

Quantum and Classical Complexity Classes: Separations, Collapses, and Closure Properties

Holger Spakowski^{1*}, Mayur Thakur^{2**}, and Rahul Tripathi^{2***}

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

² Department of Computer Science, University of Rochester, Rochester, NY 14627, USA.
{thakur, rahult}@cs.rochester.edu

Abstract. We study the complexity of quantum complexity classes like EQP, BQP, NQP (quantum analogs of P, BPP, and NP, respectively) using classical complexity classes like ZPP, WPP, C=P. The contributions of this paper are threefold. First, we show that relative to an oracle, ZPP is not contained in WPP. As an immediate consequence, this implies that no relativizable proof technique can improve the best known classical upper bound for BQP (BQP \subseteq AWPP [16]) to BQP \subseteq WPP and the best known classical lower bound for EQP (P \subseteq EQP) to ZPP \subseteq EQP. Second, we extend some known oracle constructions involving counting and quantum complexity classes to immunity separations. Third, motivated by the fact that counting classes (like LWPP, AWPP, etc.) are the best known classical upper bounds on quantum complexity classes, we study properties of these counting classes. We prove that WPP is closed under polynomial-time truth-table reductions, while we construct an oracle relative to which WPP is not closed under polynomial-time Turing reductions. This shows that proving the equality of the similar appearing classes LWPP and WPP would require nonrelativizable techniques. We also prove that both AWPP and APP are closed under \leq_T^{UP} reductions, and use these closure properties to prove strong consequences of the following hypotheses: NQP \subseteq BQP and EQP = NQP.

1 Introduction

Quantum complexity classes like EQP, BQP [4] (quantum analogs, respectively, of P and BPP [17]), and NQP [1] (the quantum analog of NP) are defined using quantum Turing machines, the quantum analog of classical Turing machines. EQP is the class of languages L accepted by a quantum Turing machine M running in polynomial time such that, for each $x \in \Sigma^*$, if $x \in L$, then the probability that $M(x)$ accepts is 1, and if $x \notin L$, then the probability that $M(x)$ accepts is 0. BQP is the class of languages L accepted by a quantum Turing machine M running in polynomial time such that, for each $x \in \Sigma^*$, if $x \in L$, then the probability that $M(x)$ accepts is at least $2/3$, and

* Research supported in part by a grant from the DAAD. Work done while visiting the University of Rochester.

** Supported in part by grant NSF-INT-9815095/DAAD-315-PPP-gü-ab.

*** Supported in part by grant NSF-INT-9815095/DAAD-315-PPP-gü-ab.

if $x \notin L$, then the probability that $M(x)$ accepts is at most $1/3$. NQP is the class of languages L accepted by a quantum Turing machine M running in polynomial time such that, for each $x \in \Sigma^*$, $x \in L$ if and only if the probability that $M(x)$ accepts is nonzero.

Quantum complexity classes represent the computational power of quantum computers. Some fundamental computational problems—for example, factoring, discrete logarithm [33], Pell’s equation, and principal ideal problem [22]—are not believed to be in BPP (and thus, not believed to be in P), and yet are provably in BQP. One of the key issues in quantum complexity theory is studying the relationship between classical and quantum complexity classes. The inclusion relationships of BQP with some natural classical complexity classes are known. Bernstein and Vazirani [4] show that $\text{BPP} \subseteq \text{BQP} \subseteq \text{P}^{\#\text{P}}$. Adleman, DeMarrais, and Huang [1] improve that to $\text{BQP} \subseteq \text{PP}$. Fortnow and Rogers [16] show that the study of counting classes can give us insights into the classical complexity of quantum complexity classes. In particular, they study the complexity of BQP using gap-definable counting classes [12]. Loosely speaking, gap-definable counting classes capture the power of computing via counting the gap (i.e., difference) between the number of accepting and rejecting paths in a nondeterministic polynomial-time Turing machine. Fortnow and Rogers prove that $\text{BQP} \subseteq \text{AWPP}$, where AWPP is a gap-definable counting class. Since $\text{AWPP} \subseteq \text{PP}$, they give a better upper bound on BQP than that of Adleman, DeMarrais, and Huang. Thus, the best known lower and upper bounds for BQP in terms of classical complexity classes are, respectively, BPP and AWPP: $\text{BPP} \subseteq \text{BQP} \subseteq \text{AWPP} \subseteq \text{PP}$. Similarly the best known classical lower and upper bounds for EQP are, respectively, P and LWPP: $\text{P} \subseteq \text{EQP} \subseteq \text{LWPP} \subseteq \text{AWPP} \subseteq \text{PP}$.

In light of this connection, due to Fortnow and Rogers, between quantum and counting complexity classes, it is natural to ask if there are counting (or for that matter other classical) complexity classes that are better lower (or upper) bounds for BQP. More formally, is there a counting class \mathcal{C} such that $\text{BPP} \subseteq \mathcal{C} \subseteq \text{BQP}$? Is there a counting class \mathcal{D} such that $\text{BQP} \subseteq \mathcal{D} \subseteq \text{AWPP}$? Similarly, it is interesting to ask the corresponding questions for EQP. Unfortunately, resolving these inclusion relationships can be difficult, and may be out of reach of relativizable techniques. Green and Pruiam [20] construct an oracle relative to which $\text{EQP} \not\subseteq \text{P}^{\text{NP}}$, and thus they show that proving $\text{EQP} \subseteq \text{P}^{\text{NP}}$ is outside the scope of relativizable techniques. For each prime p and integer $k \geq 1$, de Graaf and Valiant [10] construct an oracle relative to which $\text{EQP} \not\subseteq \text{Mod}_p^k \text{P}$.

In this paper, we use counting classes to study the *relativized* complexity of EQP and BQP. In particular, we study the relativized complexity of EQP and BQP by separating counting classes relative to an oracle. We construct oracles A and B such that $\text{ZPP}^A \not\subseteq \text{WPP}^A$, and $\text{RP}^B \not\subseteq \text{C}=\text{P}^B$. It follows from known inclusions that $\text{BQP}^A \not\subseteq \text{WPP}^A$, $\text{ZPP}^A \not\subseteq \text{EQP}^A$, and $\text{BQP}^B \not\subseteq \text{C}=\text{P}^B$. Note that $\text{WPP} \subseteq \text{AWPP}$, $\text{P} \subseteq \text{ZPP}$, $\text{RP} \subseteq \text{BQP}$, and $\text{EQP} \subseteq \text{LWPP} \subseteq \text{WPP} \subseteq \text{C}=\text{P}$. In fact, WPP is the largest known natural gap-definable subclass of AWPP, and ZPP is the smallest known natural probabilistic complexity class that contains P. We also shed light on the relationship between problems in $\text{C}=\text{P}$ and those solvable by probabilistic algorithms. Even

though $C=P$ contains ZPP ($= RP \cap coRP$) in all relativized worlds, using relativizable techniques it is impossible to show that $C=P$ contains all problems in RP.

The separations of counting classes mentioned above, for example, $RP^B \not\subseteq C=P^B$, which lead to the separation of quantum complexity classes from counting complexity classes for reasons mentioned above, imply for example, that relativizable techniques cannot prove that $BQP \subseteq C=P$. However, this leaves open the possibility that each set in BQP^B can be approximated by a set in $C=P^B$ in the following sense: for each infinite set $L \in BQP^B$, there exists an infinite subset $L' \subseteq L$ such that $L' \in C=P^B$. A strong (or immunity) separation of BQP^B from $C=P^B$ will preclude this possibility. Strong separations have been used to study the relativized complexity of complexity classes in many different settings, for example, the polynomial-time hierarchy [25, 7], the boolean hierarchy over RP [8], and counting classes [32]. We prove strong separations between counting classes, and from these get strong separations of quantum complexity classes from counting classes. For example, we show the existence of oracles A and A' such that RP^A is $C=P^A$ -immune, and $BPP^{A'}$ is $P^{C=P^{A'}}$ -immune. Using known inclusions, we get that $BQP^{A'}$ is $P^{C=P^{A'}}$ -immune. We extend the oracle separation of EQP from $Mod_{p^k}P$ in [10] by constructing, for each prime p and integer $k \geq 1$, an oracle relative to which EQP is $Mod_{p^k}P$ -immune.

Results by Fortnow and Rogers [16], de Graaf and Valiant [10], and those of this paper, show the connection between quantum and counting complexity classes. Thus, it becomes important to study the properties of these counting complexity classes. In particular, we study the reduction closure properties of these counting classes. Fenner, Fortnow, and Kurtz [12] show that counting classes SPP and LWPP are closed under polynomial-time Turing reductions. (In fact, they prove that $SPP^{SPP} = SPP$, and $SPP^{LWPP} = LWPP$.) They ask whether the same holds for WPP. We prove that WPP is closed under polynomial-time truth-table reductions. We also show that improving this result to closure under polynomial-time Turing reductions will require non-relativizable techniques: There is an oracle A such that $P^{WPP^A} \not\subseteq WPP^A$. Thus, it follows that, relative to oracle A , WPP strictly contains LWPP. For counting classes AWPP and APP, we prove a potentially stronger closure property, namely that both AWPP and APP are closed under \leq_T^{UP} (unambiguous polynomial-time Turing) reductions.

Vyalyi [38] recently proved, using Toda's Theorem [36], that QMA, the class of languages such that a "yes" answer can be verified by a 1-round quantum interactive proof, is unlikely to contain PP, since if it does then PP contains PH. Using Vyalyi's result and the reduction closure results mentioned above, we prove consequences of the "NQP \subseteq BQP" and "EQP = NQP" hypotheses. Note that these hypotheses are quantum counterparts of the "NP \subseteq BPP" and the "P = NP" hypotheses. Zachos [40] proved that if NP \subseteq BPP, then PH \subseteq BPP. We prove that if NQP \subseteq BQP, then PH \subseteq AWPP, from which it follows that PH is low for PP. Similarly, we prove that EQP = NQP \implies PH \subseteq WPP, from which it follows that PH is low for PP.

Due to space limitations, most of the proofs are omitted. They can be found in [34].

2 Preliminaries

Our alphabet is $\Sigma = \{0, 1\}$. For any $n \in \mathbb{N}$ and any $x \in \Sigma^*$, $x\Sigma^n = \{xw \mid w \in \Sigma^n\}$. For any $x \in \Sigma^*$, $|x|$ denotes the length of the string x , and the integer $\text{bin}(x)$ corresponding to string x is defined as the value of the binary number $1x$.

For general complexity-theoretic background and for the definition of complexity classes such as P, NP, FP etc., we refer the reader to the handbook [23]. NPTM stands for “nondeterministic polynomial-time Turing machine” and DPTM stands for “deterministic polynomial-time Turing machine.” Throughout this paper, for any (non-deterministic or deterministic or quantum) machine N , and for any $x \in \Sigma^*$, we use $N(x)$ as a shorthand for “the computation of N on input x .” Given an oracle NPTM N and a set A , we use $\#\text{acc}_{N^A}(x)$ (respectively, $\#\text{rej}_{N^A}(x)$) to denote the number of accepting (respectively, rejecting) paths of N on x with oracle A .

Definition 1 ([12]). *If N is an NPTM, define the function $\text{gap}_N : \Sigma^* \rightarrow \mathbb{Z}$ as follows: for all $x \in \Sigma^*$, $\text{gap}_N(x) \stackrel{\text{df}}{=} \#\text{acc}_N(x) - \#\text{rej}_N(x)$. If N is an oracle NPTM then, for every set A , define the function $\text{gap}_{N^A} : \Sigma^* \rightarrow \mathbb{Z}$ as follows: for each $x \in \Sigma^*$, $\text{gap}_{N^A}(x) \stackrel{\text{df}}{=} \#\text{acc}_{N^A}(x) - \#\text{rej}_{N^A}(x)$.*

GapP is the class of functions f such that there exists an NPTM N such that $f = \text{gap}_N$. We define the following gap-definable counting classes [12].

Definition 2. 1. [9, 24, 2] For each $k \geq 2$, $\text{Mod}_k\text{P} = \{L \mid (\exists g \in \text{GapP})(\forall x \in \Sigma^*)[x \in L \iff g(x) \not\equiv 0 \pmod{k}]\}$.
 2. [30, 18] $\oplus\text{P} = \text{Mod}_2\text{P}$.
 3. [29, 12] $\text{SPP} = \{L \mid (\exists g \in \text{GapP})(\forall x \in \Sigma^*)[(x \in L \implies g(x) = 1) \wedge (x \notin L \implies g(x) = 0)]\}$.
 4. [12] $\text{LWPP} = \{L \mid (\exists g \in \text{GapP})(\exists h \in \text{FP} : 0 \notin \text{range}(h))(\forall x \in \Sigma^*)[(x \in L \implies g(x) = h(0^{|x|})) \wedge (x \notin L \implies g(x) = 0)]\}$.
 5. [12] $\text{WPP} = \{L \mid (\exists g \in \text{GapP})(\exists h \in \text{FP} : 0 \notin \text{range}(h))(\forall x \in \Sigma^*)[(x \in L \implies g(x) = h(x)) \wedge (x \notin L \implies g(x) = 0)]\}$.

The counting classes AWPP [13] and APP [27] were defined to study the sets that are low for PP.

Definition 3 ([13, 27]). *For every $L \subseteq \Sigma^*$,*

1. L is in AWPP if for every polynomial $r(\cdot) > 0$, there exist a $g \in \text{GapP}$ and a polynomial p such that, for all $x \in \Sigma^*$, $x \in L \implies (1 - 2^{-r(|x|)}) \leq \frac{g(x)}{2^{p(|x|)}} \leq 1$, and $x \notin L \implies 0 \leq \frac{g(x)}{2^{p(|x|)}} \leq 2^{-r(|x|)}$.
2. L is in APP if for every polynomial $r(\cdot) > 0$, there exist $g, h \in \text{GapP}$, $0 \notin \text{range}(h)$, such that, for all $x \in \Sigma^*$, $x \in L \implies (1 - 2^{-r(|x|)}) \leq \frac{g(x)}{h(0^{|x|})} \leq 1$, and $x \notin L \implies 0 \leq \frac{g(x)}{h(0^{|x|})} \leq 2^{-r(|x|)}$.

For background information on quantum complexity theory and for the definition of quantum Turing machine, we recommend [28, 21]. We now define the quantum complexity classes that will be used in this paper.

Definition 4 ([4, 1]). EQP (respectively, BQP, NQP) is the set of all languages $L \subseteq \Sigma^*$ such that there is a polynomial-time quantum Turing machine M such that, for each $x \in \Sigma^*$, $x \in L \implies \Pr[M(x) \text{ accepts}] = 1$ (respectively, $\geq 2/3, \neq 0$), and $x \notin L \implies \Pr[M(x) \text{ accepts}] = 0$ (respectively, $\leq 1/3, = 0$).

The following proposition gives the known inclusion relationships among the classes defined above.

Proposition 5 ([12, 17, 26, 13, 11, 16, 14, 39]). $P \subseteq ZPP \subseteq RP \subseteq BPP \subseteq AWPP$; $P \subseteq UP \subseteq \text{FewP} \subseteq SPP \subseteq LWPP \subseteq WPP \subseteq C=P \subseteq PP$; $ZPP \subseteq \text{coRP} \subseteq \text{coNP} \subseteq C=P$; $WPP \subseteq AWPP \subseteq APP \subseteq PP$; $P \subseteq EQP \subseteq LWPP$; $EQP \subseteq BQP \subseteq AWPP$; $SPP \subseteq \oplus P$; $\text{FewP} \subseteq NP \subseteq \text{coC=P} = NQP$;

3 Separation Results

One way to study the power of quantum complexity classes is to lower bound the complexity of these classes with well known complexity classes, for example NP. The best known lower bound for EQP is P. In fact, EQP is not known to contain even a single problem that is not already known to be in P. Bennett et al. [3] show that relative to a random oracle, NP is not contained in EQP with probability one, and, relative to a permutation oracle chosen uniformly at random, $NP \cap \text{coNP}$ is not contained in EQP with probability one. Thus, it is interesting to ask the following questions. Are there natural classes between P and $NP \cap \text{coNP}$ that are contained in EQP? Are there natural classes between P and $NP \cap \text{coNP}$ that are not contained in EQP in some relativized world? We prove that the latter is true by showing that there is a relativized world where ZPP is not contained in EQP. In fact, we prove a slightly stronger statement. We prove, as the next theorem, that there is an oracle relative to which ZPP is not contained in WPP, a superclass of EQP [16]. It is interesting to note that there is an oracle, due to Fortnow [15], relative to which SPP, a subclass of WPP, strictly contains an infinite polynomial-time hierarchy. In contrast, our oracle provides a completely different picture of WPP in a relativized world: a world in which WPP sets are not powerful enough to capture a seemingly small subclass, ZPP, of NP.

Theorem 6. $(\exists A) [ZPP^A \not\subseteq WPP^A]$.

Corollary 7. *There exists an oracle $A \subseteq \Sigma^*$ such that, for each $\mathcal{C} \in \{ZPP, RP, BPP, NP, BQP, C=P \cap \text{coC=P}, AWPP, APP\}$, and each $\mathcal{D} \in \{UP, \text{FewP}, SPP, EQP, LWPP, WPP\}$, $\mathcal{C}^A \not\subseteq \mathcal{D}^A$.*

Note that Corollary 7 shows that proving that error-free quantum polynomial-time (EQP) algorithms exist for all languages in ZPP will require nonrelativizable techniques. Corollary 7 also shows that, using relativizable techniques, we cannot lower the best known classical upper bound for BQP from AWPP to even WPP, the largest known natural gap-definable subclass of AWPP. In the light of this result, it is interesting to seek a *different* classical upper bound for BQP. That is, it is interesting to ask which other counting classes upper bound the complexity of BQP. One such counting class is $C=P$. Note that it is not known whether $AWPP \subseteq C=P$ or $C=P \subseteq AWPP$,

though both these classes contain WPP. Regardless of what the inclusion relationship is between $C=P$ and AWPP, it is conceivable for BQP to be subset of $C=P$. However, we show that proving the containment $BQP \subseteq C=P$ is beyond the reach of relativizable techniques. This result is a corollary of Theorem 8.

Tarui [35] used a lower bound technique in decision trees for a certain AC^0 function to show that BPP is not contained in $P^{C=P}$ in some relativized world. Green [19] used circuit lower bound techniques to obtain the same result. In contrast with BPP, RP is contained in $P^{C=P}$ in every relativized world. We construct an oracle relative to which RP is not contained in $C=P$. This result is optimal in the sense that the largest known natural subclass of RP, ZPP, is contained in $C=P$ in every relativized world. This oracle separation of RP from $C=P$ is also a strengthening of the oracle separation of NP from $C=P$ by Torán [37].

Theorem 8. $(\exists A) [RP^A \not\subseteq C=P^A]$.

Corollary 9. *There exists an oracle $A \subseteq \Sigma^*$ such that, for each $\mathcal{C} \in \{BPP, NP, BQP, AWPP, APP\}$, $\mathcal{C}^A \not\subseteq C=P^A$.*

4 Immunity Separation Results

In Section 3, we saw that relativizable techniques cannot prove that $BQP \subseteq C=P$. But can we at least prove that in every relativized world, every BQP set can be approximated (in some sense) by a set from $C=P$? For example, can we prove that every infinite set in BQP has an infinite subset that is in $C=P$? In this section, we prove that in many cases, proving such approximability results is beyond the reach of relativizable techniques. Loosely speaking, class \mathcal{C} is said to strongly separate from \mathcal{D} if there exists an oracle relative to which there exists a set S in \mathcal{C} that cannot even be approximated (in the sense mentioned above) by any set in \mathcal{D} . The set S is said to be \mathcal{D} -immune. Immunity separations have been used, for example, by Ko [25], and Bruschi [7] to study the nature of the polynomial-time hierarchy, by Bruschi, Joseph, and Young [8] for strongly separating the boolean hierarchy over RP, and by Rothe [32] for studying the complexity of counting classes.

Definition 10. *Let \mathcal{C} be a class of languages. An infinite language L is called \mathcal{C} -immune if $(\forall L' \in \mathcal{C}) [|L'| = \infty \Rightarrow L' \cap \bar{L} \neq \emptyset]$.*

Given relativizable classes \mathcal{C}_1 and \mathcal{C}_2 , an oracle E strongly separates \mathcal{C}_2^E from \mathcal{C}_1^E if there exists an infinite language $L \in \mathcal{C}_2^E$ which is \mathcal{C}_1^E -immune.

M. de Graaf and P. Valiant [10] prove that, for any prime p and integer $k \geq 1$, there exists an oracle A' such that $EQP^{A'} \not\subseteq \text{Mod}_{p^k} P^{A'}$. In Theorem 11, we strengthen this result by proving that there is a relativized world where EQP strongly separates from $\text{Mod}_{p^k} P$. To prove that the test language we use in the proof of this strong separation result (Theorem 11) is in (relativized) EQP, we make use of the observation by Boyer et al. [6] that quantum database searching can be done in polynomial time with *certainty* if the number of solutions is exactly one fourth of the total search-space.

Theorem 11. *For every prime p and integer $k \geq 1$, there exist an oracle A and an infinite set L_A such that $L_A \in \text{EQP}^A$ and L_A is $\text{Mod}_{p^k}\text{P}^A$ -immune.*

Corollary 12. *There exists an oracle A such that, for each $\mathcal{C} \in \{\text{EQP}, \text{BQP}, \text{LWPP}, \text{WPP}, \text{AWPP}, \text{APP}, \text{C=P}, \text{PP}\}$ and for each $\mathcal{D} \in \{\text{UP}, \text{FewP}, \text{SPP}\}$, \mathcal{C}^A is immune to \mathcal{D}^A .*

Theorem 8 separates RP from C=P , and as a corollary we get a separation of BQP from C=P . In Theorem 13, we prove that, relative to an oracle, RP strongly separates from C=P , which in turn implies that BQP strongly separates from C=P . We use a sufficient condition by Bovet, Crescenzi, and Silvestri [5] for lifting simple separations between complexity classes to immunity separations.

Theorem 13. *There exists an oracle A such that RP^A contains a C=P^A -immune set.*

Tarui [35] and Green [19] independently showed that BPP separates from $\text{P}^{\text{C=P}}$ in some relativized world. In Theorem 14 we extend oracle separation of BPP from $\text{P}^{\text{C=P}}$ to a strong separation result. From this it follows that, relative to an oracle, BQP strongly separates from $\text{P}^{\text{C=P}}$.

Theorem 14. *There exists an oracle A such that for every complexity class $\mathcal{C} \in \{\text{BPP}, \text{BQP}, \Sigma_2^p \cap \Pi_2^p, \text{AWPP}, \text{APP}, \text{PP}\}$, \mathcal{C}^A contains a $\text{P}^{\text{C=P}^A}$ -immune set.*

5 Closure and Collapse Results

We have seen that the study of counting complexity classes like WPP , C=P etc. can give us useful insights into the classical complexity of quantum classes. In this section, we further study properties of these counting classes, and use these properties to prove consequences of the following hypothesis: $\text{NQP} \subseteq \text{BQP}$. Note that this hypothesis is the quantum analog of the “ $\text{NP} \subseteq \text{BPP}$ ” hypothesis. Zachos [40] proved that $\text{NP} \not\subseteq \text{BPP}$ unless the entire polynomial-time hierarchy is contained in BPP , and thus it is unlikely that $\text{NP} \subseteq \text{BPP}$. In this section, we prove as Corollary 17 a strong consequence for $\text{NQP} \subseteq \text{BQP}$: $\text{NQP} \subseteq \text{BQP} \implies \text{PP}^{\text{PH}} = \text{PP}$. We prove this implication by showing a reduction closure property of AWPP , and then using a recent result by Vyalyi [38]. Recently, Vyalyi [38] showed that if QMA , the quantum analog of the Merlin-Arthur class MA , equals PP then the entire polynomial-time hierarchy is contained in PP . In his proof, he implicitly proves, using Toda’s [36] theorem, that PH is contained in $\text{UP}^{\text{C=P}}$ in every relativized world.

Theorem 15 ([38]). $\text{PH} \subseteq \text{UP}^{\text{C=P}}$.

In Theorem 16 we show that both AWPP and APP are closed under \leq_T^{UP} (*unambiguous polynomial-time Turing*) reductions. From this closure property of AWPP and Theorem 15, we conclude that if $\text{NQP} \subseteq \text{BQP}$ then PH is low for PP .

Theorem 16. (a) $\text{UP}^{\text{AWPP}} \subseteq \text{AWPP}$. (b) $\text{UP}^{\text{APP}} \subseteq \text{APP}$.

Corollary 17. *If $\text{NQP} \subseteq \text{BQP}$ then PH is low for PP .*

Proof. This follows from Theorem 15 and Theorem 16(a), and the facts that $\text{NQP} = \text{coC=P}$ [14, 39], $\text{BQP} \subseteq \text{AWPP}$ [16], and AWPP is low for PP . ■

Theorem 16 shows that AWPP is closed under \leq_T^{UP} reductions. What about WPP ? Closure properties of WPP are also interesting in light of the results due to Fenner, Fortnow, and Kurtz [12]. Fenner et al. study the closure of gap-definable classes under polynomial-time Turing reductions. They show that LWPP and SPP are closed under polynomial-time Turing reductions. However, they leave open the corresponding problem for WPP : Is WPP closed under polynomial-time Turing reductions? Theorem 21 gives a negative answer in a suitable relativized world. Since LWPP is (robustly, i.e., for all oracles) closed under polynomial-time Turing reductions [12], it follows that we have also an oracle separating the seemingly similar classes WPP and LWPP . In Theorem 18, we show that WPP is closed under the weaker polynomial-time truth-table reduction, while its closure under potentially stronger \leq_T^{UP} reduction is contained in coC=P . The later result along with Theorem 15 allows us to conclude that if EQP contains NQP , then the entire polynomial-time hierarchy is contained in WPP .

Theorem 18. (a) WPP is closed under polynomial-time truth-table reductions. (b) $\text{UP}^{\text{WPP}} \subseteq \text{coC=P}$.

Corollary 19. If $\text{NQP} \subseteq \text{EQP}$ then $\text{PH} \subseteq \text{WPP}$.

Proof. This follows from Theorem 15 and Theorem 18(b), and the facts that $\text{NQP} = \text{coC=P}$ [14, 39], $\text{EQP} \subseteq \text{LWPP}$ [16], and $\text{LWPP} \subseteq \text{WPP}$ [12]. ■

We now prove that there is an oracle relative to which WPP is not closed under polynomial-time Turing reductions. Before we state and prove Theorem 21, we state a lemma that will be needed in the proof of this result.

Lemma 20 ([31]). For every $n \geq 17$, the number of primes less than or equal to n , $\pi(n)$, satisfies

$$n / \ln n < \pi(n) < 1.25506 n / \ln n.$$

Theorem 21. $(\exists A)[\text{P}^{\text{WPP}^A} \not\subseteq \text{WPP}^A]$.

Proof. For any $w \in \Sigma^*$, let $\text{pos}(w) = \text{bin}(w) - 2^{|w|} + 1$ represent the lexicographic position of w among strings of length $|w|$. For every set $A \subseteq \Sigma^*$, $w \in \Sigma^*$, and $n \in \mathbb{N}$, we define “Witcount”, “Promise” and “Boundary” as follows.

$$\begin{aligned} \text{Witcount}(A, w) &= |\{x \in \Sigma^* \mid |x| = |w| \wedge wx \in A\}|, \\ \text{Promise}(A, n) &\equiv (\forall w \in \Sigma^n)[\text{Witcount}(A, w) = 0 \vee \text{Witcount}(A, w) = \text{pos}(w)] \wedge \\ &\quad (\forall w_1, w_2 \in \Sigma^n)[\text{pos}(w_1) \leq \text{pos}(w_2) \wedge \text{Witcount}(A, w_2) \neq 0 \Rightarrow \\ &\quad \quad \quad \text{Witcount}(A, w_1) \neq 0], \text{ and} \\ \text{Boundary}(A, n) &= \max\{\text{pos}(w) \mid |w| = n \wedge \text{Witcount}(A, w) \neq 0\}. \end{aligned}$$

For every set $A \subseteq \Sigma^*$, define L_A as follows.

$$L_A = \{0^n \mid \text{Boundary}(A, n) \equiv 1 \pmod{2}\}.$$

Clearly, if A satisfies $\text{Promise}(A, n)$ at each length n , then L_A is in PWPP^A (using binary search along the strings w with $|w| = n$). We construct an oracle A such that, for each n , $\text{Promise}(A, n)$ is true, and $L_A \notin \text{WPP}^A$. Let $(N_s, M_s, p_s)_{s \geq 1}$ be an enumeration of all triples such that N_s is a nondeterministic polynomial-time oracle Turing machine, M_s is a deterministic polynomial-time oracle transducer, p_s is a polynomial, and the running time of both N_s and M_s is bounded by p_s regardless of the oracle. We assume that the computation paths of an oracle machine include the answers from the oracle. Given NPTM N , $x \in \Sigma^*$, and a computation path $\rho \in \Sigma^*$, we let $\text{sign}(N, x, \rho) = +1$ (respectively, -1) if ρ is an accepting (respectively, rejecting) computation path in $N(x)$. We need the following technical lemmas.

Lemma 22. *Let $N, p \in \mathbb{N}$, where $1 < p \leq N/2$. Let $s(y_1, \dots, y_N)$ be a multilinear polynomial with rational coefficients, where each monomial has exactly $p - 1$ different variables. Suppose that for some $\text{val} \in \mathbb{Q}$ it holds that $s(y_1, \dots, y_N) = \text{val}$ for every $y_1, \dots, y_N \in \{0, 1\}$ with $\sum_{i=1}^N y_i = p$. Then each monomial in $s(y_1, \dots, y_N)$ has the same rational coefficient, i.e.,*

$$s(y_1, y_2, \dots, y_N) = \sum_{1 \leq i_1 < i_2 < \dots < i_{p-1} \leq N} (\text{val}/p) \cdot y_{i_1} y_{i_2} \dots y_{i_{p-1}}.$$

Lemma 23. *Let $N \in \mathbb{N}$ and p be a prime with $p \leq N/2$. Let $s(y_1, \dots, y_N)$ be a multilinear polynomial of total degree $< p$ with integer coefficients. If for some $\text{val} \in \mathbb{Z}$, it holds that*

1. $s(0, \dots, 0) = 0$, and
2. $s(y_1, \dots, y_N) = \text{val}$, for every $y_1, \dots, y_N \in \{0, 1\}$ with $\sum_{i=1}^N y_i = p$,

then $p \mid \text{val}$.

The oracle A is constructed in stages. In stage s , the membership in A of strings of length $2n_s$ is decided, and the initial segment A_{s-1} is extended to A_s . Our choice of n_s guarantees that the oracle extension in stage s does not affect the computation in earlier stages. Set $A_0 := \emptyset$ and $n_0 := 17$.

Stage s where $s \geq 1$: Let n_s be large enough so that the previous stages are not affected and $2^{n_s} > 4n_s^2 p_s(n_s)$. We diagonalize against nondeterministic polynomial-time oracle Turing machine N_s and deterministic polynomial-time oracle transducer M_s . Let val be the value computed by $M_s^{A_{s-1}}(0^{n_s})$. Without loss of generality, we assume that $\text{val} \neq 0$. Let $T = \{w \in \Sigma^{2n_s} \mid M_s^{A_{s-1}}(0^{n_s}) \text{ queries } w\}$.

(\star) Choose a set B , $B \subseteq \overline{T} \cap \Sigma^{2n_s}$, satisfying $\text{Promise}(B, n_s)$ such that the following holds:

$$\text{Boundary}(B, n_s) \equiv 1 \pmod{2} \wedge \text{gap}_{N_s^{A_{s-1} \cup B}}(0^{n_s}) \neq \text{val}, \text{ or}$$

$$\text{Boundary}(B, n_s) \equiv 0 \pmod{2} \wedge \text{gap}_{N_s^{A_{s-1} \cup B}}(0^{n_s}) \neq 0.$$

Let $A_s := A_{s-1} \cup B$. Clearly, the construction guarantees that $L_A \notin \text{WPP}^A$. The feasibility of the construction follows from the following claim.

Claim. For each $s \geq 1$, there exists an oracle extension B satisfying (\star) .

Proof. Suppose that in stage s no set B satisfying (\star) exists. Then, for every $B \subseteq \overline{T} \cap \Sigma^{2n_s}$ satisfying $\text{Promise}(B, n_s)$, the following hold.

$$\text{Boundary}(B, n_s) \equiv 1 \pmod{2} \implies \text{gap}_{N_s^{A_{s-1} \cup B}}(0^{n_s}) = \text{val}, \text{ and} \quad (1)$$

$$\text{Boundary}(B, n_s) \equiv 0 \pmod{2} \implies \text{gap}_{N_s^{A_{s-1} \cup B}}(0^{n_s}) = 0. \quad (2)$$

Let $U = \{w \in \Sigma^{n_s} \mid \text{pos}(w) \text{ is prime, and } 2^{n_s-2} \leq \text{pos}(w) \leq \frac{3}{2} \cdot 2^{n_s-2}\}$. Fix an arbitrary $w \in U$. Choose a set $C_w \subseteq \overline{T} \cap \Sigma^{2n_s}$ satisfying (a) $\text{Promise}(C_w, n_s)$, and (b) $\text{Boundary}(C_w, n_s) = \text{pos}(w) - 1$. Such a set C_w always exists because $2^{n_s} - p_s(n_s) > \frac{3}{2} \cdot 2^{n_s-2}$. Statements (1) and (2) in particular imply that, for all $D_w \subseteq \overline{T} \cap w\Sigma^{n_s}$, it holds that

$$\text{Witcount}(D_w, w) = 0 \implies \text{gap}_{N_s^{A_{s-1} \cup C_w \cup D_w}}(0^{n_s}) = 0, \text{ and} \quad (3)$$

$$\text{Witcount}(D_w, w) = \text{pos}(w) \implies \text{gap}_{N_s^{A_{s-1} \cup C_w \cup D_w}}(0^{n_s}) = \text{val}. \quad (4)$$

Henceforth, we use N to denote $|\overline{T} \cap w\Sigma^{n_s}|$. Let x_1, \dots, x_N be the lexicographic enumeration of the strings in $\overline{T} \cap w\Sigma^{n_s}$. We define s_w to be the function $\{0, 1\}^N \rightarrow \mathbb{Z}$ that has the following property. For all $D_w \subseteq \overline{T} \cap w\Sigma^{n_s}$,

$$s_w(\chi_{D_w}(x_1), \chi_{D_w}(x_2), \dots, \chi_{D_w}(x_N)) = \text{gap}_{N_s^{A_{s-1} \cup C_w \cup D_w}}(0^{n_s}). \quad (5)$$

We will show that s_w can be represented by a multilinear polynomial having low total degree. For arbitrary $z_1, \dots, z_N \in \{0, 1\}$, we call a computation path ρ of $N_s^{(\cdot)}(0^{n_s})$ “ (z_1, \dots, z_N) -allowable” if, along ρ , all queries $q \in A_{s-1} \cup C_w$ have a “yes” answer, all queries $q \notin A_{s-1} \cup C_w \cup (\overline{T} \cap w\Sigma^{n_s})$ have a “no” answer, all queries x_i with $z_i = 1$ are answered “yes”, and all queries x_i with $z_i = 0$ are answered “no”. Let $z_1, \dots, z_N \in \{0, 1\}$, and ρ be a (z_1, \dots, z_N) -allowable path of $N_s^{(\cdot)}(0^{n_s})$. Let $x_{i_1}, \dots, x_{i_\ell}$, where $\ell \leq p_s(n_s) < 2^{n_s-2}/n_s^2 < 2^{n_s-2}$, be the distinct queries to strings in $\overline{T} \cap w\Sigma^{n_s}$ along ρ . Create a monomial $\text{mono}(\rho)$ that is the product of terms γ_k , $k = 1, 2, \dots, \ell$, where $\gamma_k = y_{i_k}$ if $z_{i_k} = 1$, and $\gamma_k = (1 - y_{i_k})$ otherwise. Let

$$s'_w(y_1, \dots, y_N) = \sum_{z_1, \dots, z_N \in \{0, 1\}} \sum_{\rho: \rho \text{ is } (z_1, \dots, z_N)\text{-allowable}} \text{sign}(N_s, 0^{n_s}, \rho) \cdot \text{mono}(\rho).$$

It is easy to see that the thus constructed multilinear polynomial $s'_w(y_1, \dots, y_N)$ coincides with s_w on $\{0, 1\}^N$, and has total degree $\leq p_s(n_s) < 2^{n_s-2}/n_s^2 < \text{pos}(w) < N/2$. Statements (3) and (4) imply that for all $z_1, \dots, z_N \in \{0, 1\}$ such that $\sum_{i=1}^N z_i = \text{pos}(w)$,

$$s_w(z_1, \dots, z_N) = \text{val}, \text{ and } s_w(0, 0, \dots, 0) = 0.$$

It follows from Lemma 23 that $\text{pos}(w) \mid \text{val}$. Therefore, for each $w \in U$, $\text{pos}(w) \mid \text{val}$. Hence,

$$\text{val} \geq \prod_{w \in U} \text{pos}(w) \geq 2^{|U|} \geq 2^{\pi(\frac{3}{2} \cdot 2^{n_s-2}) - \pi(2^{n_s-2})} \geq 2^{2^{n_s-2}/n_s^2} > 2^{p_s(n_s)},$$

where the fourth inequality follows from Lemma 20 and the fifth inequality follows because, $2^{n_s} > 4n_s^2 p_s(n_s)$. However, $val \leq 2^{p_s(n_s)}$, because the running time of $M_s^{(\cdot)}(0^{n_s})$ is bounded by $p_s(n_s)$. Thus, for each $s \geq 1$, A_{s-1} can always be extended in stage s . ■ (Claim and Theorem 21)

Corollary 24. $(\exists A)[WPP^A \not\subseteq LWPP^A]$.

Acknowledgment We thank Lane Hemaspaandra for helpful advice and guidance throughout the project and anonymous referees for useful comments.

References

1. L. Adleman, J. DeMarrais, and M. Huang. Quantum computability. *SIAM Journal on Computing*, 26(5):1524–1540, 1997.
2. R. Beigel and J. Gill. Counting classes: Thresholds, parity, mods, and fewness. *Theoretical Computer Science*, 103(1):3–23, 1992.
3. C. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, October 1997.
4. E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.
5. D. Bovet, P. Crescenzi, and R. Silvestri. A uniform approach to define complexity classes. *Theoretical Computer Science*, 104(2):263–283, 1992.
6. M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4–5):493–506, 1998.
7. D. Bruschi. Strong separations of the polynomial hierarchy with oracles: Constructive separations by immune and simple sets. *Theoretical Computer Science*, 102(2):215–252, 1992.
8. D. Bruschi, D. Joseph, and P. Young. Strong separations for the boolean hierarchy over RP. *International Journal of Foundations of Computer Science*, 1(3):201–218, 1990.
9. J. Cai and L. Hemachandra. On the power of parity polynomial time. *Mathematical Systems Theory*, 23(2):95–106, 1990.
10. M. de Graaf and P. Valiant. Comparing EQP and MOD_{p^k}P using polynomial degree lower bounds. Technical Report quant-ph/0211179, Quantum Physics, 2002.
11. S. Fenner. PP-lowness and a simple definition of AWPP. *Theory of Computing Systems*, 36(2):199–212, 2003.
12. S. Fenner, L. Fortnow, and S. Kurtz. Gap-definable counting classes. *Journal of Computer and System Sciences*, 48(1):116–148, 1994.
13. S. Fenner, L. Fortnow, S. Kurtz, and L. Li. An oracle builder’s toolkit. *Information and Computation*, 182(2):95–136, 2003.
14. S. Fenner, F. Green, S. Homer, and R. Pruim. Quantum NP is hard for PH. In *Proceedings of the 6th Italian Conference for Theoretical Computer Science*, pages 241–252. World Scientific, 1998.
15. L. Fortnow. Relativized worlds with an infinite hierarchy. *Information Processing Letters*, 69(6):309–313, 1999.
16. L. Fortnow and J. Rogers. Complexity limitations on quantum computation. *Journal of Computer and System Sciences*, 59(2):240–252, 1999.
17. J. Gill. Computational complexity of probabilistic Turing machines. *SIAM Journal on Computing*, 6(4):675–695, 1977.
18. L. Goldschlager and I. Parberry. On the construction of parallel computers from various bases of boolean functions. *Theoretical Computer Science*, 43(1):43–58, 1986.

19. F. Green. On the power of deterministic reductions to $C=P$. *Mathematical Systems Theory*, 26(2):215–233, 1993.
20. F. Green and R. Pruim. Relativized separation of EQP from P^{NP} . *Information Processing Letters*, 80:257–260, 2001.
21. J. Gruska. *Quantum Computing*. McGraw-Hill, New York, 1999.
22. S. Hallgren. Polynomial-time quantum algorithms for Pell’s equation and principal ideal problem. In *Proceedings of the 34th ACM Symposium on Theory of Computing*, pages 653–658, New York, 2002. ACM Press.
23. L. Hemaspaandra and M. Ogihara. *The Complexity Theory Companion*. Springer, 2002.
24. U. Hertrampf. Relations among MOD-classes. *Theoretical Computer Science*, 74(3):325–328, 1990.
25. K. Ko. A note on separating the relativized polynomial-time hierarchy by immune sets. *RAIRO Theoretical Informatics and Applications*, 24(3):229–240, 1990.
26. J. Köbler, U. Schöning, S. Toda, and J. Torán. Turing machines with few accepting computations and low sets for PP. *Journal of Computer and System Sciences*, 44(2):272–286, 1992.
27. L. Li. On PP-low classes. Technical Report TR-93-03, Department of Computer Science, University of Chicago, May 14 1993.
28. M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. CUP, 2000.
29. M. Ogiwara and L. Hemachandra. A complexity theory for feasible closure properties. *Journal of Computer and System Sciences*, 46(3):295–325, 1993.
30. C. Papadimitriou and S. Zachos. Two remarks on the power of counting. In *Proceedings 6th GI Conference on Theoretical Computer Science*, pages 269–276. Springer-Verlag *Lecture Notes in Computer Science #145*, 1983.
31. J. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *Illinois Journal of Mathematics*, 6:64–94, 1962.
32. J. Rothe. Immunity and simplicity for exact counting and other counting classes. *RAIRO Theoretical Informatics and Applications*, 33(2):159–176, 1999.
33. P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.
34. H. Spakowski, M. Thakur, and R. Tripathi. Quantum and classical complexity classes: Separations, collapses, and closure properties. Technical Report TR801, Department of Computer Science, University of Rochester, June 2003.
35. J. Tarui. Degree complexity of boolean functions and its applications to relativized separations. In *Proceedings of the 6th Annual Conference on Structure in Complexity Theory (SCTC '91)*, pages 285–285, Chicago, IL, USA, June 1991. IEEE Computer Society Press.
36. S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 20(5):865–877, 1991.
37. J. Torán. Complexity classes defined by counting quantifiers. *Journal of the ACM*, 38(3):753–774, 1991.
38. M. Vyalı. QMA = PP implies that PP contains PH. In *ECCC’TR: Electronic Colloquium on Computational Complexity, technical reports*, 2003.
39. T. Yamakami and A. Yao. $NQP_C = co-C=P$. *Information Processing Letters*, 71(2):63–69, July 1999.
40. S. Zachos. Probabilistic quantifiers and games. *Journal of Computer and System Sciences*, 36(3):433–451, 1988.