in the sense of Blum and Kannan [9]) are low for AM and ZPP\textsuperscript{NP}.
- Sets whose characteristic functions are in NPMV\textsubscript{1}/poly are low for \Sigma\textsubscript{p}^P.

1 Introduction

In the recent past the probabilistic class ZPP\textsuperscript{NP} has appeared in different results and contexts in complexity theory research. E.g. consider the result MA \subseteq ZPP\textsuperscript{NP} [1, 14] which sharpens and improves Sipser’s theorem BPP \subseteq \Sigma\textsubscript{p}^P. The proof in [1] uses derandomization techniques based on hardness assumptions [22].

Another example is the result that if SAT \in P/poly then PH = ZPP\textsuperscript{NP} [21, 5], which improves the classic Karp-Lipton theorem.\textsuperscript{1} Actually, Köbler and Watanabe in [21] prove that every self-reducible set\textsuperscript{2} A in (NP \cap \text{co-NP})/poly is low for ZPP\textsuperscript{NP}, i.e. ZPP\textsuperscript{NP}\textsuperscript{A} = ZPP\textsuperscript{NP}. This stronger result is in a sense natural, since there is usually an underlying lowness result that implies a collapse consequence result like the Karp-Lipton theorem. We may recall here that the lowness result underlying the Karp-Lipton theorem is that self-reducible sets in P/poly are low for \Sigma\textsubscript{p}^P [25].

The notion of lowness was first introduced in complexity theory by Schöning in [25]. It has since then been an important conceptual tool in complexity theory, see e.g. the survey paper [17].

\textsuperscript{1} The Karp-Lipton theorem states that if SAT \in P/poly then PH collapses to \Sigma\textsubscript{p}^P.

\textsuperscript{2} By self-reducibility we mean word-decreasing self-reducibility which is adequate because standard complexity classes contained in EXP have such self-reducible complete problems.
1.1 Lowness for ZPP\textsuperscript{NP}

We recall the formal definition of lowness [25]. For a relativizable complexity class \( \mathcal{C} \) such that for all sets \( A, A \in \mathcal{C}^A \), let \( \text{Low}(\mathcal{C}) \) denote \( \{ A \mid \mathcal{C}^A = \mathcal{C} \} \). Clearly, \( \text{Low}(\mathcal{C}) \) is contained in \( \mathcal{C} \) and consists of languages that are powerless as oracles for \( \mathcal{C} \).

Few complexity classes have their low sets exactly characterized. These are well-known examples: \( \text{Low}(\text{NP}) = \text{NP} \cap \text{co-NP} \). \( \text{Low}(\text{AM}) = \text{AM} \cap \text{coAM} \) [26]. For most complexity classes however, a complete characterization of the low sets appears to be a challenging open question. Regarding \( \text{Low}(\Sigma_2^p) \), Schöning proved [26] that \( \text{AM} \cap \text{coAM} \) is contained in \( \text{Low}(\Sigma_2^p) \), implying that \( \text{Low}(\text{AM}) \subseteq \text{Low}(\Sigma_2^p) \). This containment is anomalous because \( \text{AM} \nsubseteq \Sigma_2^p \) in some relativized worlds [24]. Indeed, lowness appears to have other anomalous properties: it is not known to preserve containment of complexity classes, for example \( \text{NP} \nsubseteq \text{PP} \) but \( \text{NP} \cap \text{co-NP} \) is not known to be in \( \text{Low}(\text{PP}) \). Similarly, \( \text{NP} \nsubseteq \text{MA} \) but \( \text{NP} \cap \text{co-NP} \) is not known to be in \( \text{Low}(\text{MA}) \). Little is known about \( \text{Low}(\text{MA}) \) except that it contains \( \text{BPP} \) and is contained in \( \text{MA} \cap \text{co-MA} \) [19].

Regarding \( \text{ZPP}^{\text{NP}} \), it is shown in [21] that \( \text{Low}(\text{ZPP}^{\text{NP}}) \subseteq \text{Low}(\Sigma_2^p) \). No characterization of \( \text{Low}(\text{ZPP}^{\text{NP}}) \) is known. Our aim is to show some inclusions in \( \text{Low}(\text{ZPP}^{\text{NP}}) \) as a first step.

We first show in this paper that \( \text{AM} \cap \text{coAM} \) is low for \( \text{ZPP}^{\text{NP}} \), i.e. \( \text{AM} \cap \text{coAM} \subseteq \text{Low}(\text{ZPP}^{\text{NP}}) \). Hence we have the inclusion chain

\[
\text{Low}(\text{MA}) \subseteq \text{Low}(\text{AM}) \subseteq \text{Low}(\text{ZPP}^{\text{NP}}) \subseteq \text{Low}(\Sigma_2^p).
\]

It follows that Graph Isomorphism and other group-theoretic problems known to be in \( \text{AM} \cap \text{coAM} \) [4] are low for \( \text{ZPP}^{\text{NP}} \).

We prove another lowness result for \( \text{ZPP}^{\text{NP}} \): Let \( \text{IP}[\text{P/poly}] \) denote languages that have interactive proof systems with honest prover in \( \text{P/poly} \). We show that \( \text{IP}[\text{P/poly}] \subseteq \text{Low}(\text{ZPP}^{\text{NP}}) \), improving the containment \( \text{IP}[\text{P/poly}] \subseteq \text{Low}(\Sigma_2^p) \) shown in [3]. Our proof has a derandomization component in which the Nisan-Wigderson pseudorandom generator [22] is used to derandomize the verifier in the IP[\text{P/poly}] protocol. The rest of the proof is based on the random sampling technique as applied in [5, 18].

1.2 \( \text{NP/\text{poly}} \cap \text{co-NP/\text{poly}} \) and subclasses

As shown in [21], self-reducible sets in \( \text{(NP \cap co-NP)/poly} \) are low for \( \text{ZPP}^{\text{NP}} \). However, there are technical difficulties due to which this result does not carry over to \( \text{NP/\text{poly}} \cap \text{co-NP/\text{poly}} \). The best known collapse consequence of \( \text{NP} \subseteq \text{NP/\text{poly}} \cap \text{co-NP/\text{poly}} \) is \( \text{PH} \subseteq \text{ZPP}(\Sigma_2^p) \) [21].

In order to better understand this aspect of \( \text{NP/\text{poly}} \cap \text{co-NP/\text{poly}} \) the authors of [11] introduce two interesting subclasses of \( \text{NP/\text{poly}} \cap \text{co-NP/\text{poly}} \) which we discuss in Section 5. We notice firstly that \( \text{NP/\text{poly}} \cap \text{co-NP/\text{poly}} \) and the above-mentioned subclasses are closely connected to the function classes \( \text{NP}^{\text{MV}}/\text{poly}, \text{NP}^{\text{SV}}/\text{poly}, \text{NP}^{\text{MV}}{i}/\text{poly}, \) and \( \text{NP}^{\text{SV}}{i}/\text{poly} \), which are nonuniform analogues of the function classes \( \text{NP}^{\text{MV}}, \text{NP}^{\text{SV}}, \text{NP}^{\text{MV}}{i}, \) and \( \text{NP}^{\text{SV}}{i} \).
introduced and studied by Selman and other researchers [27, 12]. More precisely, we note that $A \in (\text{NP} \cap \text{co-NP})/\text{poly}$ if and only if $\chi_A \in \text{NP}^{\text{NPV}}/\text{poly}$, where $\chi_A$ denotes the characteristic function of a language $A$. Similarly, $A \in \text{NP}/\text{poly} \cap \text{co-NP}/\text{poly}$ if and only if $\chi_A \in \text{NP}^{\text{PMV}}/\text{poly}$. Likewise, $\text{NP}^{\text{SV}}/\text{poly}$ and $\text{NP}^{\text{MV}}/\text{poly}$ capture the two new subclasses of $\text{NP}/\text{poly} \cap \text{co-NP}/\text{poly}$ defined in [11].

We prove the following new lowness results for these classes:

- We show that self-reducible sets whose characteristic functions are in the function class $\text{NP}^{\text{MV}}/\text{poly}$ are low for $\Sigma^P_2$ (this result is essentially the lowness result underlying the collapse consequence i.e. Theorem 5.2 in [11]).
- We show that all self-checkable sets — In the program checking sense of Blum and Kannan [9]— whose characteristic functions are in $\text{NP}^{\text{SV}}/\text{poly}$ are low for $\text{AM}$.

Several proofs are omitted from this extended abstract. A full version of the paper is available as a technical report [2].

2 Preliminaries

Let $\Sigma = \{0, 1\}$. We denote the cardinality of a set $X$ by $|X|$ and the length of a string $x \in \Sigma^*$ by $|x|$. The characteristic function of a language $L \subseteq \Sigma^*$ is denoted by $\chi_L$. The definitions of standard complexity classes like P, NP, E, EXP etc. can be found in standard books [8, 23]. A relativized complexity class $\mathcal{C}$ with oracle $A$ is denoted by either $\mathcal{C}^A$ or $\mathcal{C}(A)$. Likewise, we denote an oracle Turing machine $M$ with oracle $A$ by $M^A$ or $M(A)$.

For a class $\mathcal{C}$ of sets and a class $\mathcal{F}$ of functions from $\mathbb{N}$ to $\Sigma^*$, let $\mathcal{C}/\mathcal{F}$ [15] be the class of sets $A$ such that there is a set $B \in \mathcal{C}$ and a function $h \in \mathcal{F}$ such that for all $x \in \Sigma^*$,

$$x \in A \iff \langle x, h(|x|) \rangle \in B.$$ 

The function $h$ is called an advice function for $A$.

We recall definitions of $\text{AM}$ and $\text{MA}$. A language $L$ is in $\text{AM}$ if there exist a polynomial $p$ and a set $B \in \text{P}$ such that for all $x$, $|x| = n$,

$$x \in A \Rightarrow \Pr_{y \in \{0, 1\}^p \sim n} [\exists y, |y| = p(n) : \langle x, y, r \rangle \in B] = 1,$n

$$x \notin A \Rightarrow \Pr_{y \in \{0, 1\}^p \sim n} [\forall y, |y| = p(n) : \langle x, y, r \rangle \in B] \leq 1/4.$n

A language $L$ is in $\text{MA}$ if there exist a polynomial $p$ and a set $B \in \text{P}$ such that for all $x$, $|x| = n$,

$$x \in A \Rightarrow \Pr_{y \in \{0, 1\}^p \sim n} [\exists y, |y| = p(n) : \langle x, y, r \rangle \in B] \geq 3/4,$n

$$x \notin A \Rightarrow \Pr_{y \in \{0, 1\}^p \sim n} [\forall y, |y| = p(n) : \langle x, y, r \rangle \in B] \leq 1/4.$n

Notice that we have taken the definition of $\text{AM}$ with 1-sided error, known to be equivalent to $\text{AM}$ with 2-sided error. Definitions for single and multiprover
interactive proof systems can be found in standard texts, e.g. [23]. Let MIP denote the class of languages with multiprover interactive protocols and IP denote the class of languages with single-prover interactive protocols. We denote by MIP[\mathcal{C}] and IP[\mathcal{C}] the respective language classes where the prover complexity is bounded by FP[\mathcal{C}], which is the set of functions that can be computed by a polynomial-time oracle transducer with oracle in \mathcal{C}.

3 AM \cap \text{coAM} is low for ZPP^{NP}

In this section we show that AM \cap \text{coAM} is low for ZPP^{NP}. It follows that Graph Isomorphism and a host of group-theoretic problems known to be in AM \cap \text{coAM} [4] are all low for ZPP^{NP}. We recall here that it is already known that AM \cap \text{coAM} is low for \Sigma^p_2 [26] and also for AM [19].

We notice first that although AM \cap \text{coAM} \subseteq ZPP^{NP} (because AM \subseteq \text{coR}^{NP} and the equality ZPP = R \cap \text{coR} relativizes) and AM \cap \text{coAM} is low for itself, it doesn’t follow that AM \cap \text{coAM} is low for ZPP^{NP}. As mentioned before, NP \cap \text{co-NP} is trivially low for NP but is not known to be low for PP or MA.

**Theorem 1.** AM \cap \text{coAM} is low for ZPP^{NP}.

**Proof.** Let L be any set in AM \cap \text{coAM}. We need to show that a given ZPP^{NP,k} machine M can be simulated in ZPP^{NP}. Consider an input x of length bounded by n to the machine M. Suppose the lengths of all the queries made to L during the computation are bounded by m. Since L ∈ AM \cap \text{coAM}, it follows from standard probability amplification techniques and quantifier swapping (cf. [26]) that there are NP sets A and B and a polynomial p such that \forall y : |y| \leq m, there is a subset S ⊆ \{0, 1\}^{p(m)} of size ||S|| \geq 2^{p(m)-1} with the following property:

\forall w : \langle y, w \rangle \in A \text{ and } \forall w \in S : \langle y, w \rangle \notin B

and \ y \notin L \text{ implies }

\forall w : \langle y, w \rangle \in B \text{ and } \forall w \in S : \langle y, w \rangle \notin A.

Notice that in the above we are using the fact that AM protocols can be assumed to have one-sided error.

In other words, a large fraction of the w’s act as advice strings using which membership in L for strings of length m can be decided with an NP \cap \text{co-NP} computation. Notice, however, that it would be incorrect for us to claim from here that L ∈ (NP \cap \text{co-NP})/poly, because if we use a string from \{0, 1\}^{p(m)} \subseteq S as advice, the resulting combination of machines for A and B may not yield an NP \cap \text{co-NP} computation for some input y ∈ \Sigma^{\leq m}. However, we observe that the above property of advice strings in S implies that w ∈ S if and only if using w as advice yields an NP \cap \text{co-NP} computation for all inputs y ∈ \Sigma^{\leq m}.  

Thus, a candidate advice \( w \in \Sigma^{m(n)} \) is not in \( S \) if and only if it satisfies the following NP predicate:

\[
\exists y \in \Sigma^{\leq m}: \langle y, w \rangle \in A \cap B.
\]

We now describe the \( \text{ZPP}^{\text{NP}} \) machine \( N \) that simulates the given \( \text{ZPP}^{\text{NP}} \)-machine \( M \) on some input \( x \). Machine \( N \) first randomly guesses an advice string in \( w \in \Sigma^{m(n)} \) which, by assumption, is in \( S \) with probability \( 1/2 \). A single NP query using the above NP predicate is now used to certify that \( w \in S \). Using such a \( w \) as advice, \( N \) can replace the oracle \( L \) with an NP \( \cap \text{co-NP} \) computation when it simulates \( M \).

**Corollary 1.** Graph Isomorphism is low for \( \text{ZPP}^{\text{NP}} \).

The above corollary follows since Graph Isomorphism is in \( \text{AM} \cap \text{coAM} \) [13]. The lowness result also holds for various group-theoretic problems known to be in \( \text{AM} \cap \text{coAM} \) [4].

Notice that the previous theorem essentially shows that we can simulate \( \text{AM} \cap \text{coAM} \) with an \( \text{NP} \cap \text{co-NP} \) computation using a random string in a \text{coNP} set as advice for the computation. This observation combined with the result of [21] (that self-reducible sets in \( (\text{NP} \cap \text{co-NP})/\text{poly} \) are low for \( \text{ZPP}^{\text{NP}} \)) immediately yields the following corollary.

**Corollary 2.** Self-reducible sets in \( (\text{AM} \cap \text{coAM})/\text{poly} \) are low for \( \text{ZPP}^{\text{NP}} \).

Additionally, we also have the following corollary in the average-case complexity setting. We first recall the definition of \( \text{AP} \) (see, e.g., [20] for a detailed treatment): \( \text{AP} \) is the class of decision problems \( A \) such that for every polynomial-time computable distribution there is an algorithm that decides \( A \) and is polynomial-time on the average for that distribution.

**Corollary 3.** If \( \text{NP} \subseteq \text{AP} \) then \( \text{AM} \cap \text{coAM} = \text{NP} \cap \text{co-NP} \).

The proof follows from the assumption \( \text{NP} \subseteq \text{AP} \) combined with the fact that for any set in \( \text{AM} \cap \text{coAM} \) a large fraction of strings satisfying a \text{coNP} predicate are good advice strings, as we have already seen in the proof of Theorem 1. Thus, a \( \text{ZPP} \) computation can randomly guess such an advice string and use an \( \text{AP} \) algorithm for the uniform distribution to decide the \text{coNP} predicate. This \( \text{AP} \) algorithm, with its running time truncated to a suitable polynomial bound, will still accept many of the randomly picked good advice strings. This is an application of ideas from [20].

4 **IP[\text{P/poly}] is low for \( \text{ZPP}^{\text{NP}} \)**

The class \( \text{IP[\text{P/poly}]} \) already figures, though implicitly, in the proof of the result in [6] that if \( \text{EXP} \subseteq \text{P/poly} \) then \( \text{EXP} = \text{MA} \). We quickly recall the proof: Suppose \( \text{EXP} \subseteq \text{P/poly} \). Note that each language in \( \text{EXP} \) has a multiprover
interactive protocol in which the provers are in EXP. By assumption, therefore, the honest provers can be simulated by polynomial size circuits. Thus the (MIP) protocol can be simulated by an MA protocol where Merlin simply sends the circuits for the provers to Arthur in the first round. In other words, the proof shows the inclusion chain \( \text{EXP} \subseteq \text{MIP}^{[\text{P}/\text{poly}]} \subseteq \text{MA} \). Since the MA protocol is a single prover interactive protocol, we also have \( \text{MIP}^{[\text{P}/\text{poly}]} = \text{IP}^{[\text{P}/\text{poly}]} \subseteq \text{MA} \).

The above collapse consequence result of [6] motivates the study of lowness properties of \( \text{IP}^{[\text{P}/\text{poly}]} \). Our next result states that \( \text{IP}^{[\text{P}/\text{poly}]} \subseteq \text{Low}(\text{ZPP}^{\text{NP}}) \), improving the containment \( \text{IP}^{[\text{P}/\text{poly}]} \subseteq \text{Low}(\Sigma_2^p) \) shown in [3]. Our result strengthens the result of [18] that NP sets in P/poly with self-computable witnesses are low for ZPP^{NP}. \( \text{IP}^{[\text{P}/\text{poly}]} \) contains such NP sets, but \( \text{IP}^{[\text{P}/\text{poly}]} \) may not even be contained in NP. Although \( \text{IP}^{[\text{P}/\text{poly}]} \subseteq \text{MA} \subseteq \text{AM} \), \( \text{IP}^{[\text{P}/\text{poly}]} \) is not known to be closed under complement, and it is not known if \( \text{IP}^{[\text{P}/\text{poly}]} \) is contained in coAM. Thus, \( \text{IP}^{[\text{P}/\text{poly}]} \subseteq \text{Low}(\text{ZPP}^{\text{NP}}) \) appears incomparable to \( \text{AM} \cap \text{coAM} \subseteq \text{Low}(\text{ZPP}^{\text{NP}}) \) shown in Theorem 1 in the previous section. Our result is also incomparable to the result in [21] that self-reducible sets in P/poly are low for ZPP^{NP}. An interesting aspect of our proof is that it combines derandomization and almost uniform random sampling.

**Theorem 2.** \( \text{IP}^{[\text{P}/\text{poly}]} \) is low for ZPP^{NP}.

The above lowness result easily extends to \( \text{IP}^{(\text{NP} \cap \text{co-NP})/\text{poly}} \) by observing that the proof relativizes in the following sense: for any oracle set \( A \), \( \text{NP}^{\text{IP}^{[\text{P}/\text{poly}]} \mid A} \subseteq \text{ZPP}^{\text{NP} \mid A} \).

We conclude this section with another connection to the average-case complexity setting.

**Theorem 3.** If \( \text{NP} \subseteq \text{AP} \) and \( \text{NP} \subseteq \text{P}/\text{poly} \) then \( \text{PH} \) collapses to \( \Delta_2^p \).

5 Nonuniform function classes and lowness

We now study lowness properties of \( \text{NP}^{\text{MV}/\text{poly}}, \text{NP}^{\text{SV}/\text{poly}}, \text{NP}^{\text{MV}_i/\text{poly}}, \) and \( \text{NP}^{\text{SV}_i/\text{poly}} \). These are nonuniform analogs of the function classes \( \text{NP}^{\text{MV}}, \text{NP}^{\text{SV}}, \text{NP}^{\text{MV}_i}, \) and \( \text{NP}^{\text{SV}_i} \) studied by Selman [27] and other researchers, e.g. [12]. These nonuniform classes are interesting because when restricted to characteristic functions of sets, \( \text{NP}^{\text{SV}_i/\text{poly}} \) coincides with \( \text{NP} \cap \text{co-NP}/\text{poly} \) and \( \text{NP}^{\text{MV}_i/\text{poly}} \) coincides with \( \text{NP} \cap \text{co-NP}/\text{poly} \). Likewise, we note that the two subclasses of \( \text{NP}/\text{poly} \cap \text{co-NP}/\text{poly} \) studied in [11], namely all sets underproductively reducible to sparse sets and all sets overproductively reducible to sparse sets, also coincide with \( \text{NP}^{\text{SV}/\text{poly}} \) and \( \text{NP}^{\text{MV}_i/\text{poly}} \), respectively.

Following Selman’s notation in [27], a transducer is an NDTM \( T \) with a write-only output tape. On input \( x \) machine \( T \) outputs \( y \in \Sigma^* \) if there is an accepting path on input \( x \) along which \( y \) is output. Hence, the function defined by \( T \) on \( \Sigma^* \) could be multivalued and partial. Given a multivalued function \( f \)
on $\Sigma^*$ and $x \in \Sigma^*$ we use the notation
\[
\text{set-}f(x) = \{ y \mid f : x \mapsto y \}
\]
to denote the (possibly empty) set of function values for input $x$. We recall the basic definitions.

**Definition 1.** [10]

1. NPMV is the class of multivalued, partial functions $f$ for which there is a polynomial-time NDTM $N$ such that
   - (a) $f(x)$ is defined (i.e., $\text{set-}f(x) \neq \emptyset$) if and only if $N(x)$ has an accepting path.
   - (b) $y \in \text{set-}f(x)$ if and only if there is an accepting path of $N(x)$ where $y$ is output.
2. NPSV is the class of single-valued partial functions in NPMV.
3. NPMV$_1$ is the class of total functions in NPMV.
4. NPSV$_1$ is the class of total single-valued functions in NPMV.

The classes NPMV/poly, NPSV/poly, NPMV$_1$/poly, and NPSV$_1$/poly are the standard nonuniform analogs of the above classes defined as usual [15]: for $F \in \{ \text{NPMV, NPSV, NPMV}_1, \text{NPSV}_1 \}$, a multivalued partial function $f$ is in $F$/poly if there is a function $g \in F$, a polynomial $p$, and an advice function $h : 1^n \mapsto \Sigma^*$ with $|h(1^n)| \leq p(n)$ for all $n$, such that for all $x \in \Sigma^*$,
\[
\text{set-}f(x) = \text{set-}g(x, h(1^n))
\]

Before we connect these classes to $\text{NP/poly} \cap \text{co-NP/poly}$ and its subclasses defined in [11], we recall definitions from [11]: Consider polynomial-time nondeterministic oracle machines $N$ whose computation paths can have three possible outcomes: accept, reject, or ?. The machine $N$ can also be viewed as a transducer which computes, for given oracle $D$ and input $x$, a multivalued function. More precisely, if we identify accept with value 1 and reject with 0, and consider the ? computation paths as rejecting paths then $N^D$ defines a partial multivalued function: $\text{set-}N^D(x) \subseteq \{0, 1\}$. Machine $N^D$ is said to be underproductive if for each $x$ we have $\{0, 1\} \nsubseteq \text{set-}N^D(x)$, and $N$ is said to be robustly underproductive if for each oracle $D$ and input $x$ we have $\{0, 1\} \nsubseteq \text{set-}N^D(x)$. Likewise, $N^D$ is overproductive if for each $x$ we have $\text{set-}N^D(x) \neq \emptyset$, and $N$ is said to be robustly overproductive if for each oracle $D$ and input $x$ we have $\text{set-}N^D(x) \neq \emptyset$. With standard arguments we can convert a sparse set into a polynomial-size advice string and vice-versa (see, e.g., [8]). It follows that $A \in \text{NP/poly} \cap \text{co-NP/poly}$ if and only if there is a sparse set $S$ and a nondeterministic machine $N$ such that $N^S$ is both overproductive and underproductive and $A = L(N^S)$. Similarly, $A \in (\text{NP} \cap \text{co-NP})/\text{poly}$ if and only if there is a sparse set $S$ and a nondeterministic machine $N$ such that $A = L(N^S)$ and $N$ is both robustly overproductive and robustly underproductive and $A = L(N^S)$.

**Proposition 1.** Let $\chi_A$ denote the characteristic function for a set $A \subseteq \Sigma^*$:
1. $\chi_A$ is in NPMV/poly if and only if $A$ is in NP/poly $\cap$ co-NP/poly.
2. $\chi_A$ is in NPSV$_2$/poly if and only if $A$ is in (NP $\cap$ co-NP)/poly.
3. $\chi_A$ is in NPSV/poly if and only if there are a sparse set $S$ and a robustly
underproductive machine $N$ such that $A = L(N^S)$.
4. $\chi_A$ is in NPMV$_1$/poly if and only if there are a sparse set $S$ and a robustly
overproductive machine $N$ such that $A = L(N^S)$.

By abuse of notation, we identify $\chi_A$ with $A$ in this section. E.g., we write
$A \in$ NPSV/poly when we mean $\chi_A \in$ NPSV/poly. We now turn to lowness
questions for the nonuniform function classes. The classes NP/poly $\cap$ co-NP/poly
and (NP $\cap$ co-NP)/poly are of interest in the context of deriving strong collapse
consequences from the assumption that NP (or other hard complexity classes)
is contained in one of these classes. We recall the known collapse consequence
result shown in [21] for NP/poly $\cap$ co-NP/poly under the assumption that NP is
contained therein: If $\text{NP} \subseteq \text{NP/poly} \cap \text{co-NP/poly}$ then PH collapses to $\text{ZP}^{\text{NP}}$.
The open question here is whether the collapse consequence can possibly be
improved to $\text{ZP}^{\text{NP}}$. This is one reason to consider classes that lie between
NP/poly $\cap$ co-NP/poly and (NP $\cap$ co-NP)/poly.

5.1 A lowness result for NPMV$_1$/poly

It is shown in [11] that if an NP-complete problem is in NPMV$_1$/poly then
PH collapses to $\Sigma^p_2$. In [11] the authors actually state this result in terms of
overproductive reductions to sparse sets. We use ideas in their proof to show
the underlying lowness result for functions: all word-decreasing self-reducible
functions in NPMV$_1$/poly are low for $\Sigma^p_2$. We first recall the definition of word-
\textit{decreasing self-reducible sets} (and define its obvious extension to total single-
\textit{valued functions}).

\textbf{Definition 2. [7]} For strings $x, y \in \Sigma^*$, $x \prec y$ if $|x| < |y|$ or $|x| = |y|$ and $x$
is lexicographically smaller than $y$. A set $A$ is word-decreasing self-reducible if
there is a polynomial-time oracle machine $M$ such that $A = L(M^A)$, where on
any input $x$ the machine $M$ queries the oracle only about strings $y$ such that
$y \prec x$. Similarly, a total single-valued function $f$ on $\Sigma^*$ is word-decreasing self-
\textit{reducible} if there is a polynomial-time oracle transducer $T$ such that $T$ computes $f$,
where on any input $x$, transducer $T$ can query the oracle only about strings $y$
such that $y \prec x$.

The definition of lowness extends naturally to total, single-valued functions:
A functional oracle $f$ returns $f(x)$ on query $x$. For any relativizable complexity
class $\mathcal{C}$ we say that $f \in \text{Low}(\mathcal{C})$ if $\mathcal{C}^f = \mathcal{C}$. We show next that self-reducible sets
and self-reducible functions in NPMV/poly have identical lowness properties.
Hence it suffices to prove lowness of self-reducible sets in NPMV/poly.

\textbf{Theorem 4.} Let $\mathcal{F}$ contain all self-reducible functions in any of the four function
classes \{NPMV/poly, NPSV/poly, NPMV$_1$/poly, NPSV$_1$/poly\}. Let $\mathcal{C}$ be the
subclass of $\mathcal{F}$ consisting of characteristic functions (making $\mathcal{C}$ a language
class, essentially). For every self-reducible function \( f \in \mathcal{F} \) there is a self-reducible set \( A \in \mathcal{C} \) such that \( f \) and \( A \) are polynomial-time Turing equivalent.

**Proof.** Given \( f \in \mathcal{F} \), we can define the corresponding set \( A \in \mathcal{C} \) by suitably encoding, for each \( x \), the bits of \( f(x) \) in \( A \). We can easily ensure that the self-reducibility of \( f \) carries over to \( A \) and \( f \) and \( A \) are polynomial-time Turing equivalent.

**Theorem 5.** Word-decreasing self-reducible sets in NPMV\(_i\)/poly are low for \( \Sigma^p_2 \).

Since \( \Sigma^p_k, \Pi^p_k, \text{PP}, \text{C}_\text{-P}, \text{Mod}_m \text{P}, \text{PSPACE}, \text{and EXP} \) have many-one complete word-decreasing self-reducible sets [7], the following corollary is immediate.

**Corollary 4.** If \( \mathcal{C} \in \{ \Sigma^p_k, \Pi^p_k, \text{PP}, \text{C}_\text{-P}, \text{Mod}_m \text{P}, \text{PSPACE}, \text{EXP} \} \), for \( k \geq 1 \), has a complete set in NPMV\(_i\)/poly then \( \mathcal{C} \subseteq \Sigma^p_2 \) and \( \text{PH} = \Sigma^p_2 \).

The proof follows since for each \( \mathcal{C} \in \{ \Sigma^p_k, \Pi^p_k, \text{PP}, \text{C}_\text{-P}, \text{Mod}_m \text{P}, \text{PSPACE}, \text{EXP} \} \) and any set \( A \) complete for \( \mathcal{C} \) w.r.t. polynomial-time Turing reductions we have \( \Sigma^p_3 \subseteq \Sigma^p_2 \).

We end this section with the observation that \( \text{AM} \cap \text{coAM} \) is contained in NPMV\(_i\)/poly. It is interesting to now compare the lowness results (Theorems 1 and 5) for these classes.

**Proposition 2.** If \( L \in \text{AM} \cap \text{coAM} \) then \( L \) is in NPMV\(_i\)/poly.

**Proof.** Given \( L \in \text{AM} \cap \text{coAM} \), as already observed in an earlier proof by probability amplification techniques and quantifier swapping, there are NP sets \( A \) and \( B \) and a polynomial \( p \) such that \( \forall x : |x| \leq m \), there is a subset \( S \subseteq \{0,1\}^{p(m)} \) of size \( |S| \geq 2^{p(m)}-1 \) with the following property: \( x \in L \) implies

\[
\forall w : \langle x, w \rangle \in A \quad \text{and} \quad \forall w \in S : \langle x, w \rangle \notin B
\]

and \( x \notin L \) implies

\[
\forall w : \langle x, w \rangle \in B \quad \text{and} \quad \forall w \in S : \langle x, w \rangle \notin A.
\]

We can combine the NP machines for \( A \) and \( B \) and build a transducer \( I \) that takes pair \( \langle x, w \rangle \) as input, where \( w \) is the advice string. Observe that \( S \) constitutes the set of \( w \)’s that are correct advice strings. Using a \( w \in S \) membership in \( L \) for strings of length \( m \) can be decided and for such advice strings the transducer \( I \) will always yield a single-valued, total computation for all inputs of length \( m \), outputting either 1 or 0 depending on the membership of input \( x \). Notice that the above properties also already imply \( L \) is in NPMV\(_i\)/poly, because no matter which \( w \in \{0,1\}^{p(m)} \) is used as advice, \( \langle x, w \rangle \) is in the NP set \( A \) or in the NP set \( B \) and so the transducer \( I \) always outputs at least one of 0 or 1 for any advice string and any input.
5.2 A lowness result for NPSV/poly

In [11] it is left as an open problem to discover new lowness (or collapse consequence) results for NPSV/poly. As noted in [11], nothing better is known for NPSV/poly than the collapse consequence result: if SAT is in NPSV/poly then PH collapses to ZPP_\exp^P, which holds even for the larger class NP/poly \cap co-NP/poly [21].

We show that sets in NPSV/poly that are checkable, in the sense of program checking as defined by Blum and Kannan [9], are low for AM and for ZPP^{NP}. Since \oplus P, PP, PSPACE, and EXP have checkable complete problems, it follows that for any of these classes inclusion in NPSV/poly implies its containment in AM \cap coAM. This result is proved on the same lines as the Babai et al result [6]: If EXP is contained in P/poly then EXP \subseteq MA.

Recall the definitions of MIP[C] and IP[C] for a class C of languages. We prove a technical lemma that immediately yields the lowness result.

**Lemma 1.** If A ∈ NPSV/poly then MIP[A] \subseteq AM.

**Proof.** Let L ∈ MIP[A] for some set A ∈ NPSV/poly. Let T be the nondeterministic transducer that witnesses that A ∈ NPSV/poly. We describe an MAM protocol for L:

1. Let x be an input of length n to the protocol. Let m = p(n), where p is a polynomial bounding the size of the queries to A made by the verifier during the protocol for inputs of length n.
2. **Merlin** sends advice w of length q(m) to Arthur.
3. **Arthur** sends a polynomial random string r (used for simulating the original IP protocol) to Merlin.
4. **Merlin** sends back the list of successive queries to set A (generated by simulating the original IP protocol with random string r), the list of answers to those queries along with the computation paths of transducer T with advice w that certify the answers to the queries.
5. **Arthur** can verify in polynomial time that Merlin’s message is all correct and accept if and only if the original IP protocol accepts.

By the fact that T computes a single-valued partial function for any advice w, although the verifier is simulating the nondeterministic transducer T, it is guaranteed that each accepting computation path has identical output and hence does identical computation. Thus, what makes the above MAM protocol work is the fact that for any advice w and query q all accepting computation paths of T(q, w) output the same value. So, regardless of which computation paths are sent to Arthur by Merlin in Step 4 of the above protocol, Arthur’s decision will be the same. In other words, Arthur’s acceptance depends only on the random string r, hence exactly preserving the acceptance probability of the original IP protocol.

Standard techniques (cf. [4]) can be used to convert the MAM protocol to an AM protocol. This completes the proof.
We have an immediate consequence of the following lowness result.

**Theorem 6.** If L is a checkable set in NPSV/poly then \( L \in \text{AM} \cap \text{coAM} \) and hence low for AM and ZPP\(^{\text{NP}}\).

**Proof.** The assumption in the theorem's statement implies that both \( L \) and \( \overline{L} \) are in \( \text{MIP}[L] \) by the checker characterization theorem of [9]. Now, applying Lemma 1 yields that both \( L \) and \( \overline{L} \) are in AM and the result follows.

We can derive new collapse consequences as corollary, since the classes \( \oplus \text{P}, \text{PP}, \text{PSPACE}, \) and \( \text{EXP} \) all have checkable complete problems. It follows that for any of these classes inclusion in NPSV/poly implies its containment in \( \text{AM} \cap \text{coAM} \).

**Corollary 5.** If any of the classes \( \oplus \text{P}, \text{PP}, \text{PSPACE}, \) and \( \text{EXP} \) is contained in NPSV/poly then it is low for \( \text{AM} \) and hence \( \text{PH} = \text{AM} \).

Notice that we have the same lowness for checkable functions in NPSV/poly.

**Theorem 7.** Checkable functions in NPSV/poly are low for \( \text{AM} \) and \( \text{ZPP}^{\text{NP}} \).

**Proof.** Let \( f \) be a checkable function in NPSV/poly. We can suitably encode, for each \( x \), the bits of \( f(x) \) in a language \( A \) which is polynomial-time Turing equivalent to \( f \) and hence \( A \) is also checkable. The lowness result now follows by invoking Theorem 6.

**Acknowledgements.** The first author was partially supported by an Alexander von Humboldt fellowship in the year 1990, and he is grateful to Prof. Uwe Schöning for hosting his visit to Ulm university where this work was carried out.

**References**