On Boolean Lowness and Boolean Highness

Steffen Reith Klaus W. Wagner
Lehrstuhl für Theoretische Informatik
Universität Würzburg
[streit,wagner]@informatik.uni-wuerzburg.de

March 20, 1998

Abstract

The concepts of lowness and highness originate from recursion theory and were introduced into the complexity theory by Schöning [Sch85]. Informally, a set is low (high, resp.) for a relativizable class $\mathcal{K}$ of languages if it does not add (adds maximal, resp.) power to $\mathcal{K}$ when used as an oracle. In this paper we introduce the notions of boolean lowness and boolean highness. Informally, a set is boolean low (boolean high, resp.) for a class $\mathcal{K}$ of languages if it does not add (adds maximal, resp.) power to $\mathcal{K}$ when combined with $\mathcal{K}$ by boolean operations. We prove properties of boolean lowness and boolean highness which show a lot of similarities with the notions of lowness and highness. Using Kadin's technique of hard strings (see [Kad88, Wag87, CK96, BCO93]) we show that the sets which are boolean low for the classes of the boolean hierarchy are low for the boolean closure of Σ_2^p. Furthermore, we prove a result on boolean lowness which has as a corollary the best known result (see [BCO93]; in fact even a bit better) on the connection of the collapses of the boolean hierarchy and the polynomial-time hierarchy: If $BH = NP(k)$ then $PH = \Sigma_2^p(k-1) \oplus NP(k)$.

Keywords: Computational complexity, lowness, highness, boolean lowness, boolean highness, boolean hierarchy, polynomial-time hierarchy, hard/easy, advice, collapse.

1 Introduction

The concept of lowness and highness was originally studied in a recursion theoretic context ([Coo74] and [Soa74]). At that time the question arose how to measure the content of information of an oracle, used by a Turing machine. An oracle was called *low* for a given class $\mathcal{K}$ if it does not add power to the machines accepting sets from $\mathcal{K}$. It was called *high* for $\mathcal{K}$ if it adds (in a sense) maximal power to $\mathcal{K}$. These notions have been studied particularly in the context of the arithmetic hierarchy.

Many ideas and concepts were translated from recursion theory into the terms of complexity theory. So Schöning ([Sch83],[Sch85]) introduced the notions of lowness and highness into complexity theory. In the context of the polynomial-time hierarchy he defined $Low_k^p =_{\mathrm{df}} \{A \in NP | (\Sigma_k^p)^A = \Sigma_k^p\}$ and $High_k^p =_{\mathrm{df}} \{A \in NP | (\Sigma_k^p)^A = \Sigma_{k+1}^p\}$ for $k \geq 0$. It was shown in [Sch85] that

- $Low_k^p \subseteq Low_{k+1}^p$ and $High_k^p \subseteq High_{k+1}^p$ for any $k \geq 0$,
- $PH = \Sigma_k^p \Leftrightarrow Low_k^p = NP \Leftrightarrow High_k^p = NP \Leftrightarrow Low_k^p \cap High_k^p \neq \emptyset$ for any $k \geq 0$, and
- $Low_0^p = P$, $Low_1^p = NP \cap co\text{-}NP$ and $High_0^p = \{A | A \text{ is } \leq_T^{\mathrm{p}}\text{-complete for } NP\}$.

In Section 3 we introduce the notions of boolean lowness and boolean highness. Informally, a set is boolean low (boolean high, resp.) for a class $\mathcal{K}$ of languages if it does not add (adds maximal, resp.)

power to $\mathcal{K}$ when combined with $\mathcal{K}$ by boolean operations. We make this precise in the context of the classes $NP(k)$ of the boolean hierarchy. Let $\mathcal{R}^p_m(A)$ be the class of all languages which are $\leq^p_m$-reducible to A and let $\hat{\mathcal{R}}^p_m(A) =_{\rm df} \mathcal{R}^p_m(A) \cup P$. For classes $\mathcal{K}$ and $\mathcal{K}'$ let further *co*-$\mathcal{K} =_{\rm df} \{\overline{A} | A \in \mathcal{K}\}$, $\mathcal{K} \wedge \mathcal{K}' =_{\rm df} \{A \cap B | A \in \mathcal{K}, B \in \mathcal{K}'\}$, and $\mathcal{K} \vee \mathcal{K}' =_{\rm df} \{A \cup B | A \in \mathcal{K}, B \in \mathcal{K}'\}$. For $k \geq 0$, a set $A \in NP$ is in low^p_k iff $NP(k) \wedge \mathcal{R}^p_m(A) = NP(k) \vee \mathcal{R}^p_m(A) = NP(k) \wedge co\text{-}\mathcal{R}^p_m(A) = NP(k) \vee co\text{-}\mathcal{R}^p_m(A) = NP(k)$; a set $A \in NP$ is in $high^p_k$ iff $NP(k) \wedge \mathcal{R}^p_m(A) = NP(k) \wedge NP$, $NP(k) \vee \mathcal{R}^p_m(A) = NP(k) \vee NP$, $NP(k) \wedge co\text{-}\mathcal{R}^p_m(A) = NP(k) \wedge co\text{-}NP$, and $NP(k) \vee co\text{-}\mathcal{R}^p_m(A) = NP(k) \vee co\text{-}NP$. For the classes low^p_k and $high^p_k$ we prove results which are similar to those for the classes Low^p_k and $High^p_k$:

- $low^p_k \subseteq low^p_{k+1}$, $high^p_k \subseteq high^p_{k+2}$, and $high^p_{2k} \subseteq high^p_{2k+1}$ for any $k \geq 0$,
- $BH = NP(k) \Leftrightarrow low^p_k = NP \Leftrightarrow high^p_k = NP \Leftrightarrow low^p_k \cap high^p_k \neq \emptyset$ for any $k \geq 0$, and
- $low^p_0 = P$, $low^p_1 = NP \cap co\text{-}NP$, and $high^p_0 = \{A | A \text{ is } \leq^{\rm p}_m\text{-complete for } NP\}$ if $P \neq NP$.

In Section 4 we relate boolean lowness to lowness. Using Kadin's technique of hard strings (see [Kad88, Wag87, CK96, BCO93]) we prove that $A \in low^p_k$ implies $(\Sigma^p_2)^A \subseteq \Sigma^p_2(2k-1)$ where $\Sigma^p_2(k)$ denotes the k-th level of the boolean hierarchy over Σ^p_2. Hence every low^p_k set is low for the boolean closure of Σ^p_2, and consequently $low^p_k \subseteq Low^p_3$ for all $k \geq 0$.

These results have interesting consequences to the connection between the collapses of the boolean hierarchy and the polynomial-time hierarchy. Kadin [Kad88] showed that $BH = NP(k)$ implies $PH = \Delta^p_3$. This was improved in [Wag87] where Δ^p_3 could be replaced by the boolean closure of Σ^p_2 and independently in [CK96] where Δ^p_3 was replaced by $\Sigma^p_2(k)$. Eventually, Beigel, Chang, and Ogihara [BCO93] proved that $BH = NP(k)$ implies the collapse of the polynomial-time hierarchy to the class of all languages that are computable in polynomial time with $k-1$ parallel queries to a Σ^p_2 set and an unbounded number of queries in NP. Adapting their method we prove in Section 5 that $low^p_k \cap High^p_1 \neq \emptyset$ implies $PH = \Sigma^p_2(k-1) \oplus \Delta^p_2$ for $k \geq 1$, where $\mathcal{K} \oplus \mathcal{K}' =_{\rm df} \{A \triangle B | A \in \mathcal{K}, B \in \mathcal{K}'\}$. From this and a recent result by Chang [Cha97] one can conclude: If $BH = NP(k)$ then $PH = \Sigma^p_2(k-1) \oplus NP(k)$, which even slightly improves the [BCO93] result.

2 Preliminaries

With P^B (NP^B) we denote the class of all languages accepted by a deterministic polynomial time (nondeterministic polynomial time, resp.) oracle machine using the oracle B. If the queries to an oracle depend on the answers to previous queries, we will call them *adaptive* queries. The queries are said to be made in *parallel*, if a list of all queries is calculated before the machine asks the oracle. The class of all languages accepted by deterministic polynomial time (nondeterministic polynomial time) oracle machines making only parallel queries to an oracle B will be denoted by $P^B_{\|}$ ($NP^B_{\|}$). For a class $\mathcal{K}$ of languages we define $P^{\mathcal{K}} =_{\rm df} \bigcup_{B \in \mathcal{K}} P^B$, $NP^{\mathcal{K}} =_{\rm df} \bigcup_{B \in \mathcal{K}} NP^B$ and $P^{\mathcal{K}}_{\|} =_{\rm df} \bigcup_{B \in \mathcal{K}} P^B_{\|}$.

For a function $r : \mathbb{N} \mapsto \mathbb{N}$ we denote by $P^B[r]$ the class of languages accepted by some deterministic polynomial time oracle machine asking the oracle only $r(n)$ times for an input of length n. Similarly let $P^B_{\|}[r]$ be the class of languages accepted by deterministic polynomial oracle machines making only $r(n)$ parallel queries to B.

For classes $\mathcal{K}$ and $\mathcal{K}'$ of languages we define *co*-$\mathcal{K} =_{\rm df} \{\overline{A} | A \in \mathcal{K}\}$, $\mathcal{K} \wedge \mathcal{K}' =_{\rm df} \{A \cap B | A \in \mathcal{K}, B \in \mathcal{K}'\}$, $\mathcal{K} \vee \mathcal{K}' =_{\rm df} \{A \cup B | A \in \mathcal{K}, B \in \mathcal{K}'\}$, and $\mathcal{K} \oplus \mathcal{K}' =_{\rm df} \{A \triangle B | A \in \mathcal{K}, B \in \mathcal{K}'\}$. The *boolean hierarchy* over a complexity class $\mathcal{K} \supseteq P$ consists of the classes $\mathcal{K}(k)$ and *co*-$\mathcal{K}(k)$ for $k = 0, 1, \ldots$ which are defined inductively by $\mathcal{K}(0) =_{\rm df} P$ and $\mathcal{K}(k+1) =_{\rm df}$ *co*-$\mathcal{K}(k) \wedge \mathcal{K}$. Furthermore let $BH(\mathcal{K}) =_{\rm df} \bigcup_{k \geq 0} \mathcal{K}(k)$.

With $\mathcal{K} = NP$ we obtain the well known boolean hierarchy over NP (see [CH86, WW85, Köb85, KSW87, CGH+88]). For convenience we set $BH =_{\mathrm{df}} BH(NP)$.

We will need the following lemma which is proved for $\mathcal{K} = NP$ in [KSW87]. The proofs remain valid in the general case. The equality in statement 5 can be found in [Wag97].

Lemma 1 *Let $\mathcal{K} \supseteq P$ be closed under union and intersection, and let $k \geq 0$.*

1. $\mathcal{K}(2k+1) = \mathcal{K}(2k) \vee \mathcal{K}$
2. $\mathcal{K}(2k+2) = \mathcal{K}(2k+1) \wedge co\text{-}\mathcal{K}$
3. $\mathcal{K}(k+2) = \mathcal{K}(k) \vee (\mathcal{K} \wedge co\text{-}\mathcal{K}) = \mathcal{K}(k) \wedge (\mathcal{K} \vee co\text{-}\mathcal{K})$
4. $\mathcal{K}(k) = \underbrace{\mathcal{K} \oplus \mathcal{K} \oplus \ldots \oplus \mathcal{K}}_{k \text{ times}}$
5. $\mathcal{K}(k) \subseteq P_{\|}^{\mathcal{K}}[k] = P \oplus \mathcal{K}(k) \subseteq \mathcal{K}(k+1)$

The *polynomial-time hierarchy* consists of the classes Σ_k^p, Π_k^p, Δ_k^p and Θ_k^p by defining inductively $\Sigma_0^p =_{\mathrm{df}} \Pi_0^p =_{\mathrm{df}} \Delta_0^p =_{\mathrm{df}} \Theta_0^p =_{\mathrm{df}} P$ and $\Sigma_{k+1}^p =_{\mathrm{df}} NP^{\Sigma_k^p}$, $\Pi_{k+1}^p =_{\mathrm{df}} co\text{-}\Sigma_{k+1}^p$, $\Delta_{k+1}^p =_{\mathrm{df}} P^{\Sigma_k^p}$, $\Theta_{k+1}^p =_{\mathrm{df}} P^{\Sigma_k^p}[O(\log n)]$. Obviously $\Sigma_k^p \cup \Pi_k^p \subseteq BH(\Sigma_k^p) \subseteq \Theta_{k+1}^p \subseteq \Delta_{k+1}^p \subseteq \Sigma_{k+1}^p \cap \Pi_{k+1}^p$ for all $k \geq 0$. Finally let $PH =_{\mathrm{df}} \bigcup_{k \geq 0} \Sigma_k^p$.

3 Definitions and properties

In a general way, one can define the notions of lowness and highness as follows:

Definition 2

- *For any relativizable class $\mathcal{K}$, the set A is* low for $\mathcal{K}$ *if and only if $\mathcal{K}^A = \mathcal{K}$.*
  Let $Low(\mathcal{K})$ be the class of all sets which are low for $\mathcal{K}$.
- *For any relativizable class $\mathcal{K}$ and any class $\mathcal{M}$, the set $A \in \mathcal{M}$ is* high for $\mathcal{K}$ with respect to $\mathcal{M}$ *if and only if $\mathcal{K}^A = \mathcal{K}^{\mathcal{M}}$.*
  Let $High(\mathcal{K}, \mathcal{M})$ be the class of all sets which are high for $\mathcal{K}$ with respect to $\mathcal{M}$.

Schöning introduced in [Sch85] the classes $Low_k^p =_{\mathrm{df}} Low(\Sigma_k^p) \cap NP$ and $High_k^p =_{\mathrm{df}} High(\Sigma_k^p, NP)$ for $k \geq 0$, where $(\Sigma_0^p)^A =_{\mathrm{df}} P^A$ and $(\Sigma_{k+1}^p)^A =_{\mathrm{df}} NP^{(\Sigma_k^p)^A}$. The following facts are known about the classes Low_k^p and $High_k^p$.

Theorem 3 ([Sch85]) *For $k \geq 0$, the following are equivalent:*

- $PH = \Sigma_k^p$
- $Low_k^p = NP$
- $High_k^p = NP$
- $Low_k^p \cap High_k^p \neq \emptyset$

Theorem 4 ([Sch85])

1. $Low_0^p = P$
2. $Low_1^p = NP \cap co\text{-}NP$
3. $High_0^p = \{A | A \leq_T^{\mathrm{p}} \text{-complete for } NP\}$

Theorem 5 ([Sch85]) *For all $k \geq 0$,*

1. $Low_k^p \subseteq Low_{k+1}^p$
2. $High_k^p \subseteq High_{k+1}^p$

Now we want to have notions of boolean lowness and boolean highness, i.e. notions which are not based on oracle constructions (which build the polynomial-time hierarchy) but on the boolean construction which build the boolean hierarchy. Let $\mathcal{R}_m^p(A)$ be the class of all languages which are $\leq_m^p$-reducible to A and let $\hat{\mathcal{R}}_m^p(A) =_{\mathrm{df}} \mathcal{R}_m^p(A) \cup P$. Note that $\hat{\mathcal{R}}_m^p(A) = \mathcal{R}_m^p(A)$ if $A \neq \emptyset$ and $\overline{A} \neq \emptyset$, and that $\hat{\mathcal{R}}_m^p(A) = P$ otherwise.

Definition 6

- *For any class $\mathcal{K} \supseteq P$, the set A is* boolean low for $\mathcal{K}$ *if and only if $\mathcal{K} \wedge \hat{\mathcal{R}}_m^p(A) = \mathcal{K} \vee \hat{\mathcal{R}}_m^p(A) = \mathcal{K} \wedge co\text{-}\hat{\mathcal{R}}_m^p(A) = \mathcal{K} \vee co\text{-}\hat{\mathcal{R}}_m^p(A) = \mathcal{K}$.*
  Let $low(\mathcal{K})$ be the class of all sets which are boolean low for $\mathcal{K}$.
- *For any class $\mathcal{K} \supseteq P$ and any $\mathcal{M} \supseteq P$ which is closed under $\leq_m^{\mathrm{p}}$, the set $A \in \mathcal{M}$ is* boolean high for $\mathcal{K}$ with respect to $\mathcal{M}$ *if and only if $\mathcal{K} \wedge \hat{\mathcal{R}}_m^p(A) = \mathcal{K} \wedge \mathcal{M}$, $\mathcal{K} \vee \hat{\mathcal{R}}_m^p(A) = \mathcal{K} \vee \mathcal{M}$, $\mathcal{K} \wedge co\text{-}\hat{\mathcal{R}}_m^p(A) = \mathcal{K} \wedge co\text{-}\mathcal{M}$, and $\mathcal{K} \vee co\text{-}\hat{\mathcal{R}}_m^p(A) = \mathcal{K} \vee co\text{-}\mathcal{M}$.*
  Let $high(\mathcal{K}, \mathcal{M})$ be the class of all sets which are boolean high for $\mathcal{K}$ with respect to $\mathcal{M}$.

The following properties are easy to prove.

Proposition 7 *Let $\mathcal{K} \supseteq P$ and $\mathcal{M} \supseteq P$. For sets A and B such that $A \leq_m^p B$,*

1. *If $B \in low(\mathcal{K})$ then $A \in low(\mathcal{K})$.*
2. *If $A \in high(\mathcal{K}, \mathcal{M})$ and $B \in \mathcal{M}$ then $B \in high(\mathcal{K}, \mathcal{M})$.*

Proposition 8 *Let $\mathcal{K} \supseteq P$ be closed under union, intersection, and $\leq_m^{\mathrm{p}}$.*

1. *$low(\mathcal{K}) = \mathcal{K} \cap co\text{-}\mathcal{K}$.*
2. *If $\mathcal{M} \cup co\text{-}\mathcal{M} \subseteq \mathcal{K}$ then $high(\mathcal{K}, \mathcal{M}) = \mathcal{M}$.*

Proof: First let $A \in low(\mathcal{K})$. Thus $\mathcal{K} \wedge \hat{\mathcal{R}}_m^p(A) = \mathcal{K}$ and $\mathcal{K} \wedge co\text{-}\hat{\mathcal{R}}_m^p(A) = \mathcal{K}$ giving us $\overline{\emptyset} \wedge A = A \in \mathcal{K}$ and $\overline{\emptyset} \wedge \overline{A} = \overline{A} \in \mathcal{K}$. Now let $A \in \mathcal{K} \cap co\text{-}\mathcal{K}$. Then $\mathcal{K} \subseteq \mathcal{K} \wedge \hat{\mathcal{R}}_m^p(A) \subseteq \mathcal{K} \wedge \mathcal{K} = \mathcal{K}$ and $\mathcal{K} \subseteq \mathcal{K} \wedge co\text{-}\hat{\mathcal{R}}_m^p(A) \subseteq \mathcal{K} \wedge \mathcal{K} = \mathcal{K}$. The other two cases can be done analogously.
The first direction follows immediately by definition 6. Conversely let $A \in \mathcal{M}$ giving $\mathcal{K} \wedge \mathcal{M} \subseteq \mathcal{K} \wedge \mathcal{K} \subseteq \mathcal{K} \subseteq \mathcal{K} \wedge \hat{\mathcal{R}}_m^p(A) \subseteq \mathcal{K} \wedge \mathcal{M}$ and $\mathcal{K} \wedge co\text{-}\mathcal{M} \subseteq \mathcal{K} \wedge \mathcal{K} \subseteq \mathcal{K} \subseteq \mathcal{K} \wedge co\text{-}\hat{\mathcal{R}}_m^p(A) \subseteq \mathcal{K} \wedge co\text{-}\mathcal{M}$. Again the other two cases can be done similarly. ❏

Consequently, boolean lowness and boolean highness are interesting mainly for classes $\mathcal{K}$ which are not closed under union and intersection. The classes of the boolean hierarchy (besides the levels 0 and 1) most likely have this properties. For them we define special classes low_k^p and $high_k^p$ as analogues for the classes Low_k^p and $High_k^p$ for the polynomial-time hierarchy.

Definition 9 *For $k \geq 0$,*

1. $low_k^p =_{\mathrm{df}} low(NP(k)) \cap NP$
2. $high_k^p =_{\mathrm{df}} high(NP(k), NP)$

Boolean lowness and boolean highness for the class $NP(k)$ are strongly connected with collapse properties for the boolean hierarchy.

Theorem 10 *For $k \geq 0$, the following are equivalent:*

- $BH = NP(k)$
- $low_k^p = NP$
- $high_k^p = NP$
- $low_k^p \cap high_k^p \neq \emptyset$

The next result exhibits the nature of the classes low_0^p, low_1^p and $high_0^p$.

Theorem 11

1. $low_0^p = P$
2. $low_1^p = NP \cap co\text{-}NP$
3. $high_0^p = \begin{cases} \{A | A \text{ is } \leq_m^{\mathrm{p}}\text{-complete for } NP\} & \text{if } P \neq NP \\ NP & \text{if } P = NP \end{cases}$

Proof: The statements 1 and 2 follow directly from Lemma 8. The case $P = NP$ of statement 3 can be done by Theorem 10. Now let $P \neq NP$. It's clear that any $\leq_m^{\mathrm{p}}$-complete set A is in $high_0^p$ since $\hat{\mathcal{R}}_m^p(A) = NP$. Let $A \in high_0^p$. Since $P \subseteq \hat{\mathcal{R}}_m^p(A)$ we obtain $\hat{\mathcal{R}}_m^p(A) = P \wedge \hat{\mathcal{R}}_m^p(A) = P \wedge NP = NP$ showing that A is $\leq_m^{\mathrm{p}}$-complete for NP. ❏

Thus these classes are similar to the corresponding Low^p and $High^p$ classes. More precisely: $low_0^p = Low_0^p$, $low_1^p = Low_1^p$ and $high_0^p \subseteq High_0^p$. The next lemma demonstrates, that the inclusion structure of the low^p classes ($high^p$ classes, resp.) is similar to that of Low^p classes ($High^p$ classes, resp.).

Theorem 12 *For all $k \geq 0$,*

1. $low_k^p \subseteq low_{k+1}^p$
2. $high_k^p \subseteq high_{k+2}^p$
3. $high_{2k}^p \subseteq high_{2k+1}^p$

Proof: First let $A \in low_k^p$. We know that $NP(k) \left\{ \begin{smallmatrix} \wedge \\ \vee \end{smallmatrix} \right\} \left\{ \begin{smallmatrix} \hat{\mathcal{R}}_m^p(A) \\ co\text{-}\hat{\mathcal{R}}_m^p(A) \end{smallmatrix} \right\} = NP(k)$ and $A \in NP$. Using Lemma 1 and the definition of the boolean hierarchy of NP the following lines show that $low_k^p \subseteq low_{k+1}^p$.

$$NP(k+1) \wedge \hat{\mathcal{R}}_m^p(A) = co\text{-}NP(k) \wedge NP \wedge \hat{\mathcal{R}}_m^p(A) = co\text{-}NP(k) \wedge NP = NP(k+1)$$

$$\begin{aligned} NP(k+1) \wedge co\text{-}\hat{\mathcal{R}}_m^p(A) &= co\text{-}NP(k) \wedge NP \wedge co\text{-}\hat{\mathcal{R}}_m^p(A) = co\text{-}(NP(k) \vee \hat{\mathcal{R}}_m^p(A)) \wedge NP \\ &= co\text{-}NP(k) \wedge NP = NP(k+1) \end{aligned}$$

$$\begin{aligned} NP(k+1) \vee \hat{\mathcal{R}}_m^p(A) &= (co\text{-}NP(k) \wedge NP) \vee \hat{\mathcal{R}}_m^p(A) \\ &\subseteq (co\text{-}NP(k) \vee \hat{\mathcal{R}}_m^p(A)) \wedge (NP \vee \hat{\mathcal{R}}_m^p(A)) \\ &= co\text{-}(NP(k) \wedge co\text{-}\hat{\mathcal{R}}_m^p(A)) \wedge NP \\ &= co\text{-}NP(k) \wedge NP = NP(k+1) \end{aligned}$$

Now we have to distinguish the odd and the even case:
Case $k = 2i$:

$$NP(2i+1) \vee co\text{-}\hat{\mathcal{R}}^p_m(A) \;=\; NP(2i) \vee NP \vee co\text{-}\hat{\mathcal{R}}^p_m(A) = NP(2i) \vee NP = NP(2i+1)$$

Case $k = 2i + 1$:

$$\begin{aligned} NP(2i+2) \vee co\text{-}\hat{\mathcal{R}}^p_m(A) &= (NP(2i+1) \wedge co\text{-}NP) \vee co\text{-}\hat{\mathcal{R}}^p_m(A) \\ &\subseteq (NP(2i+1) \vee co\text{-}\hat{\mathcal{R}}^p_m(A)) \wedge (co\text{-}NP \vee co\text{-}\hat{\mathcal{R}}^p_m(A)) \\ &= NP(2i+1) \wedge co\text{-}NP \\ &= NP(2i+2) \end{aligned}$$

If $A \in high^p_k$ the equations $NP(k) \left\{ \begin{smallmatrix} \wedge \\ \vee \end{smallmatrix} \right\} \left\{ \begin{smallmatrix} \hat{\mathcal{R}}^p_m(A) \\ co\text{-}\hat{\mathcal{R}}^p_m(A) \end{smallmatrix} \right\} = NP(k) \left\{ \begin{smallmatrix} \wedge \\ \vee \end{smallmatrix} \right\} \left\{ \begin{smallmatrix} NP \\ co\text{-}NP \end{smallmatrix} \right\}$hold. By Lemma 1(3) one can conclude:

$$\begin{aligned} NP(k+2) \vee \hat{\mathcal{R}}^p_m(A) &= NP(k) \vee (NP \wedge co\text{-}NP) \vee \hat{\mathcal{R}}^p_m(A) \\ &= (NP(k) \vee \hat{\mathcal{R}}^p_m(A)) \vee (NP \wedge co\text{-}NP) \\ &= NP(k) \vee NP \vee (NP \wedge co\text{-}NP) \\ &= NP(k+2) \vee NP \end{aligned}$$

$$\begin{aligned} NP(k+2) \vee co\text{-}\hat{\mathcal{R}}^p_m(A) &= NP(k) \vee (NP \wedge co\text{-}NP) \vee co\text{-}\hat{\mathcal{R}}^p_m(A) \\ &= (NP(k) \vee co\text{-}\hat{\mathcal{R}}^p_m(A)) \vee (NP \wedge co\text{-}NP) \\ &= NP(k) \vee co\text{-}NP \vee (NP \wedge co\text{-}NP) \\ &= NP(k+2) \vee co\text{-}NP \end{aligned}$$

$$\begin{aligned} NP(k+2) \wedge \hat{\mathcal{R}}^p_m(A) &= NP(k) \wedge (NP \vee co\text{-}NP) \wedge \hat{\mathcal{R}}^p_m(A) \\ &= (NP(k) \wedge \hat{\mathcal{R}}^p_m(A)) \wedge (NP \vee co\text{-}NP) \\ &= NP(k) \wedge NP \wedge (NP \vee co\text{-}NP) \\ &= NP(k+2) \wedge NP \end{aligned}$$

$$\begin{aligned} NP(k+2) \wedge co\text{-}\hat{\mathcal{R}}^p_m(A) &= NP(k) \wedge (NP \vee co\text{-}NP) \wedge co\text{-}\hat{\mathcal{R}}^p_m(A) \\ &= (NP(k) \wedge co\text{-}\hat{\mathcal{R}}^p_m(A)) \wedge (NP \vee co\text{-}NP) \\ &= NP(k) \wedge co\text{-}NP \wedge (NP \vee co\text{-}NP) \\ &= NP(k+2) \wedge co\text{-}NP \end{aligned}$$

Hence we have $A \in high^p_{k+2}$.

If $A \in high^p_{2k}$ the equations $NP(2k) \left\{ \begin{smallmatrix} \wedge \\ \vee \end{smallmatrix} \right\} \left\{ \begin{smallmatrix} \hat{\mathcal{R}}^p_m(A) \\ co\text{-}\hat{\mathcal{R}}^p_m(A) \end{smallmatrix} \right\} = NP(2k) \left\{ \begin{smallmatrix} \wedge \\ \vee \end{smallmatrix} \right\} \left\{ \begin{smallmatrix} NP \\ co\text{-}NP \end{smallmatrix} \right\}$hold. By Lemma 1(1) we get:

$$\begin{aligned} NP(2k+1) \wedge \hat{\mathcal{R}}^p_m(A) &= co\text{-}NP(2k) \wedge NP \wedge \hat{\mathcal{R}}^p_m(A) = co\text{-}NP(2k) \wedge NP \\ &= (co\text{-}NP(2k) \wedge NP) \wedge NP \\ &= NP(2k+1) \wedge NP \end{aligned}$$

$$\begin{aligned} NP(2k+1) \wedge co\text{-}\hat{\mathcal{R}}^p_m(A) &= co\text{-}NP(2k) \wedge NP \wedge co\text{-}\hat{\mathcal{R}}^p_m(A) = co\text{-}(NP(2k) \vee \hat{\mathcal{R}}^p_m(A)) \wedge NP \\ &= co\text{-}(NP(2k) \vee NP) \wedge NP = (co\text{-}NP(2k) \wedge co\text{-}NP) \wedge NP \\ &= NP(2k+1) \wedge co\text{-}NP \end{aligned}$$

$$\begin{aligned} NP(2k+1) \vee \hat{\mathcal{R}}^p_m(A) &= NP(2k) \vee NP \vee \hat{\mathcal{R}}^p_m(A) = NP(2k) \vee NP \\ &= (NP(2k) \vee NP) \vee NP \\ &= NP(2k+1) \vee NP \end{aligned}$$

$$\begin{aligned} NP(2k+1) \vee co\text{-}\hat{\mathcal{R}}^p_m(A) &= NP(2k) \vee NP \vee co\text{-}\hat{\mathcal{R}}^p_m(A) = (NP(2k) \vee co\text{-}\hat{\mathcal{R}}^p_m(A)) \vee NP \\ &= (NP(2k) \vee co\text{-}NP) \vee NP \\ &= NP(2k+1) \vee co\text{-}NP \end{aligned}$$

Consequently, $A \in high^p_{2k+1}$.

❏

Unfortunately, we are not able to prove $high^p_{2k+1} \subseteq high^p_{2k+2}$; even the easiest case $high^p_1 \subseteq high^p_2$. This is equivalent with proving that $NP \wedge co\text{-}\hat{\mathcal{R}}^p_m(A) = NP \wedge co\text{-}NP$ and $NP \vee co\text{-}\hat{\mathcal{R}}^p_m(A) = NP \vee co\text{-}NP$ implies $(NP \wedge co\text{-}NP) \vee \hat{\mathcal{R}}^p_m(A) = (NP \wedge co\text{-}NP) \vee NP$ and $(NP \wedge co\text{-}NP) \vee co\text{-}\hat{\mathcal{R}}^p_m(A) = (NP \wedge co\text{-}NP) \vee co\text{-}NP$.
By Theorem 11 it is obvious that the classes low^p_0 and low^p_1 are closed under complement. The next theorem shows the consequences of other low^p_k and $high^p_k$ classes to be closed under complement. The behavior of low^p_k and $high^p_k$ classes seems to differ from the one of Low^p_k and $High^p_k$ classes, which are obviously closed under complement.

Theorem 13

1. *For $k \geq 2$, $low^p_k = co\text{-}low^p_k \Leftrightarrow low^p_k = NP \cap co\text{-}NP$*
2. *For $k \geq 0$, $high^p_k = co\text{-}high^p_k \Leftrightarrow NP = co\text{-}NP$.*

Proof: The first statement is obvious because of $NP \cap co\text{-}NP \subseteq low^p_k \subseteq NP$ for $k \geq 2$. We prove the second statement:
"$\Rightarrow$" We know $\mathbf{SAT} \in high^p_0$ and $high^p_0 \subseteq high^p_k \subseteq co\text{-}high^p_k \subseteq co\text{-}NP$. Hence $NP = co\text{-}NP$.
"$\Leftarrow$" ($k \geq 1$) If $NP = co\text{-}NP$ then $BH = NP(k) = NP$ for all $k \geq 1$. By Theorem 10 we get $high^p_k = NP$ and hence $co\text{-}high^p_k = co\text{-}NP = NP = high^p_k$.
($k = 0$, $P \neq NP$) If $A \in co\text{-}high^p_0$ then, by Theorem 11, $\overline{A}$ is $\leq^{\mathrm{p}}_m$-complete for $NP = co\text{-}NP$. Hence A is $\leq^{\mathrm{p}}_m$-complete for NP, i.e. $A \in high^p_0$.
($k = 0$, $P = NP$) By Theorem 11 we get $high^p_0 = NP = co\text{-}NP = co\text{-}high^p_0$. ❏

4 low^p_k sets are low for $BH(\Sigma^p_2)$

In this section we show, that our low^p_k sets are low for $BH(\Sigma^p_2)$ and hence in Low^p_3. To prove this, we use similar *hard* and *easy* arguments as introduced in [Kad88] (see also: [CK96] and [BCO93]).

The projection function that selects the i^{th} to the j^{th} component of a k tuple will be denoted by $\langle x_1, \ldots, x_k \rangle_{(i,j)} = \langle x_i, \ldots, x_j \rangle$. The projection function of the i^{th} component will be denoted by the shortcut $\langle x_1, \ldots, x_k \rangle_{(i)} =_{\mathrm{df}} \langle x_1, \ldots, x_k \rangle_{(i,i)}$.

For every set A and $k \geq 0$ we define the set A_k by

$$\begin{aligned} A_0 &=_{\mathrm{df}} \overline{A} \\ A_{k+1} &=_{\mathrm{df}} \{\langle x_1, \ldots, x_{k+2} \rangle | \langle x_1, \ldots, x_{k+1} \rangle \in \overline{SAT_k} \wedge x_{k+2} \in \overline{A}\} \quad (k \geq 0) \end{aligned}$$

Lemma 14 *For $k \geq 0$ and $A \in NP \backslash \{\emptyset, \Sigma^*\}$, if $A \in low^p_{k+1}$ then $A_k \leq^{\mathrm{p}}_m \overline{\mathbf{SAT}_k}$.*

Proof: Define $NP'(2k) =_{\mathrm{df}} NP(2k)$ and $NP'(2k+1) =_{\mathrm{df}} co\text{-}NP(2k+1)$ for $k \geq 0$. We observe that $NP'(k+1) = co\text{-}NP'(k) \wedge co\text{-}NP$. Hence $\mathbf{SAT}_k$ is $\leq^{\mathrm{p}}_m$-complete for $NP'(k+1)$ for $k \geq 0$, and $A_k \in co\text{-}NP'(k) \wedge co\text{-}\hat{\mathcal{R}}^p_m(A)$ for $k \geq 1$. Using these facts and $A \in low^p_{k+1}$ we conclude $A_k \in co\text{-}NP'(k+1) \wedge co\text{-}\hat{\mathcal{R}}^p_m(A) = co\text{-}NP'(k+1) = co\text{-}\mathcal{R}^p_m(\mathbf{SAT}_k) = \mathcal{R}^p_m(\overline{\mathbf{SAT}_k})$. ❏

Similar to [CK96] we define the notion of hard sequences. Note that a hard sequence wrt. h for length m there corresponds to a hard sequence wrt. $(\mathbf{SAT}, h, m)$ here.

Definition 15

- *Let $k \geq 1$, $m \geq 1$, $j = 1, \ldots, k-1$, $A \subseteq \Sigma^*$, and $h : (\Sigma^*)^k \to (\Sigma^*)^k$. We call $\langle x_1, \ldots, x_j \rangle$ a* hard sequence *wrt. (A, h, m) iff $j = 0$ or*

  1. $1 \leq j \leq k-1$,
  2. $|x_j| \leq m$,
  3. $x_j \in \overline{A}$ *if* $j = 1$ *and* $x_j \in \overline{\mathbf{SAT}}$ *if* $j > 1$,
  4. $\forall y_1, \ldots, y_{k-j} \in \Sigma^{\leq m}(h(y_1, \ldots, y_{k-j}, x_j, \ldots, x_1)_{(k-j+1)} \in \overline{\mathbf{SAT}})$,
  5. $\langle x_1, \ldots, x_{j-1} \rangle$ *is a hard sequence wrt.* (A, h, m).

- *We call* j order *of the hard sequence* $\langle x_1, \ldots, x_j \rangle$.

- *If there is no hard sequence wrt. (A, h, m) of order greater than j then every hard sequence wrt. (A, h, m) of order j is called a* maximal hard sequence *wrt. (A, h, m).*

Now let $k \geq 1$ and $A \in low_k^p$. By Lemma 14 we get $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$. The next lemma shows, that under this assumption one can reduce also A_{k-j-1} to $\overline{\mathbf{SAT}_{k-j-1}}$ by using a hard sequence of order j.

Lemma 16 *Let $A \in NP \backslash \{\emptyset, \Sigma^*\}$, $k \geq 2$, $m \geq 1$, $j = 1, \ldots, k-1$, and $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$ via $h \in FP$. If $\langle x_1, \ldots, x_j \rangle$ is a hard sequence wrt. (A, h, m) then for all $y_1, \ldots, y_{k-j} \in \Sigma^{\leq m}$:*

$$\langle y_1, \ldots, y_{k-j} \rangle \in \mathbf{SAT}_{k-j-1} \Leftrightarrow h(y_1, \ldots, y_{k-j}, x_j, \ldots, x_1)_{(1,k-j)} \in \overline{\mathbf{SAT}_{k-j-1}}.$$

Proof: By induction on j. For $j = 1$ let $\langle x_1 \rangle$ be a hard sequence wrt. (A, h, m). Hence $x_1 \in \overline{A}$ and $h(y_1, \ldots, y_{k-1}, x_1)_{(k)} \in \overline{\mathbf{SAT}}$ for all $y_1, \ldots, y_{k-1} \in \Sigma^{\leq m}$. We conclude

$$\begin{aligned}
\langle y_1, \ldots, y_{k-1} \rangle \in \overline{\mathbf{SAT}_{k-2}} &\Leftrightarrow \langle y_1, \ldots, y_{k-1} \rangle \in \overline{\mathbf{SAT}_{k-2}} \wedge x_1 \in \overline{A} \\
&\Leftrightarrow \langle y_1, \ldots, y_{k-1}, x_1 \rangle \in A_{k-1} \\
&\Leftrightarrow h(y_1, \ldots, y_{k-1}, x_1) \in \overline{\mathbf{SAT}_{k-1}} \\
&\Leftrightarrow h(y_1, \ldots, y_{k-1}, x_1)_{(1,k-1)} \in \mathbf{SAT}_{k-2} \\
&\qquad \vee h(y_1, \ldots, y_{k-1}, x_1)_{(k)} \in \mathbf{SAT} \\
&\Leftrightarrow h(y_1, \ldots, y_{k-1}, x_1)_{(1,k-1)} \in \mathbf{SAT}_{k-2}
\end{aligned}$$

For $j \geq 1$ let $\langle x_1, \ldots, x_{j+1} \rangle$ be a hard sequence wrt. (A, h, m). Hence we have $x_{j+1} \in \overline{\mathbf{SAT}}$ and $h(y_1, \ldots, y_{k-j-1}, x_{j+1}, \ldots, x_1)_{(k-j)} \in \overline{\mathbf{SAT}}$ for all $y_1, \ldots, y_{k-j-1} \in \Sigma^{\leq m}$. We conclude

$$\begin{aligned}
\langle y_1, \ldots, y_{k-j-1} \rangle \in \overline{\mathbf{SAT}_{k-j-2}} &\Leftrightarrow \langle y_1, \ldots, y_{k-j-1} \rangle \in \overline{\mathbf{SAT}_{k-j-2}} \wedge x_{j+1} \in \overline{\mathbf{SAT}} \\
&\Leftrightarrow \langle y_1, \ldots, y_{k-j-1}, x_{j+1} \rangle \in \mathbf{SAT}_{k-j-1} \\
&\Leftrightarrow h(y_1, \ldots, y_{k-j-1}, x_{j+1}, \ldots, x_1)_{(1,k-j)} \in \overline{\mathbf{SAT}_{k-j-1}} \\
&\qquad \text{(induction hypothesis)} \\
&\Leftrightarrow h(y_1, \ldots, y_{k-j-1}, x_{j+1}, \ldots, x_1)_{(1,k-j-1)} \in \mathbf{SAT}_{k-j-2} \\
&\qquad \vee h(y_1, \ldots, y_{k-j-1}, x_{j+1}, \ldots, x_1)_{(k-j)} \in \mathbf{SAT} \\
&\Leftrightarrow h(y_1, \ldots, y_{k-j-1}, x_{j+1}, \ldots, x_1)_{(1,k-j-1)} \in \mathbf{SAT}_{k-j-2}
\end{aligned}$$

❑

The next lemma shows, that we can use maximal hard sequences to reduce $\overline{A}$ to $\mathbf{SAT}$. In this case the hard sequence is used as an *advice* by the reduction function. (See [KL80] and [Wag87] for more information about advices.)

Lemma 17 *Let $A \in NP\backslash\{\emptyset, \Sigma^*\}$, $k \geq 1$, and $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$ via $h \in FP$. There exist a set $B \in NP$ and a polynomial r such that for every n: If $\langle x_1, \ldots, x_j\rangle$ is a maximal hard sequence wrt. $(A, h, r(n))$ then for all $y \in \Sigma^{\leq n}$:*

$$y \in \overline{A} \Leftrightarrow \langle y, 1^n, x_1, \ldots, x_j\rangle \in B$$

Proof: Let $A \leq_m^{\mathrm{p}} \mathbf{SAT}$ via $g \in FP$ and choose a polynomial r such that $|g(x)| \leq r(|x|)$ for all $x \in \Sigma^*$. Define the NP-set

$$\begin{aligned} B =_{\mathrm{df}} \{\langle y, 1^n, x_1, \ldots, x_j\rangle | 1 \leq j \leq k-1 \wedge \exists y_1, \ldots, y_{k-j-1} \in \Sigma^{\leq r(n)} \\ (h(y_1, \ldots, y_{k-j-1}, g(y), x_j, \ldots, x_1)_{(k-j)} \in \mathbf{SAT})\} \\ \cup \{(y, 1^n) | \exists y_1, \ldots, y_{k-1} \in \Sigma^{\leq r(n)} (h(y_1, \ldots, y_{k-1}, y)_{(k)} \in \mathbf{SAT}\} \end{aligned}$$

We consider three cases.

Case $j = 0$. For every $y \in \Sigma^{\leq r(n)} \supseteq \Sigma^{\leq n}$ the sequence $\langle y\rangle$ is not hard wrt. $(A, h, r(n))$. Hence $y \in A$ or $\exists y_1, \ldots, y_{k-1} \in \Sigma^{\leq r(n)} (h(y_1, \ldots, y_{k-1}, y)_{(k)} \in \mathbf{SAT}$ and consequently $y \in \overline{A} \Rightarrow (y, 1^n) \in B$. On the other hand, the reduction $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$ via h implies for all $y_1, \ldots, y_{k-1}, y \in \Sigma^{\leq r(n)}$:

$$\begin{aligned} \langle y_1, \ldots, y_{k-1}\rangle \in \overline{\mathbf{SAT}_{k-2}} \wedge y \in \overline{A} \Leftrightarrow \\ \Leftrightarrow h(y_1, \ldots, y_{k-1}, y)_{(1,k-1)} \in \mathbf{SAT}_{k-2} \vee h(y_1, \ldots, y_{k-1}, y)_{(k)} \in \mathbf{SAT}. \end{aligned}$$

If $(y, 1^n) \in B$ then there exist $y_1, \ldots, y_{k-1} \in \Sigma^{\leq r(n)}$ such that $h(y_1, \ldots, y_{k-1}, y)_{(k)} \in \mathbf{SAT}$. Consequently, $y \in \overline{A}$.

Case $1 \leq j = k-1$. By Lemma 16 we obtain for all $y \in \Sigma^{\leq n}$:

$$y \in \overline{A} \Leftrightarrow g(y) \in \overline{\mathbf{SAT}} \Leftrightarrow h(g(y), x_{k-1}, \ldots, x_1) \in \mathbf{SAT} \Leftrightarrow \langle y, 1^n, x_1, \ldots, x_{k-1}\rangle \in B$$

Case $1 \leq j < k-1$. This case is similar to the first case. Since $\langle x_1, \ldots, x_j\rangle$ is maximal, the sequence $\langle x_1, \ldots, x_j, g(y)\rangle$ cannot be hard wrt. $(A, h, r(n))$ for all $y \in \Sigma^{\leq n}$. Hence $g(y) \in \mathbf{SAT}$ or $\exists y_1, \ldots, y_{k-j-1} \in \Sigma^{\leq r(n)} (h(y_1, \ldots, y_{k-j-1}, g(y), x_j, \ldots, x_1)_{(k-j)} \in \mathbf{SAT})$. Consequently, $y \in \overline{A} \Rightarrow g(y) \in \overline{\mathbf{SAT}} \Rightarrow \langle y, 1^n, x_1, \ldots, x_j\rangle \in B$. On the other hand, Lemma 16 implies for all $y_1, \ldots, y_{k-j-1} \in \Sigma^{\leq r(n)}$ and $y \in \Sigma^{\leq n}$:

$$\begin{aligned} \langle y_1, \ldots, y_{k-j-1}\rangle \in \overline{\mathbf{SAT}_{k-j-2}} \wedge g(y) \in \overline{\mathbf{SAT}} \Leftrightarrow \\ \Leftrightarrow h(y_1, \ldots, y_{k-j-1}, g(y), x_j, \ldots, x_1)_{(1,k-j-1)} \in \mathbf{SAT}_{k-j-2} \vee \\ \vee h(y_1, \ldots, y_{k-j-1}, g(y), x_j, \ldots, x_1)_{(k-j)} \in \mathbf{SAT}. \end{aligned}$$

Now, if $\langle y, 1^n, x_j, \ldots, x_1\rangle \in B$ then there exist $y_1, \ldots, y_{k-j-1} \in \Sigma^{\leq r(n)}$ such that $h(y_1, \ldots, y_{k-j-1}, g(y), x_j, \ldots, x_1)_{(k-j)} \in \mathbf{SAT}$. Consequently, $g(y) \in \overline{\mathbf{SAT}}$ and hence $y \in \overline{A}$. ❑

Lemma 18 *Let $A \in NP$, $k \geq 1$, and $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$ via $h \in FP$. For every set $L \in NP^A$ there exists a set $C \in NP$ and a polynomial s such that for every n: If $\langle x_1, \ldots, x_j\rangle$ is a maximal hard sequence wrt. $(A, h, s(n))$ then for all $w \in \Sigma^{\leq n}$:*

$$w \in L \Leftrightarrow \langle w, 1^n, x_1, \ldots, x_j\rangle \in C$$

Proof: For $A = \emptyset$ and $\overline{A} = \emptyset$ this lemma is obviously valid. Now let $A \neq \emptyset$ and $\overline{A} \neq \emptyset$, and let $B \in NP$ and the polynomial r as given by Lemma 17. Let M be an NP machine accepting L with oracle A, and let s be a polynomial bounding the running time of M. Let M_1 and M_2 be NP machines accepting A and B, resp. Let $\langle x_1, \ldots, x_j \rangle$ be a maximal hard sequence wrt. $(A, h, s(n))$. Now define an NP machine M' (without oracle) which works on input $\langle w, 1^n, x_1, \ldots, x_j \rangle$ for $w \in \Sigma^{\leq n}$ as follows: M' simulates M as long as no oracle query appears. If M on path p queries z to oracle A then M' splits path p into two paths. On one of these paths M' simulates the work of M_1 on input z. Accepting paths of M_1 (witnessing $z \in A$) are continued by M' by simulating path p of M after an affirmative answer of A to query z. On the other path M' simulates the work of M_2 on input $\langle z, 1^{s(n)}, x_1, \ldots, x_j \rangle$. Accepting paths of M_2 (witnessing $z \in \overline{A}$) are continued by M' by simulating path p of M after a negative answer of A to query z. The machine M' stops on all non accepting paths of M_1 and M_2 . It is obvious that exactly those paths are kept alive by M' which simulate a path of M correctly (i.e. with the correct answers of A to all queries). Consequently, M' accepts $\langle w, 1^n, x_1, \ldots, x_j \rangle$ if and only if w is accepted by M. We define C to be the NP-set accepted by M'. ❏

Lemma 19 *Let $A \in NP$, $k \geq 1$, and $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$ via $h \in FP$. For every set $L \in (\Sigma_2^p)^A$ there exists a set $D \in \Sigma_2^p$ and a polynomial t such that for every n: If $j \in \{0, 1, \ldots, k-1\}$ is the order of a maximal hard sequence wrt. $(A, h, t(n))$ then for all $z \in \Sigma^{\leq n}$:*

$$z \in L \Leftrightarrow \langle z, 1^n, j \rangle \in D$$

If j is greater than the order of a maximal hard sequence wrt. $(A, h, t(n))$ then $\langle z, 1^n, j \rangle \notin D$.

Proof: Let M be an NP oracle machine accepting L with an oracle $L' \in NP^A$, and let u be a polynomial bounding the running time of M. Apply Lemma 18 to L' and get $C \in NP$ and a polynomial s such that for all n: If $\langle x_1, \ldots, x_j \rangle$ is a maximal hard sequence wrt. $(A, h, s(u(n)))$ then

$$w \in L' \Leftrightarrow \langle w, 1^{u(n)}, x_1, \ldots, x_j \rangle \in C$$

for all $w \in \Sigma^{\leq u(n)}$. Define $t(n) =_{\mathrm{df}} s(u(n))$. Let E be the set of all sequences $\langle 1^m, x_1, \ldots, x_j \rangle$ which are not hard wrt. (A, h, m). By the definition of hard sequences, E is in NP. We define F to be the disjoint union of C and E. Now define an NP machine M' which works with oracle F on input $\langle z, 1^n, j \rangle$ for $z \in \Sigma^{\leq n}$ as follows: M' guesses a sequence $\langle x_1, \ldots, x_j \rangle$ such that $|x_i| \leq t(n)$ for $i = 1, \ldots, j$ and then it queries the oracle E whether $\langle x_1, \ldots, x_j \rangle$ is hard wrt. $(A, h, t(n))$. If the answer is "yes" then M' rejects on this path. Otherwise M' simulates the work of M on input z with oracle L'. Whenever M queries w to oracle L' then M' queries (equivalently) $\langle w, 1^{u(n)}, x_1, \ldots, x_j \rangle$ to the oracle C. Obviously, M' accepts $\langle z, 1^n, j \rangle$ if and only if M accepts z. We define D to be the Σ_2^p set accepted by M' with oracle $F \in NP$. ❏

Lemma 20 *Let $k \geq 1$ and let $A \in low_k^p$.*

1. $(\Sigma_2^p)^A \subseteq \Sigma_2^p(2k-1)$
2. $(\Delta_2^p)^A \subseteq \Sigma_2^p(k-1) \oplus \Delta_2^p$

Proof: Let $A \in low_k^p$. By Lemma 14 we get $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$ via a suitable function $h \in FP$.

For $L \in (\Sigma_2^p)^A$ we obtain by Lemma 19 a set $D \in \Sigma_2^p$ and a polynomial t such that for every n: If $j \in \{0, 1, \ldots, k-1\}$ is the order of a maximal hard sequence wrt. $(A, h, t(n))$ then $z \in L \Leftrightarrow$

$\langle z, 1^n, j\rangle \in D$ for all $z \in \Sigma^{\leq n}$ and $(z, 1^n, j) \notin D$ if j is greater than the order of a maximal hard sequence wrt. $(A, h, t(n))$. We define the Σ_2^p set

$$T =_{\mathrm{df}} \{(1^m, j) | \exists x_1, \ldots, x_j \in \Sigma^{\leq m} \text{ and } \langle 1^m, x_1, \ldots, x_j\rangle \text{ is a hard sequence wrt. } (A, h, t(n))\},$$

and we obtain:

$$\begin{aligned} z \in L \quad \Leftrightarrow \quad & (((z, 1^{|z|}, 0) \in D) \vee ((1^{t(|z|)}, 1) \in T)) \oplus \\ & ((1^{t(|z|)}, 1) \in T) \oplus (((z, 1^{|z|}, 1) \in D) \vee ((1^{t(|z|)}, 2) \in T)) \oplus \\ & \quad \vdots \\ & ((1^{t(|z|)}, k-2) \in T) \oplus (((z, 1^{|z|}, k-2) \in D) \vee ((1^{t(|z|)}, k-1) \in T)) \oplus \\ & ((1^{t(|z|)}, k-1) \in T) \oplus ((z, 1^{|z|}, k-1) \in D) \end{aligned}$$

Consequently, $L \in \underbrace{\Sigma_2^p \oplus \Sigma_2^p \oplus \ldots \oplus \Sigma_2^p}_{2k-1} = \Sigma_2^p(2k-1)$ (Lemma 1.4).

For $L \in P^{NP^A}$ let M be a deterministic polynomial time machine accepting L with oracle $L' \in NP^A$ and running time bounded by a polynomial t. By Lemma 14 we have $A_{k-1} \leq_m^{\mathrm{p}} \overline{\mathbf{SAT}_{k-1}}$ via a suitable function $h \in FP$. By Lemma 18 there exists a set $C \in NP$ and a polynomial s such that for every n: If $\langle x_1, \ldots, x_j\rangle$ is a maximal hard sequence wrt. $(A, h, s(n))$ then $w \in L' \Leftrightarrow \langle w, 1^n, x_1, \ldots, x_j\rangle \in C$ for all $w \in \Sigma^{\leq n}$. Hence there a set $E \in P^{NP}$ such that for every n: If $\langle x_1, \ldots, x_j\rangle$ is a maximal hard sequence wrt. $(A, h, s(t(n)))$ then $z \in L \Leftrightarrow \langle z, 1^{t(n)}, x_1, \ldots, x_j\rangle \in E$ for all $z \in \Sigma^{\leq n}$. To use the *mind-change technique* we consider on the set $(\Sigma^*)^{<k}$ a partial order $\sqsubseteq$ defined by $\langle x_1, \ldots, x_i\rangle \sqsubseteq \langle y_1, \ldots, y_j\rangle$ iff ($i \leq j$ and $x_l = y_l$ for $l = 1, \ldots, i$). Define

$$\begin{aligned} F =_{\mathrm{df}} \{(z, 1^n, j) | z \in \Sigma^{\leq n} \wedge \exists \overline{x_1} \ldots \exists \overline{x_j} (\overline{x_1} \sqsubseteq \ldots \sqsubseteq \overline{x_j} \wedge \overline{x_j} \text{ is hard wrt. } (A, h, s(t(n))) \wedge \\ \wedge c_E(z, 1^{t(n)}) \neq c_E(z, 1^{t(n)}, \overline{x_1}) \neq \ldots \neq c_E(z, 1^{t(n)}, \overline{x_j}))\}. \end{aligned}$$

Obviously $F \in \Sigma_2^p$, and we obtain

$$z \in L \Leftrightarrow (z, 1^{t(n)}) \in E \oplus (z, 1^{t(n)}, 1) \in F \oplus (z, 1^{t(n)}, 2) \in F \oplus \cdots \oplus (z, 1^{t(n)}, k-1) \in F$$

for all $z \in \Sigma^{\leq n}$. Hence $L \in \Delta_2^p \oplus \underbrace{\Sigma_2^p \oplus \Sigma_2^p \oplus \ldots \oplus \Sigma_2^p}_{k-1} = \Sigma_2^p(k-1) \oplus \Delta_2^p$ (Lemma 1.4). ❏

Now we are able to relate boolean lowness to classical lowness.

Theorem 21 *For $k \geq 0$, if $A \in low_k^p$ then $BH(\Sigma_2^p)^A = BH(\Sigma_2^p)$, i.e. $low_k^p \subseteq Low(BH(\Sigma_2^p))$.*

Proof: By Lemma 20 and Lemma 1.5 we conclude $(\Sigma_2^p(m))^A = (\Sigma_2^p)^A(m) \subseteq (\Sigma_2^p(2k-1))(m) \subseteq P_{\|}^{\Sigma_2^p}[(2k-1) \cdot m] \subseteq \Sigma_2^p(2km - m + 1)$. ❏

Corollary 22 $low_0^p = Low_0^p \subseteq low_1^p = Low_1^p \subseteq low_2^p \subseteq low_3^p \subseteq \ldots \subseteq Low(BH(\Sigma_2^p)) \subseteq Low_3^p$.

5 Consequences to collapse results

From Lemma 20.1 we get immediately a statement on the connection between the collapses of the boolean hierarchy and the polynomial-time hierarchy.

Corollary 23 *For $k \geq 1$, if $BH = NP(k)$ then $PH = \Sigma_2^p(2k-1)$.*

Proof: From $BH = NP(k)$ we get $\mathbf{SAT} \in low_k^p$ by Theorem 10 and $\Sigma_3^p = (\Sigma_2^p)^{\mathbf{SAT}} \subseteq \Sigma_2^p(2k-1)$ by Lemma 20.1. Hence $PH = \Sigma_2^p(2k-1)$. ❑

This improves Kadin's original result from [Kad88] but it is not as good as the result $BH = NP(k) \Rightarrow PH = \Sigma_2^p(k)$ from [CK96] or the further improvement in [BCO93]. It is obvious that an improvement of Lemma 20.1 by replacing $\Sigma_2^p(2k-1)$ by a smaller class $\mathcal{K}$ yields the improvement $BH = NP(k) \Rightarrow PH = \mathcal{K}$ of Corollary 23. However, to follow the idea from [BCO93] to improve Lemma 20.1 we have to combine Lemma 20.1 and Lemma 20.2. But this cannot be done without the additional assumption that A is not "too easy".

Lemma 24 *For $k \geq 1$, if $A \in low_k^p \cap High_1^p$ then $(\Sigma_2^p)^A \subseteq \Sigma_2^p(k-1) \oplus \Delta_2^p$.*

Proof: By Lemma 20 we obtain $(\Sigma_2^p)^A \subseteq \Sigma_2^p(2k-1)$ and $(\Delta_2^p)^A \subseteq \Sigma_2^p(k-1) \oplus \Delta_2^p$. Since $A \in High_1^p$ we can conclude $(\Sigma_2^p)^A \subseteq P^{\Sigma_2^p} = P^{NP^A} \subseteq \Sigma_2^p(k-1) \oplus \Delta_2^p$. ❑

Theorem 25 *For $k \geq 1$, if $low_k^p \cap High_1^p \neq \emptyset$ then $PH = \Sigma_2^p(k-1) \oplus \Delta_2^p$.*

Proof: For $A \in low_k^p \cap High_1^p$ we obtain $(\Sigma_2^p)^A \subseteq \Sigma_2^p(k-1) \oplus \Delta_2^p$ by Lemma 24 and $(\Sigma_2^p)^A = \Sigma_3^p$ because of $High_1^p \subseteq High_2^p$. Hence $\Sigma_3^p = \Sigma_2^p(k-1) \oplus \Delta_2^p$ and consequently $PH = \Sigma_2^p(k-1) \oplus \Delta_2^p$. ❑

Now we get as a corollary the best known result on the connection between the collapses of the boolean hierarchy and the polynomial-time hierarchy. This can be found implicitly in [BCO93].

Corollary 26 *For $k \geq 1$, if $BH = NP(k)$ then $PH = \Sigma_2^p(k-1) \oplus NP(k)$.*

Proof: From $BH = NP(k)$ we get $\mathbf{SAT} \in low_k^p$ by Theorem 10. Since $\mathbf{SAT} \in High_1^p$ we obtain $PH = \Sigma_2^p(k-1) \oplus \Delta_2^p$ by Theorem 25. However, it is known from [Cha97] that $BH = NP(k)$ implies $\Delta_2^p = NP(k)$. ❑

Since all the above results are relativizable we have also[1]

Corollary 27 *For $m, k \geq 1$, if $BH(\Sigma_m^p) = \Sigma_m^p(k)$ then $PH = \Sigma_{m+1}^p(k-1) \oplus \Sigma_m^p(k)$.*

References

[BCO93] R. BEIGEL, R. CHANG, AND M. OGIHARA. A relationship between difference hierarchies and relativized polynomial hierarchies. *Mathematical Systems Theory*, 26:293 – 310, 1993.

[CGH+88] J.-Y. CAI, T. GUNDERMANN, J. HARTMANIS, L. A. HEMACHANDRA, V. SEWELSON, K. WAGNER, AND G. WECHSUNG. The boolean hierarchy I: Structural properties. *SIAM Journal on Computing*, 17(6):1232 – 1252, 1988.

[CH86] J. CAI AND L. HAMACHANDRA. The boolean hierarchy: Hardware over NP. *Proceedings of the Structure in Complexity Theory Conference, Lecture Notes in Computer Science 223*, pages 105 – 124, 1986.

[Cha97] R. CHANG. Bounded queries, approximation and the boolean hierarchy. Technical report, TR CS 97-04, University of Maryland, Department of Computer Science and Electrical Engineering, 1997.

[CK96] R. CHANG AND J. KADIN. The boolean hierarchy and the polynomial hierarchy: A closer connection. *SIAM Journal on Computing*, 25(2):340, 1996.

[Coo74] S. B. COOPER. Minimal pairs and high recursively enumerable degree. *Journal of Symbolic Logic*, 39:655 – 660, 1974.

[1] After completion of this work the authors were informed by Lane A. Hemaspaandra about the fact that this result was independently obtained by Hemaspaandra, Hemaspaandra, and Hempel [HHH].

[HHH] E. HEMASPAANDRA, L. A. HEMASPAANDRA, AND H. HEMPEL. Unpublished manuscript.

[Kad88] J. KADIN. The polynomial time hierarchy collapses if the boolean hierarchy collapses. *SIAM Journal on Computing*, 17(6):1263 – 1282, 1988.

[KL80] R. KARP AND R. LIPTON. Some connections between nonuniform and uniform complexity classes. In *Proc. 12th ACM STOC*, pages 302 – 309. ACM, 1980.

[Köb85] J. KÖBLER. Untersuchungen verschiedener polynomieller Reduktionsklassen von NP. Diploma Thesis, Universität Stuttgart, 1985.

[KSW87] J. KÖBLER, U. SCHÖNING, AND K. W. WAGNER. The difference and truth-table hierarchies for NP. *Theoretical Informatics and Applications*, 21(4):419 – 435, 1987.

[Sch83] U. SCHÖNING. A low and a high hierarchy within NP. *Journal of Computer and System Sciences*, 27:14 – 28, 1983.

[Sch85] U. SCHÖNING. *Complexity and Structure*, volume 211 of *Lecture Notes in Computer Science*. Springer-Verlag, 1985.

[Soa74] R. I. SOARE. Automorphisms of the lattice of recursively enumerable sets. *Bulletin of the American Mathematical Society*, 80:53 – 58, 1974.

[Wag87] K. W. WAGNER. Number-of-query hierachies. Technical Report 158, University of Augsburg, 1987.

[Wag97] K. W. WAGNER. A note on bounded queries and the difference hierarchy. Technical report, Technical Report 173, Institut für Informatik, Universität Würzburg, 1997.

[WW85] G. WECHSUNG AND K. WAGNER. On the boolean closure of NP, manuscript. Extended abstract as: Wechsung, G., On the boolean closure of NP. In *Proc. of the Conference of Foundations on Computation Theory*, volume 199 of *Lecture Notes in Computer Science*, pages 485–493. Springer Verlag, 1985.