

# On the Power of Unambiguity in Alternating Machines

Holger Spakowski<sup>1\*\*\*</sup> and Rahul Tripathi<sup>†2</sup>

<sup>1</sup> Institut für Informatik, Heinrich-Heine-Universität, 40225 Düsseldorf, Germany.  
spakowsk@cs.uni-duesseldorf.de.

<sup>2</sup> Dept. of Computer Science and Engineering, University of South Florida, Tampa, FL 33620, USA. rahul.tripathi007@gmail.com.

**Abstract.** Recently, the property of unambiguity in alternating Turing machines has received considerable attention in the context of analyzing globally-unique games by Aida et al. [1] and in the design of efficient protocols involving globally-unique games by Crăsmaru et al. [7]. This paper investigates the power of unambiguity in alternating Turing machines in the following settings:

1. We construct a relativized world where unambiguity based hierarchies—AUPH, UPH, and  $\mathcal{UPH}$ —are infinite. We construct another relativized world where UAP (unambiguous alternating polynomial-time) is not contained in the polynomial hierarchy.
2. We define the bounded-level unambiguous alternating solution class  $\text{UAS}(k)$ , for every  $k \geq 1$ , as the class of sets for which strings in the set are accepted unambiguously by some polynomial-time alternating Turing machine  $N$  with at most  $k$  alternations, while strings not in the set either are rejected or are accepted with ambiguity by  $N$ . We construct a relativized world where, for all  $k \geq 1$ ,  $\text{UP}_{\leq k} \subset \text{UP}_{\leq k+1}$  and  $\text{UAS}(k) \subset \text{UAS}(k+1)$ .
3. Finally, we show that robustly  $k$ -level unambiguous polynomial-time alternating Turing machines accept languages that are computable in  $\text{P}^{\Sigma_k^p \oplus \mathcal{A}}$ , for every oracle  $\mathcal{A}$ . This generalizes a result of Hartmanis and Hemachandra [11].

## 1 Introduction

Chandra, Kozen, and Stockmeyer [6] introduced the notion of *alternation* as a generalization of nondeterminism: Alternation allows switching of existential and universal quantifiers, whereas nondeterminism allows only existential quantifiers throughout the computation. Alternation has proved to be a central notion in complexity theory. For instance, the polynomial hierarchy has a characterization in terms of bounded-level alternation [6, 23], the complexity class

\*\*\* Supported in part by the DFG under grant RO 1202/9-1.

† Supported in part by NSF grant CCF-0426761. Work done in part while affiliated with the Department of Computer Science at the University of Rochester, Rochester, NY 14627, USA.

PSPACE can be characterized in terms of polynomial length-bounded alternation [6], and many important classes have characterizations based on variants of alternation (see Chapter 19 of [20]).

*Unambiguity* in nondeterministic computation is related to issues such as worst-case cryptography and the closure properties of  $\#P$  (the class of functions that count the number of accepting paths of NP machines). The complexity class UP captures the notion of unambiguity in nondeterministic polynomial-time Turing machines. It is known that one-to-one one-way functions exist if and only if  $P \neq UP$  [10, 14] and that UP equals probabilistic polynomial-time if and only if  $\#P$  is closed under every polynomial-time computable operation [19]. Factoring, a natural problem with cryptographic applications, belongs to  $UP \cap \text{coUP}$  and is not known to belong to a subclass of  $UP \cap \text{coUP}$  nontrivially.

This paper studies the power of unambiguity in alternating computations. Niedermeier and Rossmanith [18] gave the following definition of unambiguity in alternating Turing machines: An alternating Turing machine is *unambiguous* if every accepting existential configuration has exactly one move to an accepting configuration and every rejecting universal configuration has exactly one move to a rejecting configuration. They introduced a natural analog UAP (unambiguous alternating polynomial-time) of UP for alternating Turing machines. Lange and Rossmanith [17] proposed three different approaches to define a hierarchy for unambiguous computations: The alternating unambiguous polynomial hierarchy AUPH, the unambiguous polynomial hierarchy UPH, and the promise unambiguous hierarchy *UPH*. Though it is known that  $\text{Few} \subseteq \text{UAP} \subseteq \text{SPP}$  [18] and  $\text{AUPH} \subseteq \text{UPH} \subseteq \text{UPH} \subseteq \text{UAP}$  [7, 17], a number of questions—such as, whether UAP is contained in the polynomial hierarchy, whether the unambiguity based hierarchies intertwine, whether these hierarchies are infinite, or whether some hierarchy is contained in a fixed level of the other hierarchy—related to these hierarchies have remained open [17]. Relatedly, Hemaspaandra and Rothe [13] showed that the existence of a sparse Turing-complete set for UP has consequences on the structure of unambiguity based hierarchies.

Recently, Aida et al. [1] introduced “uniqueness” properties for two-player games of perfect information such as Checker, Chess, and Go. A two-person perfect information game has *global uniqueness* property if every winning position of player 1 has a unique move to win and every mis-step by player 1 is punishable by a unique winning reply by player 2 throughout the course of the game. Aida et al. [1] showed that the class of languages that reduce to globally-unique games, i.e., games with global uniqueness property, is the same as the class UAP. In another recent paper, Crăsmaru et al. [7] designed a protocol by which a series of globally-unique games can be combined into a single globally-unique game, even under the condition that the result of the new game is a non-monotone function of the results of the individual games that are unknown to the players. In complexity theoretic terms, they showed that the class UAP is self-low, i.e.,  $\text{UAP}^{\text{UAP}} = \text{UAP}$ . They also observed that the graph isomorphism problem, whose membership in SPP was shown by Arvind and Kurur [2], in fact belongs to the subclass UAP of SPP.

In this paper, we investigate the power of unambiguity based alternating computation in three different settings. First, we construct a relativized world in which the unambiguity based hierarchies—AUPH, UPH, and  $\mathcal{UPH}$ —are infinite. We construct another relativized world where UAP is not contained in the polynomial hierarchy. This latter oracle result strengthens a result (relative to an oracle, UAP differs from the second level of  $\mathcal{UPH}$ ) of Crăsmaru et al. [7]. Our results show that proving that any of the unambiguity based hierarchies is finite or that UAP is contained in the polynomial hierarchy is impossible by relativizable proof techniques. We mention that the structure of relativized hierarchies of classes has been investigated extensively in complexity theory (see, for instance [5, 12, 15, 16, 25]) and our investigation is a work in this direction.

Second, for every  $k \geq 1$ , we define a complexity class  $\text{UAS}(k)$  as the class of sets for which every string in the set is accepted unambiguously by some polynomial-time alternating Turing machine  $N$  with at most  $k$  alternations, while strings not in the set either are rejected or are accepted with ambiguity by  $N$ . A variant of this class (denoted by UAS in this paper), where the number of alternations is allowed to be unbounded, was studied by Wagner [24] as the class  $\nabla\text{P}$  of all sets which can be accepted by polynomial-time alternating Turing machines using partially defined AND and OR functions.<sup>3</sup> Beigel [3] defined the class  $\text{UP}_{\leq k(n)}$  as the class of sets in NP that are accepted by nondeterministic polynomial-time Turing machines with at most  $k(n)$  accepting paths on each input of length  $n$ . Beigel [3] constructed an oracle  $\mathcal{A}$  such that  $\text{P}^{\mathcal{A}} \subset \text{UP}^{\mathcal{A}} \subset \text{UP}_{\leq k(n)}^{\mathcal{A}} \subset \text{UP}_{\leq k(n)+1}^{\mathcal{A}} \subset \text{FewP}^{\mathcal{A}} \subset \text{NP}^{\mathcal{A}}$ , for every polynomial  $k(n) \geq 2$ . We show that there is a relativized world  $\mathcal{B}$  such that, for all  $k \geq 1$ ,  $\text{UP}_{\leq k}^{\mathcal{B}} \subset \text{UP}_{\leq k+1}^{\mathcal{B}}$ ,  $\text{UAS}(k)^{\mathcal{B}} \subset \text{UAS}(k+1)^{\mathcal{B}}$ , and relative to  $\mathcal{B}$ , the second level of  $\mathcal{UPH}$  is not contained in any level of AUPH.

Finally, we investigate the power of alternating Turing machines that preserve the bounded-level unambiguity property for every oracle. We show that a polynomial-time alternating Turing machine that preserves  $k$ -level alternation unambiguously in every relativized world requires only weak oracle access in every relativized world, i.e., for every oracle  $\mathcal{A}$ , the language of such a machine can be computed in  $\text{P}^{\Sigma_k^p \oplus \mathcal{A}}$ . This is a generalization of a result of Hartmanis and Hemachandra [11], which states that if a nondeterministic polynomial-time Turing machine is robustly categorical (i.e., for no oracle and for no input, the machine has more than one accepting path), then for every oracle  $\mathcal{A}$ , the machine accepts a language in  $\text{P}^{\text{NP} \oplus \mathcal{A}}$ .

The paper is organized as follows. Section 2 describes the notations and the definitions that are relevant to this paper. In Section 3, we describe our results on relativized separations of unambiguity based hierarchies and relativized non-inclusion of UAP in the polynomial hierarchy. In Section 4, for every  $k \geq 1$ , we define a complexity class  $\text{UAS}(k)$  and study its relativized complexity w.r.t. the

<sup>3</sup> The partial counterparts AND\* and OR\* differ from boolean functions AND and OR, respectively, as follows: AND\* is undefined for input (0, 0) and OR\* is undefined for input (1, 1). Thus, these partially defined boolean functions are the unambiguous counterparts of boolean AND and OR functions, respectively.

bounded-ambiguity class  $UP_{\leq k+1}$ . Finally, Section 5 includes our results on the power of robustly bounded-level unambiguous polynomial-time alternating Turing machines. (Proofs omitted due to space limitations can be found in the detailed version available at <http://www.cs.rochester.edu/trs/theory-trs.html>.)

## 2 Preliminaries

Let  $\mathbb{N}^+$  denote the set of positive integers. We assume that the root of a computation tree of every alternating Turing machine (or, ATM in short) is an existential node. We recursively assign levels in a computation tree  $T$  of an ATM as follows: (a) the root of  $T$  is at level 1, (b) if a node  $v$  is assigned a level  $i$  and if  $v$  is an existential node, then the first nonexistential (i.e., universal or leaf) node  $w$  reachable along some path from  $v$  to a leaf node of  $T$  is assigned level  $i + 1$ , (c) if a node  $v$  is assigned a level  $i$  and if  $v$  is a universal node, then the first nonuniversal (i.e., existential or leaf) node  $w$  reachable along some path from  $v$  to a leaf node of  $T$  is assigned level  $i + 1$ , and (d) for all other nodes of  $T$ , the concept of levels is insignificant to this work and so the levels are undefined. We term the nonleaf nodes for which levels are defined as the *salient* nodes in the computation tree of an ATM. For any  $k \in \mathbb{N}^+$ , a  $k$ -level ATM is one for which, on any input, the maximum level assigned to a salient node in the computation tree of the ATM is at most  $k$ .

For every polynomial  $p(\cdot)$  and for every predicate  $R(x, y, z)$  of variables  $x, y, z$ , we use  $(\exists^{p!}y)(\forall^{p!}z)R(x, y, z)$  to indicate that there exists a unique value  $y_1$  for the  $y$  variable with  $|y_1| \leq p(|x|)$ , such that for all values  $z_1$  for the  $z$  variable with  $|z_1| \leq p(|x|)$ ,  $R(x, y_1, z_1)$  is true, and for all values  $y_2 \neq y_1$  for the  $y$  variable with  $|y_2| \leq p(|x|)$ , there exists a unique value  $z(y_2)$  for the  $z$  variable with  $|z(y_2)| \leq p(|x|)$ , such that  $R(x, y_2, z(y_2))$  is false. In the same way, we interpret expressions, such as  $(\exists^{p!}y_1)(\forall^{p!}y_2)(\exists^{p!}y_3) \dots R(x, y_1, y_2, y_3, \dots)$ , with an arbitrary number of unambiguous alternations.

### Definition 1 (Unambiguity Based Hierarchies [17, 18]).

1. The alternating unambiguous polynomial hierarchy  $AUPH =_{df} \bigcup_{k \geq 0} AU\Sigma_k^p$ , where  $AU\Sigma_0^p =_{df} P$  and for every  $k \geq 1$ ,  $AU\Sigma_k^p$  is the class of all sets  $L \subseteq \Sigma^*$  for which there exist a polynomial  $p(\cdot)$  and a polynomial-time computable predicate  $R$  such that, for all  $x \in \Sigma^*$ ,

$$\begin{aligned} x \in L &\implies (\exists^{p!}y_1)(\forall^{p!}y_2) \dots (Q^{p!}y_k)R(x, y_1, y_2, \dots, y_k), \text{ and} \\ x \notin L &\implies (\forall^{p!}y_1)(\exists^{p!}y_2) \dots (\overline{Q}^{p!}y_k)\neg R(x, y_1, y_2, \dots, y_k), \end{aligned}$$

where  $Q = \exists$  and  $\overline{Q} = \forall$  if  $k$  is odd, and  $Q = \forall$  and  $\overline{Q} = \exists$  if  $k$  is even.

2. The unambiguous polynomial hierarchy is  $UPH =_{df} \bigcup_{k \geq 0} U\Sigma_k^p$ , where  $U\Sigma_0^p =_{df} P$  and for every  $k \geq 1$ ,  $U\Sigma_k^p =_{df} UP^{\Sigma_{k-1}^p}$ .
3. The promise unambiguous polynomial hierarchy is  $UPH =_{df} \bigcup_{k \geq 0} \mathcal{U}\Sigma_k^p$ , where  $\mathcal{U}\Sigma_0^p =_{df} P$ ,  $\mathcal{U}\Sigma_1^p =_{df} UP$ , and for every  $k \geq 2$ ,  $\mathcal{U}\Sigma_k^p$  is the class of all sets  $L \in \Sigma_k^p$  such that for some oracle NPTMs  $N_1, N_2, \dots, N_k$ ,

$L = L(N_1^{L(N_2^{L(N_k)})})$ , and for every  $x \in \Sigma^*$  and for every  $1 \leq i \leq k - 1$ ,  $N_1^{L(N_2^{L(N_k)})}(x)$  has at most one accepting path and if  $N_i$  asks a query  $w$  to its oracle  $L(N_{i+1}^{L(N_k)})$  during the computation of  $N_1^{L(N_2^{L(N_k)})}(x)$ , then  $N_{i+1}^{L(N_k)}(w)$  has at most one accepting path.

**Definition 2.** [18] UAP is the class of all sets accepted by unambiguous ATMs in polynomial time.

**Theorem 3.** 1. For all  $k \geq 0$ ,  $\text{AU}\Sigma_k^p \subseteq \text{U}\Sigma_k^p \subseteq \text{U}\Sigma_k^p \subseteq \Sigma_k^p$  [17].  
 2. For all  $k \geq 1$ ,  $\text{UP}_{<k} \subseteq \text{AU}\Sigma_k^p \subseteq \text{U}\Sigma_k^p \subseteq \text{U}\Sigma_k^p \subseteq \text{UAP}$  ([7] + [17]).  
 3.  $\text{Few} \subseteq \text{UAP} \subseteq \text{SPP}$  ([17] + [18]).

### 3 Relativized Separations of Unambiguity Based Hierarchies

In this section, we apply random restrictions of circuits for separating the levels of unambiguity based hierarchies. Sheu and Long [22] constructed an oracle  $\mathcal{A}$  relative to which UP contains a language that is not in any level of the low hierarchy in NP. Formally, Sheu and Long [22] showed that  $(\exists \mathcal{A})(\forall k \geq 1)[\Sigma_k^{p, \text{UP}^{\mathcal{A}}} \not\subseteq \Sigma_k^{p, \mathcal{A}}]$ . In their proof, they introduced special kinds of random restrictions that were motivated by, but different from, the restrictions used by Håstad [12]. Using the random restrictions of Sheu and Long [22], we construct a relativized world  $\mathcal{A}$  in which the unambiguity based hierarchies—AUPH, UPH, and  $\text{UPH}$ —are infinite. This extends the separation of relativized polynomial hierarchy [25, 12] to the separations of unambiguity based relativized hierarchies. We use the same restrictions to construct an oracle  $\mathcal{A}$  relative to which UAP is not contained in the polynomial hierarchy. Our separation results imply that proving that any of the unambiguity based hierarchies extend up to a finite level or proving that UAP is contained in the polynomial hierarchy is beyond the limits of relativizable proof techniques.

We now introduce certain notions that are prevalent in the theory of circuit lower bounds. We represent the variables of a circuit by  $v_z$ , for some  $z \in \Sigma^*$ . The dual of a circuit  $C$  is obtained from  $C$  by replacing OR gates with ANDs, AND gates with ORs, variables  $x_i$  with  $\bar{x}_i$ , and variables  $\bar{x}_j$  with  $x_j$ . A restriction  $\rho$  of a circuit  $C$  is a mapping from the variables of  $C$  to  $\{0, 1, \star\}$ . We say that a restriction  $\rho$  of a circuit  $C$  is a *full restriction* if  $\rho$  assigns 0 or 1 to all the variables in  $C$ . Given a circuit  $C$  and a restriction  $\rho$ ,  $C \upharpoonright_\rho$  denotes the circuit obtained from  $C$  by substituting each variable  $x$  with  $\rho(x)$  if  $\rho(x) \neq \star$ . For every  $A \subseteq \Sigma^*$ , the restriction  $\rho_A$  on the variables  $v_z$  of a circuit  $C$  is  $\rho_A(v_z) = 1$  if  $z \in A$ , and  $\rho_A(v_z) = 0$  if  $z \notin A$ . The composition of two restrictions  $\rho_1$  and  $\rho_2$ , denoted by  $\rho_1\rho_2$ , is defined as follows: For every  $x \in \Sigma^*$ ,  $\rho_1\rho_2(x) = \rho_2(\rho_1(x))$ .

We define specialized circuits,  $\Sigma_k(m)$ -circuits and  $\Pi_k(m)$ -circuits, used for constructing relativized worlds involving  $\Sigma_k$  and  $\Pi_k$  classes.

**Definition 4.** For every  $m \geq 1$  and  $k \geq 1$ , a  $\Sigma_k(m)$ -circuit is a depth  $k + 1$  circuit with alternating OR and AND gates such that

1. the top gate, i.e., the gate at level 1, is an OR gate,
2. the number of gates at level 1 to level  $k - 1$  is bounded by  $2^m$ ,
3. the fanin of gates at level  $k + 1$  is  $\leq m$ .

A  $\Pi_k(m)$ -circuit is the dual circuit of a  $\Sigma_k(m)$ -circuit.

For every  $k \geq 1$ , we say that  $\sigma$  is a  $\Sigma_k^{p,(\cdot)}$ -predicate if there exist a predicate  $R(A; x, y_1, \dots, y_k)$  over a set variable  $A$  and string variables  $x, y_1, y_2, \dots, y_k$ , and a polynomial  $q$  such that the following hold: (i)  $R(A; x, y_1, y_2, \dots, y_k)$  is computable in polynomial time by a deterministic oracle Turing machine that uses  $A$  as the oracle and  $\langle x, y_1, \dots, y_k \rangle$  as the input and (ii) for every set  $A$  and string  $x$ ,  $\sigma(A; x)$  is true if and only if  $(\exists^q y_1)(\forall^q y_2) \dots (Q_k^q y_k) R(A; x, y_1, y_2, \dots, y_k)$  is true, where  $Q_k = \exists$  if  $k$  is odd and  $Q_k = \forall$  if  $k$  is even. We say that  $\sigma$  is a  $\Pi_k^{p,(\cdot)}$ -predicate, for  $k \geq 1$ , if  $\neg\sigma$  is a  $\Sigma_k^{p,(\cdot)}$ -predicate.

The following proposition states the relationship between  $\Sigma_k^{p,(\cdot)}$ -predicates ( $\Pi_k^{p,(\cdot)}$ -predicates) and  $\Sigma_k(m)$ -circuits (respectively,  $\Pi_k(m)$ -circuits).

**Proposition 5 (see [15, 21, 22]).** Let  $k \geq 1$ . For every  $\Sigma_k^{p,(\cdot)}$ -predicate ( $\Pi_k^{p,(\cdot)}$ -predicate)  $\sigma$ , there is a polynomial  $q(\cdot)$  such that, for all  $x \in \Sigma^*$ , there is a  $\Sigma_k(q(|x|))$ -circuit (respectively,  $\Pi_k(q(|x|))$ -circuit)  $C_{\sigma,x}$  with the following properties:

1. For every  $A \subseteq \Sigma^*$ ,  $C_{\sigma,x} \upharpoonright_{\rho_A} = 1$  if and only if  $\sigma(A; x)$  is true, and
2. if  $v_z$  represents a variable in  $C_{\sigma,x}$ , then  $|z| \leq q(|x|)$ .

Let  $\mathcal{B} = \{B_i\}_{i=1}^r$ , where  $B_i$ 's are disjoint sets that cover the variables of  $C$ , and let  $q$  be a real number between 0 and 1. Sheu and Long [22] defined two probability spaces of restrictions,  $\hat{R}_{q,\mathcal{B}}^+$  and  $\hat{R}_{q,\mathcal{B}}^-$ , and a function  $g'$  that maps a random restriction to a restriction. A random restriction  $\rho \in \hat{R}_{q,\mathcal{B}}^+$  ( $\rho \in \hat{R}_{q,\mathcal{B}}^-$ ) is defined as follows: For every  $1 \leq i \leq r$  and for every variable  $x \in B_i$ , let  $\rho(x) = \star$  with probability  $q$  and  $\rho(x) = 1$  (respectively,  $\rho(x) = 0$ ) with probability  $1 - q$ . We now define the function  $g'$  for  $\rho \in \hat{R}_{q,\mathcal{B}}^+$ . For every  $1 \leq i \leq r$ , let  $s_i = \star$  with probability  $q$  and let  $s_i = 0$  with probability  $1 - q$ . Let  $V_i \subseteq B_i$  be the set of variables  $x$  such that  $\rho(x) = \star$ .  $g'(\rho)$  selects the variable  $v$  with the highest index in  $V_i$ , assigns value  $s_i$  to  $v$ , and assigns value 1 to all other variables in  $V_i$ . The function  $g'(\rho)$  for  $\rho \in \hat{R}_{q,\mathcal{B}}^-$  is defined in an analogous way by replacing 0 with 1 and vice versa.

**Lemma 6 (Switching Lemma [22]).** Let  $C$  be a circuit consisting of an AND of ORs with bottom fanin  $\leq t$ . Let  $\mathcal{B} = \{B_i\}_{i=1}^r$  be disjoint sets that cover the variables of  $C$ , and let  $q$  be a real number between 0 and 1. Then, for a random restriction  $\rho \in \hat{R}_{q,\mathcal{B}}^+$ ,  $\text{Prob}[C \upharpoonright_{\rho g'(\rho)} \text{ is not equivalent to an OR of ANDs with bottom fanin } \leq s] \leq \alpha^s$ , where  $\alpha < 6qt$ . The above probability holds even when  $\hat{R}_{q,\mathcal{B}}^+$  is replaced by  $\hat{R}_{q,\mathcal{B}}^-$ , or when  $C$  is an OR of ANDs and is being converted to an AND of ORs.

Sheu and Long [22] defined a kind of restriction, called *U condition*, on the assignment of variables in certain circuits. A restriction  $\rho$  is said to satisfy the U condition if the following holds: At most one variable is assigned  $\star$  or 0 in each set  $B_i$  if  $\rho$  is a random restriction from  $\hat{R}_{q,B}^+$ , and at most one variable is assigned  $\star$  or 1 in each set  $B_i$  if  $\rho$  is a random restriction from  $\hat{R}_{q,B}^-$  [22]. Below, we define a *global uniqueness condition* (also called *GU condition*) on full restrictions of any circuit  $C$ .

**Definition 7.** *We say that a full restriction  $\rho$  satisfies the GU condition for a circuit  $C$ , if the assignment of variables by  $\rho$  leads to the following characteristics in the computation of  $C$ :*

1. *If an OR gate  $G_i$  in  $C$  outputs 1, then there is exactly one input gate to  $G_i$  that outputs 1, and*
2. *if an AND gate  $G_i$  in  $C$  outputs 0, then there is exactly one input gate to  $G_i$  that outputs 0.*

**Theorem 8.**  $(\exists \mathcal{A})(\forall k \geq 1)[\text{AU}\Sigma_k^{p,\mathcal{A}} \not\subseteq \Pi_k^{p,\mathcal{A}}]$ .

**Proof** Our proof is inspired from that of Theorem 4.2 (relative to some oracle  $\mathcal{D}$ , for all  $k \geq 1$ ,  $\Sigma_k^{p,\text{UP}^{\mathcal{D}}} \not\subseteq \Sigma_k^{p,\mathcal{D}}$ ) by Sheu and Long [22]. For every  $k \geq 1$ , we define a test language  $L_k(B)$  as follows:  $L_k(B) \subseteq 0^* \Sigma^*$  such that, for every  $n \in \mathbb{N}^+$ ,

$$\begin{aligned} 0^n \in L_k(B) &\implies (\exists^n !y_1)(\forall^n !y_2) \dots (Q^n !y_k) [0^k 1y_1y_2 \dots y_k \in B], \text{ and} \\ 0^n \notin L_k(B) &\implies (\forall^n !y_1)(\exists^n !y_2) \dots (\bar{Q}^n !y_k) [0^k 1y_1y_2 \dots y_k \notin B], \end{aligned}$$

where  $Q = \exists$  and  $\bar{Q} = \forall$  if  $k$  is odd, and  $Q = \forall$  and  $\bar{Q} = \exists$  if  $k$  is even. Choose  $\mathcal{O} \subseteq \Sigma^*$  such that, for every  $k \geq 1$ ,  $L_k(\mathcal{O}) = 0^*$ . For every  $k \geq 1$ , let  $\sigma_{k,1}, \sigma_{k,2}, \dots$  be an enumeration of  $\Sigma_k^{p,(\cdot)}$ -predicates. In stage  $\langle k, i \rangle$ , we diagonalize against  $\sigma_{k,i}$  and change  $\mathcal{O}$  at a certain length. Finally, let  $\mathcal{A} := \lim_{n \rightarrow \infty} \cup_{n \in \mathbb{N}^+} \mathcal{O}^{=n}$ . We now define the stages involved in the construction of the oracle.

**Stage  $\langle k, i \rangle$ :** Choose a very large integer  $n$  so that the construction in this stage does not spoil the constructions in previous stages. Also,  $n$  must be large enough to meet the requirements in the proof of Claim 1. Set  $\mathcal{O} := \mathcal{O}_{-\Sigma^{k(n+1)+1}}$ . Choose a set  $B \subseteq 0^k 1 \Sigma^{kn}$  such that the following requirement is satisfied:

$$0^n \in L_k(B) \iff \sigma_{k,i}(\mathcal{O} \cup B; 0^n) \text{ is true.} \quad (1)$$

In Claim 1, we show that there is always a set  $B \subseteq 0^k 1 \Sigma^{kn}$  satisfying Eqn. (1). Let  $\mathcal{O} := \mathcal{O} \cup B$  and move to the next stage.

**End of Stage**

Clearly, the existence of a set  $B$  satisfying Eqn. (1) suffices to successfully finish stage  $\langle k, i \rangle$ . We now prove the statement in Claim 1.

**Claim 1** *In every stage  $\langle k, i \rangle$ , there is a set  $B \subseteq 0^k 1 \Sigma^{kn}$  satisfying Eqn. (1).*

**Proof** Assume to the contrary that in some stage  $\langle k, i \rangle$ , Eqn. (1) is not satisfied. Then, the following holds: For every  $B \subseteq 0^k 1 \Sigma^{kn}$ ,  $0^n \in L_k(B)$  if and only if  $\neg \sigma_{k,i}(\mathcal{O} \cup B; 0^n)$  is true. We define a  $C(n, k)$  circuit as follows: The depth of  $C(n, k)$  is  $k$ , the top gate of  $C(n, k)$  is an OR gate, the fanin of all the gates at level 1 to  $k$  is  $2^n$ , and every leaf of  $C(n, k)$  is a positive variable represented by  $v_z$ , where  $z \in 0^k 1 \Sigma^{kn}$ . The following proposition is evident.

**Proposition 9.** For every  $B \subseteq 0^k 1 \Sigma^{kn}$ ,

$$0^n \in L_k(B) \iff [\rho_B \text{ satisfies the GU condition for } C(n, k) \text{ and } C(n, k) \upharpoonright_{\rho_B} = 1],$$

and

$$0^n \notin L_k(B) \iff [\rho_B \text{ satisfies the GU condition for } C(n, k) \text{ and } C(n, k) \upharpoonright_{\rho_B} = 0].$$

For every  $h \geq 1$ , we define a family of circuits  $\mathcal{F}_k^h$ . Ko [15] defined a  $C_k^h$  circuit to be a depth  $k$  circuit in  $\mathcal{F}_k^h$  with fanin of gates at level  $k$  exactly equal to  $\sqrt{h}$  and used these circuits to separate the relativized polynomial hierarchy.

**Family  $\mathcal{F}_k^h$  of circuits, where  $h \geq 1$ :** A circuit  $C$  of depth  $\ell$ , where  $1 \leq \ell \leq k$ , is in  $\mathcal{F}_k^h$  if and only if the following holds:

1.  $C$  has alternating OR and AND gates, and the top gate, i.e., the gate at level 1, of  $C$  is an OR gate,
2. the fanin of gates at level 1 to  $\ell - 1$  is  $h$ ,
3. the fanin of gates at level  $\ell$  is  $\geq \sqrt{h}$ ,
4. every leaf of  $C$  is a unique positive variable.

Let  $C_{\sigma_{k,i}}$  be the  $\Pi_k(p_i(n))$ -circuit corresponding to  $\neg \sigma_{k,i}((\cdot); 0^n)$ , for some polynomial  $p_i(\cdot)$ . From Proposition 9, we wish to find a set  $B \subseteq 0^k 1 \Sigma^{kn}$  such that (i) if  $\rho_B$  satisfies the *GU condition* for  $C(n, k)$  and  $C(n, k) \upharpoonright_{\rho_B} = 1$ , then  $C_{\sigma_{k,i}} \upharpoonright_{\rho_{\mathcal{O} \cup B}} = 0$ , and (ii) if  $\rho_B$  satisfies the *GU condition* for  $C(n, k)$  and  $C(n, k) \upharpoonright_{\rho_B} = 0$ , then  $C_{\sigma_{k,i}} \upharpoonright_{\rho_{\mathcal{O} \cup B}} = 1$ . Clearly, the existence of a set  $B$  satisfying (i) and (ii) suffices to prove the claim. Next, we describe our approach to show the existence of such a set  $B$ .

We define a restriction  $\hat{\rho}_{\mathcal{O}}$  on  $C_{\sigma_{k,i}}$  as follows: For every variable  $v_z$  in  $C_{\sigma_{k,i}}$ , if  $z \in \mathcal{O}$  then let  $\hat{\rho}_{\mathcal{O}}(v_z) = 1$ , if  $z \notin \mathcal{O} \cup 0^k 1 \Sigma^{kn}$  then let  $\hat{\rho}_{\mathcal{O}}(v_z) = 0$ , and if  $z \in 0^k 1 \Sigma^{kn}$  then let  $\hat{\rho}_{\mathcal{O}}(v_z) = \star$ . Let  $C_{\sigma_{k,i}(\mathcal{O})} =_{df} C_{\sigma_{k,i}} \upharpoonright_{\hat{\rho}_{\mathcal{O}}}$ . Thus, the only variables  $v_z$  appearing in  $C_{\sigma_{k,i}(\mathcal{O})}$  are the ones for which  $z \in 0^k 1 \Sigma^{kn}$ . Suppose that no set  $B \subseteq 0^k 1 \Sigma^{kn}$  satisfying (i) and (ii) exists. Then, the following holds: For every  $B \subseteq 0^k 1 \Sigma^{kn}$ ,

$$\rho_B \text{ satisfies the GU condition for } C(n, k) \text{ and } C(n, k) \upharpoonright_{\rho_B} = C_{\sigma_{k,i}(\mathcal{O})} \upharpoonright_{\rho_B} = 1,$$

or

$$\rho_B \text{ satisfies the GU condition for } C(n, k) \text{ and } C(n, k) \upharpoonright_{\rho_B} = C_{\sigma_{k,i}(\mathcal{O})} \upharpoonright_{\rho_B} = 0. \quad (2)$$

**Lemma 10.** For every  $1 \leq \ell \leq k$  and for all sufficiently large  $h$ , for any circuit  $C_{\mathcal{F}} \in \mathcal{F}_k^h$  of depth  $\ell$ , and for any  $\Pi_{\ell}(m)$ -circuit  $C_{\pi}$ , if it holds that

(for every full restriction  $\rho$  satisfying the GU condition for  $C_{\mathcal{F}}$ )  $[C_{\mathcal{F}}]_{\rho} = C_{\pi}[\rho]$ , then  $m \geq \delta \cdot h^{1/3}$ , where  $\delta = 1/12$ .

Since  $C(n, k) \in \mathcal{F}_k^{2^n}$ ,  $C_{\sigma_{k,i}(\emptyset)}$  is a  $\Pi_k(p_i(n))$  circuit, and  $p_i(n) = o(2^{n/3})$ , we get a contradiction with Eqn. (2) and Lemma 10. ■ (Claim 1 and Theorem 8)

**Corollary 11.** *There is an oracle  $\mathcal{A}$  relative to which the alternating unambiguous polynomial hierarchy AUPH, the unambiguous polynomial hierarchy UPH, the promise unambiguous polynomial hierarchy UPH, and the polynomial hierarchy PH are infinite.*

Note that Theorem 8 does not imply relativized separation of UAP from PH in any obvious way. We achieve this separation, using the proof techniques of Theorem 8, in Theorem 12.

**Theorem 12.**  $(\exists \mathcal{A})[\text{UAP}^{\mathcal{A}} \not\subseteq \text{PH}^{\mathcal{A}}]$ .

Crăsmaru et al. [7] showed that there is an oracle relative to which  $\text{UAP} \neq \mathcal{U}\Sigma_2^p$ . Corollary 13 shows that in some relativized world, UAP is much more powerful than the promise unambiguous polynomial hierarchy UPH. Thus, Corollary 13 is a strengthening of their result.

**Corollary 13.** *There is an oracle relative to which  $\text{UPH} \subset \text{UAP}$ .*

## 4 Complexity of Unambiguous Alternating Solution

Wagner studied the class  $\nabla\text{P}$ , denoted by UAS in this paper, of all sets that are accepted by polynomial-time alternating Turing machines with partially defined AND and OR functions. UAS is a natural class with complete sets and is related to UAP in the same way as US [4] is related to UP. We define a variant of UAS, denoted by  $\text{UAS}(k)$ , where the number of alternations allowed is bounded by some constant  $k \geq 1$ , instead of the unbounded number of alternations in the definition of UAS. (Thus,  $\text{UAS}(1)$  is the same as the unique solution class US.)

**Definition 14.** [24] *The class UAS, denoted by  $\nabla\text{P}$  in [24], is the class of all sets  $L \subseteq \Sigma^*$  for which there exist polynomials  $p(\cdot)$  and  $q(\cdot)$ , and a polynomial-time computable predicate  $R$  such that, for all  $x \in \Sigma^*$ ,*

$$x \in L \iff (\exists^{p!} y_1)(\forall^{q!} y_2) \dots (Q^{p!} y_q) R(x, y_1, y_2, \dots, y_q),$$

where  $Q = \exists$  if  $q(|x|)$  is odd and  $Q = \forall$  if  $q(|x|)$  is even.

The class  $\text{UAS}(k)$ , for every  $k \geq 1$ , consists of all sets for which strings in the set are accepted unambiguously by some polynomial-time alternating Turing machine  $N$  with at most  $k$  alternations, while strings not in the set either are rejected or are accepted with ambiguity by  $N$ . A formal definition is as follows.

**Definition 15.** The class  $\text{UAS}(k)$ , for  $k \geq 1$ , is the class of all sets  $L \subseteq \Sigma^*$  for which there exist a polynomial  $p(\cdot)$  and a polynomial-time computable predicate  $R$  such that, for all  $x \in \Sigma^*$ ,

$$x \in L \iff (\exists^{p!}y_1)(\forall^{p!}y_2) \dots (Q^{p!}y_k)R(x, y_1, y_2, \dots, y_k)$$

where  $Q = \exists$  if  $k$  is odd and  $Q = \forall$  if  $k$  is even.

**Theorem 16.** 1.  $\text{US} \subseteq \text{UAS} \subseteq \text{C} = \text{P}$  and  $\text{UAS} \subseteq \forall \oplus \text{P}$  [24].  
 2. For every  $k \geq 1$ ,  $\text{UP} \subseteq \text{US} \subseteq \text{UAS}(k) \subseteq \text{UAS}(k+1) \subseteq \text{UAS}$ .  
 3. For every  $k \geq 1$ ,  $\text{AU}\Sigma_k^p \subseteq \text{UAS}(k) \subseteq \text{P}^{\Sigma_k^p}$ .

Recall from Theorem 3(2) that  $\text{UP}_{\leq k} \subseteq \text{AU}\Sigma_k^p$ , for every  $k \geq 1$ . Thus, it follows from Theorem 16(3) that  $\text{UP}_{\leq k} \subseteq \text{UAS}(k)$ . However, Theorem 17 shows that relative to an oracle  $\mathcal{A}$ , for all  $k \geq 1$ ,  $\text{UP}_{\leq k+1}$  is not contained in  $\text{UAS}(k)$ . Thus relative to the same oracle, the bounded ambiguity classes  $\text{UP}_{\leq k}$  and the bounded-level unambiguous alternating solution classes  $\text{UAS}(k)$ , for  $k \geq 1$ , form infinite hierarchies. Theorem 17 also implies that there is a relativized world where for all  $k \geq 1$ ,  $\text{UP}_{\leq k+1}$  is not contained in  $\text{AU}\Sigma_k^p$ . In contrast, Lange and Rossmanith [17] proved that  $\text{FewP} \subseteq \mathcal{U}\Sigma_2^p$  in every relativized world. It follows that relative to the oracle of Theorem 17, for all  $k \geq 1$ ,  $\mathcal{U}\Sigma_2^p \not\subseteq \text{AU}\Sigma_k^p$ .

**Theorem 17.**  $(\exists \mathcal{A})(\forall k \geq 1)[\text{UP}_{\leq k+1}^{\mathcal{A}} \not\subseteq \text{UAS}(k)^{\mathcal{A}}]$ .

**Corollary 18.** There is an oracle  $\mathcal{A}$  such that, for every  $k \geq 1$ ,  $\text{UP}_{\leq k}^{\mathcal{A}} \subset \text{UP}_{\leq k+1}^{\mathcal{A}}$ ,  $\text{AU}\Sigma_k^{p,\mathcal{A}} \subset \text{AU}\Sigma_{k+1}^{p,\mathcal{A}}$ ,  $\text{UAS}^{\mathcal{A}}(k) \subset \text{UAS}^{\mathcal{A}}(k+1)$ , and  $\mathcal{U}\Sigma_2^{p,\mathcal{A}} \not\subseteq \text{AU}\Sigma_k^{p,\mathcal{A}}$ .

## 5 Power of Robustly Unambiguous Alternating Machines

Hartmanis and Hemachandra [11] showed that robustly categorical nondeterministic polynomial-time Turing machines (i.e., NPTMs that for no oracle and no input have more than one accepting path) accept simple languages in the sense that, for every oracle  $\mathcal{A}$ , the languages accepted by such machines are in  $\text{P}^{\text{NP} \oplus \mathcal{A}}$ . Thus, if  $\text{P} = \text{NP}$ , then NPTMs satisfying robustly categorical property cannot separate  $\text{P}^{\mathcal{A}}$  from  $\text{NP}^{\mathcal{A}}$ , for any oracle  $\mathcal{A}$ . Theorem 19 generalizes this result of Hartmanis and Hemachandra [11] and shows that, for every oracle  $\mathcal{A}$ , robustly  $k$ -level unambiguous polynomial-time alternating Turing machines accept languages that are in  $\text{P}^{\Sigma_k^p \oplus \mathcal{A}}$ . Thus, similar to the case of robustly categorical NPTMs, if  $\text{P} = \text{NP}$ , then robustly  $k$ -level unambiguous polynomial-time alternating Turing machines cannot separate  $\text{P}^{\mathcal{A}}$  from  $\Sigma_k^{p,\mathcal{A}}$ , and consequently cannot separate  $\text{P}^{\mathcal{A}}$  from  $\text{NP}^{\mathcal{A}}$ .

**Theorem 19.** For all  $k \in \mathbb{N}^+$ , the following holds:

$$(\forall \mathcal{A})[N^{\mathcal{A}} \text{ is a } k\text{-level unambiguous polynomial-time ATM}] \implies (\forall \mathcal{A})[L(N^{\mathcal{A}}) \in \text{P}^{\Sigma_k^p \oplus \mathcal{A}}].$$

**Corollary 20.** For all  $k \in \mathbb{N}^+$ , if  $\text{P} = \text{NP}$  and  $(\forall \mathcal{A})[N^{\mathcal{A}} \text{ is a } k\text{-level unambiguous polynomial-time ATM}]$ , then  $(\forall \mathcal{A})[L(N^{\mathcal{A}}) \in \text{P}^{\mathcal{A}}]$ .

Crescenzi and Silvestri [8] showed that languages accepted by robustly complementary and categorical oracle NPTMs are in  $P^{(UP \cup \text{co}UP) \oplus \mathcal{A}}$ . In fact, their proof actually shows that the languages of such machines are in  $P^{(UP \cap \text{co}UP) \oplus \mathcal{A}}$ . Theorem 21 is a generalization of this result of Crescenzi and Silvestri [8] for robustly bounded-level unambiguous polynomial-time alternating Turing machines.

**Theorem 21.** *For all  $k_i, k_j \in \mathbb{N}^+$ , if for all oracles  $\mathcal{A}$ ,  $N_i^{\mathcal{A}}$  and  $N_j^{\mathcal{A}}$  are, respectively,  $k_i$ -level and  $k_j$ -level unambiguous polynomial-time ATMs and  $L(N_i^{\mathcal{A}}) = \overline{L(N_j^{\mathcal{A}})}$ , then for all oracles  $\mathcal{A}$ ,  $L(N_i^{\mathcal{A}}) \in P^{(UP^{\Sigma_{k-1}^p} \cap \text{co}UP^{\Sigma_{k-1}^p}) \oplus \mathcal{A}}$ , where  $k = \max\{k_i, k_j\}$ .*

## 6 Open Questions

We now mention some future research directions. Theorem 8 implies that there is a relativized world where the unambiguity based hierarchies are infinite. However, a number of questions related to the relativized structure of unambiguity based hierarchies remain open. For instance, is there a relativized world where AUPH is finite, but UPH and  $UPH$  are infinite? Is there a relativized world where the polynomial hierarchy is infinite, but AUPH and UPH collapse?

Hemaspaandra and Rothe [13] showed that if UP has a sparse Turing-complete set, then for every  $k \geq 3$ ,  $U\Sigma_k^p \subseteq \mathcal{U}\Sigma_{k-1}^p$ . Are there other complexity-theoretic assumptions that can help in concluding about the structure of unambiguity based hierarchies?

Fortnow [9] showed that  $PH \subset SPP$  relative to a random oracle. Theorem 12 shows that there is a relativized world where  $UAP \not\subseteq PH$ . Can we extend the oracle separation of UAP from PH to a random oracle separation?

Aida et al. [1] and Crăsmaru et al. [7] discussed whether UAP equals SPP. In fact, Crăsmaru et al. [7] pointed out their difficulty in building an oracle  $\mathcal{A}$  such that  $UAP^{\mathcal{A}} \neq SPP^{\mathcal{A}}$ . Can the ideas involved in oracle constructions in this paper be used to attack this problem?

Finally, is it the case that similar to robustly bounded-level unambiguous polynomial-time ATMs, robustly unbounded-level unambiguous polynomial-time ATMs require weak oracle access in every relativized world?

**Acknowledgment** We thank Lane Hemaspaandra for helpful advice and guidance throughout the project.

## References

1. S. Aida, M. Crăsmaru, K. Regan, and O. Watanabe. Games with uniqueness properties. *Theory of Computing Systems*, 37(1):29–47, 2004.
2. V. Arvind and P. Kurur. Graph isomorphism is in SPP. In *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science*, pages 743–750, Los Alamitos, November 16–19 2002. IEEE Computer Society.

3. R. Beigel. On the relativized power of additional accepting paths. In *Proceedings of the 4th Structure in Complexity Theory Conference*, pages 216–224. IEEE Computer Society Press, June 1989.
4. A. Blass and Y. Gurevich. On the unique satisfiability problem. *Information and Control*, 55(1–3):80–88, 1982.
5. J. Cai, T. Gundermann, J. Hartmanis, L. Hemachandra, V. Sewelson, K. Wagner, and G. Wechsung. The boolean hierarchy II: Applications. *SIAM Journal on Computing*, 18(1):95–111, 1989.
6. A. Chandra, D. Kozen, and L. Stockmeyer. Alternation. *Journal of the ACM*, 26(1), 1981.
7. M. Crăsmaru, C. Glaßer, K. Regan, and S. Sengupta. A protocol for serializing unique strategies. In *Proceedings of the 29th International Symposium on Mathematical Foundations of Computer Science*. Springer-Verlag *Lecture Notes in Computer Science #3153*, August 2004.
8. P. Crescenzi and R. Silvestri. Sperner’s lemma and robust machines. *Computational Complexity*, 7:163–173, 1998.
9. L. Fortnow. Relativized worlds with an infinite hierarchy. *Information Processing Letters*, 69(6):309–313, 1999.
10. J. Grollmann and A. Selman. Complexity measures for public-key cryptosystems. *SIAM Journal on Computing*, 17(2):309–335, 1988.
11. J. Hartmanis and L. Hemachandra. Robust machines accept easy sets. *Theoretical Computer Science*, 74(2):217–225, 1990.
12. J. Håstad. *Computational Limitations of Small-Depth Circuits*. MIT Press, 1987.
13. L. Hemaspaandra and J. Rothe. Unambiguous computation: Boolean hierarchies and sparse Turing-complete sets. *SIAM Journal on Computing*, 26(3):634–653, 1997.
14. K. Ko. On some natural complete operators. *Theoretical Computer Science*, 37(1):1–30, 1985.
15. K. Ko. Relativized polynomial time hierarchies having exactly  $k$  levels. *SIAM Journal on Computing*, 18(2):392–408, 1989.
16. K. Ko. Separating the low and high hierarchies by oracles. *Information and Computation*, 90(2):156–177, 1991.
17. K.-J. Lange and P. Rossmanith. Unambiguous polynomial hierarchies and exponential size. In *Proceedings of the 9th Structure in Complexity Theory Conference*, pages 106–115. IEEE Computer Society Press, June/July 1994.
18. R. Niedermeier and P. Rossmanith. Unambiguous computations and locally definable acceptance types. *Theoretical Computer Science*, 194(1–2):137–161, 1998.
19. M. Ogiwara and L. Hemachandra. A complexity theory for feasible closure properties. *Journal of Computer and System Sciences*, 46(3):295–325, 1993.
20. C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.
21. M. Sheu and T. Long. The extended low hierarchy is an infinite hierarchy. *SIAM Journal on Computing*, 23(3):488–509, 1994.
22. M. Sheu and T. Long. UP and the low and high hierarchies: A relativized separation. *Mathematical Systems Theory*, 29(5):423–449, 1996.
23. L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3:1–22, 1976.
24. K. Wagner. Alternating machines using partially defined “AND” and “OR”. Technical Report 39, Institut für Informatik, Universität Würzburg, January 1992.
25. A. Yao. Separating the polynomial-time hierarchy by oracles. In *Proceedings of the 26th IEEE Symposium on Foundations of Computer Science*, pages 1–10, 1985.