Counting classes are at least as hard as the polynomial-time hierarchy*

Seinosuke Toda and Mitsunori Ogiwara
Dept. Computer Science and Information Mathematics
University of Electro-communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182, JAPAN
toda@cso.cs.uec.ac.jp  ogiwara@cso.cs.uec.ac.jp

Abstract

In this paper, it is shown that many natural counting classes, such as PP, C=P, and MOD_k P, are at least as computationally hard as PH (the polynomial-time hierarchy) in the following sense: for each K of the counting classes above, every set in K(Ph) is polynomial-time randomized many-one reducible to a set in K with two-sided exponentially small error probability. As a consequence of the result, we see that all the counting classes above are computationally harder than PH unless PH collapses to a finite level. Some other consequences are also shown.

1 Introduction.

In the theory of computational complexity, researchers have given much attention to several questions about the computational power of counting complexity classes such as PP defined by Gill [10], C=P defined by Wagner [26], \oplus P defined by Papadimitriou and Zachos [15], and MOD_k P defined independently by Cai and Hemachandra [7] and by Beigel, Gill and Hertrampf [4]. In those investigations, it is of particular interest to compare the computational power of the counting classes with that of classes within PH (the polynomial-time hierarchy), and the researchers have considered two different types of questions, containment questions and reducibility questions, where by a containment question, we mean to ask whether a class in PH is included in a counting class, and by a reducibility question, we mean to ask whether all sets in a class in PH are polynomial-time reducible to sets in a counting class under a suitable reducibility.

For the containment question on PP, the best result at the present time is one by Beigel, Hemachandra and Wechsung [3] that \text{P}^{\text{NP}^{\oplus \text{P}}} is included in PP. For other counting classes, it was shown in [4][7][13] that Few \subseteq \oplus P, Few \subseteq C=P, and for each prime k, Few \subseteq MOD_k P, where Few was defined by Cai and Hemachandra [7] (see their paper for the detail) and is known to be below \text{P}^{\text{NP}^{\oplus \text{P}}}. From the results, we know that Few is included in \oplus P, C=P, and MOD_k P (for prime k), and these are the best results for the containment questions on all the known counting classes other than PP. Very recently, Fenner, Fortnow and Kurtz [8] have unified and improved the results on Few versus counting classes questions. The reader may refer to their paper for the current status of the results on the questions.

For some reducibility questions, it was shown by Toda [21] that all sets in PH are polynomial-time Turing reducible to sets in PP and are polynomial-time randomized reducible to sets in \oplus P (where and throughout the paper all randomized reductions are of two-sided exponentially small error probability). It was recently shown by Ogiwara [14] that all sets in II_2^P are polynomial-time randomized many-one reducible to sets in C=P. Combining these results with a result by Schöning [17], we can conclude that PP and \oplus P are computationally harder than PH un-

---

*The research reported here was done while the authors visited the Department of Mathematics, University of California, Santa Barbara, and it was supported in part by National Science Foundation under Grant CCR89-13584.
less PH collapses to a finite level, and conclude that $C_\text{P}$ are computationally harder than $\Sigma_2^P$ unless PH collapses to a finite level.

While several relationships between the counting classes and classes in PH have been established, many relationships are currently unknown. In fact, we don’t know, at the present time, whether PP includes $\Delta_2^P$ and whether the other counting classes include NP, and didn’t know whether all sets in PH are polynomial-time reducible to sets in the counting classes other than PP and $\oplus P$. Recently some oracle sets refuting the expected containments have been found. Toran [22] found oracle sets $A$ and $B$ such that $NP(A) \not\subseteq C_\text{P}(A)$ and $NP(B) \not\subseteq \oplus P(B)$, Beigel [2] found an oracle set $C$ such that $\Delta_2^P(C) \not\subseteq PP(C)$, and he showed in [1] that for all $k \geq 2$ and some oracle set $D$, $NP(D) \not\subseteq MOD_k P(D)$. These relativization results tell us at least that all the techniques known currently do not work for settling the containment questions above, and/or that it is too difficult to solve the questions. Hence the important question at the present time is whether all sets in PH are polynomial-time reducible to sets in those counting classes under a suitable reducibility.

In this paper, we investigate the reducibility question above more deeply. We will be concerned with the polynomial-time randomized many-one reducibility with two-sided exponentially small error probability (this notion will be formalized as the $\oplus P$-operator in Section 5). We will conclude that all sets in PH are polynomial-time randomized many-one reducible to sets in a wide range of counting classes. To show it, we will first introduce a family of counting classes that is a restricted range of the gap-definable classes recently developed by Fenner, Fortnow and Kurtz [8] but is still so wide as to include PP, $C_\text{P}$, and $MOD_k P$, and we will show that all counting classes in the family are at least as hard as PH. Thus, in particular, we establish the following new relationships as immediate consequences of the general result (we also summarize some relationships known earlier for the sake of comparison of our results with them):

(1) $PP(PH) \subseteq P(PP)$ and $\oplus P(PH) \subseteq \oplus P$ [21].

(2) $\Pi_2^P \subseteq \oplus P \cdot C_\text{P}$ [14]. We here note the inclusion $\Sigma_2^P \cup \Pi_2^P \subseteq \oplus P \cdot PP$ immediately follows from this result.

(3) $C_\text{P}(PH) \subseteq \oplus P \cdot C_\text{P}$ and $PP(PH) \subseteq \oplus P \cdot PP$. These improve the relationships in (2) above.

(4) For all integers $k \geq 2$, $MOD_k P(PH) \subseteq \oplus P \cdot MOD_k P$. This extends the second result in (1) above.

As an immediate consequence, we know that PH is included in $\oplus P \cdot K$ for each $K \in \{ PP, C_\text{P}, MOD_k P \}$.

Remark. We must here give some remarks on the results to the reader. After seeing an earlier version of the present paper, Richard Beigel told us, as a private communication, that the result (4) could be obtained by using the result in [4, Theorem 27] that $MOD_k P(MOD_k P) = MOD_k P$ if $k$ is a prime, the characterization of $MOD_k P$ in [4, Corollary 33] (every MOD-class is a finite union of some MOD-classes with prime modulo), and the modulo amplification technique in [21].

An important remark is that Jun Tarui [19] has independently developed the same techniques as this paper, and in fact he observed somewhat stronger relationships than ours. Roughly speaking, he has strengthened the relationships in (3) and (4) above via the polynomial-time randomized many-one reducibility with zero error probability.

This paper will proceed as follows. In Section 2, we first state our main result, without giving the main part of the technical proof, and show its immediate consequences. The main part of the proof will be given in Section 3. In the rest of this section, we give some elementary notions and notations used throughout the paper.

Our sets in this paper are over $\Sigma = \{0,1,\#\}$ unless otherwise specified. The symbol $\#$ is usually used as a delimiter among strings of $\{0,1\}^\ast$. A pairing function (resp., a k-tuple function) over $\{0,1\}^\ast$ is represented by separating two strings (resp., k strings) by this symbol. For a
string \( \omega \in \Sigma^* \), \(|\omega|\) denotes the length of \( \omega \). For a set \( L \subseteq \Sigma^* \), \( \overline{L} \) denotes the complement of \( L \). For a class \( K \) of sets, \( \text{co}-K \) denotes the class of sets whose complement is in \( K \). Let \( \Sigma^n \) (resp., \( \Sigma^{\leq n} \) and \( \Sigma^{< n} \)) denote the set of strings with length \( n \) (resp., at most \( n \) and less than \( n \)). For a finite set \( X \subseteq \Sigma^* \), \( ||X|| \) denotes the number of strings in \( X \). Let \( N \) and \( Z \), respectively, denote the set of natural numbers and the set of integers. We assume that all integers are expressed in binary notation (for negative integers, we assume a suitable representation system, such as 2’s-complement system).

For an oracle set \( X \), \( P(X) \) denotes the class of sets accepted by polynomial-time bounded deterministic oracle Turing machines (DOTM for short) with oracle \( X \) and \( \text{NP}(X) \) denotes the class of sets accepted by polynomial-time bounded nondeterministic oracle Turing machines (NOTM for short) with oracle set \( X \). Classes in the polynomial-time hierarchy [18] are denoted in the usual way: \( \Sigma^p_0 = \Pi^p_0 = \Delta^p_0 = \text{P} \), \( \Sigma^p_k = \text{NP}(\Sigma^p_{k-1}) \), \( \Pi^p_k = \text{co}-\Sigma^p_k \), \( \Delta^p_k = \text{P}(\Sigma^p_{k-1}) \), and \( \text{PH} = \bigcup_{k \geq 0} \Sigma^p_k \).

We are concerned with the following counting classes. \( \#P(X) \) [24] denotes the class of functions that give the number of accepting computation paths of polynomial-time bounded NOTM’s with oracle \( X \). \( PP(X) \) [10] (resp., \( \text{C_mP}(X) \) [26]) denotes the class of sets accepted by polynomial-time bounded nondeterministic many-one reducible to sets in \( \#P(X) \). For every string \( x \), \( x \in L \) iff \( F_1(x) > F_2(x) \) (resp., \( x \in L \) iff \( F_1(x) = F_2(x) \)). For an integer \( k \geq 2 \), we define \( \text{MOD}_kP(X) \) [4] as the class of sets \( L \) for which there exists a function \( F \in \#P(X) \) such that for all strings \( x \), \( x \in L \) iff \( F(x) \neq 0 \) (mod \( k \)). In particular, \( \text{MOD}_2P \) is usually denoted by \( \Theta \text{P} \) [15]. The unrelativized classes are defined by setting the oracle set to the empty set, and the specification of the oracle is omitted in this case.

2 Randomized reductions from \( \text{PH} \) to counting classes

In this section, we show that all sets in \( \text{PH} \) are polynomial-time randomized many-one reducible to sets in a wide range of counting classes, including \( \text{P}, \text{C}_m \text{P}, \) and \( \text{MOD}_k \text{P} \). What we will show is much stronger than this observation. For example, we will show that all sets in \( \text{PP}(\text{PH}) \) are polynomial-time randomized many-one reducible to sets in \( \text{PP} \). To state more precise statements, we first define a stronger variation of Schöning’s BP-operator [17], which formalizes the notion of polynomial-time randomized many-one reducibility with two-sided exponentially small error probability, and next define a family of counting classes to which our technique can be applied. Below, given a finite set \( X \) of strings and a predicate \( R \) over strings, we denote by \( Pr\{w \in X : R(w)\} \) the probability that \( R(w) \) is true for randomly chosen \( w \) from \( X \) under uniform distribution.

Definition 2.1 Let \( K \) be any class of sets. A set \( L \) is in \( \text{BP-K} \) if for every polynomial \( e \), there exist a set \( A \in K \) and a polynomial \( p \) such that for every string \( x \), \( Pr\{w \in \{0,1\}^{|x|} : x \in L \) iff \( x \# w \in A\} \geq 1 - 2^{-e(|x|)} \).

The essence of the following definition comes from the gap-definability notion of Fenner, Fortnow and Kurtz [8], and our notion below covers a wide range of counting classes while it defines a smaller family of their gap-definable classes.

Definition 2.2 For an oracle set \( X \), \( \text{GapP}(X) \) [8] is defined to be the class of functions \( F \) for which there exist two functions \( F_1, F_2 \in \#P(X) \) such that for all strings \( x, z \in L \) iff \( F_1(x) = F_2(z) \). For an integer \( k \geq 2 \), we define \( \text{MOD}_k \text{P}(X) \) as the class of sets \( L \) for which there exists a function \( F \in \#P(X) \) such that for all strings \( x, z \in L \) iff \( F(x) \neq 0 \) (mod \( k \)). In particular, \( \text{MOD}_2 \text{P} \) is usually denoted by \( \Theta \text{P} \) [15]. The unrelativized classes are defined by setting the oracle set to the empty set, and the specification of the oracle is omitted in this case.

Example. All the well-known counting classes can be defined in the way of Definition 2.2. For instance, \( \text{P} \) and \( \text{PP}(X) \), for any oracle set \( X \), can be defined as \( \text{C} \left[ Z^+, \text{GapP} \right] \) and \( \text{C} \left[ Z^+, \text{GapP}(X) \right] \) respectively, where \( Z^+ \) denotes the set of positive integers.
Now we show our main result. The main result essentially follows from the following technical lemma, whose proof will be given in the next section.

**Lemma 2.3** Let $F$ be any function in $\text{GapP}(\text{PH})$ and let $e$ be any polynomial. Then there exist a function $H \in \text{GapP}$ and a polynomial $s$ such that for every string $x$,

$$\Pr\{w \in \{0,1\}^{|x|} : H(x#w) = F(x)\} \geq 1 - 2^{-e(|x|)}.$$

Our main theorem below subsumes disparate observations that several counting classes are at least as computationally hard as $\text{PH}$.

**Theorem 2.4** Let $Q$ be a subset of $\mathbb{Z}$. Then, $\text{C}[Q, \text{GapP}(\text{PH})] \subseteq \text{BP} \cdot \text{C}[Q, \text{GapP}]$.

**Proof.** Let $F$ be a function in $\text{GapP}(\text{PH})$ witnessing that a set $L$ is in $\text{C}[Q, \text{GapP}(\text{PH})]$. From Lemma 2.3, there exists a function $H \in \text{GapP}$ that satisfies the condition in the lemma. Define $A = \{ x#w | H(x#w) \in Q \}$. Obviously, $A$ is in $\text{C}[Q, \text{GapP}]$, and $L$ and $A$ satisfy the condition in Definition 2.1. Thus we have $L \in \text{BP} \cdot \text{C}[Q, \text{GapP}]$.

In the rest of this section, we apply the theorem to some well-known counting classes and obtain some new relationships between those counting classes and $\text{PH}$. Also, we will show some other interesting consequences of the theorem and the corollary below.

**Corollary 2.5**

(1) $\text{PP}(\text{PH}) \subseteq \text{BP} \cdot \text{PP}$.

(2) $\text{C}_{=\text{P}}(\text{PH}) \subseteq \text{BP} \cdot \text{C}_{=\text{P}}$ and $\text{co-C}_{=\text{P}}(\text{PH}) \subseteq \text{BP} \cdot \text{co-C}_{=\text{P}}$.

(3) For all integers $k \geq 2$,

$$\text{MOD}_k \text{P}(\text{PH}) \subseteq \text{BP} \cdot \text{MOD}_k \text{P} \quad \text{and} \quad \text{co-MOD}_k \text{P}(\text{PH}) \subseteq \text{BP} \cdot \text{co-MOD}_k \text{P}.$$

(4) Thus, for each $K$ of $\text{PP}$, $\text{C}_{=\text{P}}$, $\text{co-C}_{=\text{P}}$, $\text{MOD}_k \text{P}$, and $\text{co-MOD}_k \text{P}$, we have $\text{PH} \subseteq \text{BP} \cdot K$.

**Proof.** As in the previous example, we can easily see that all the counting classes above can be defined in the way of Definition 2.2. We omit the detail here; the interesting reader may refer to [8] for some technical points.

**Remark.** For a class $K$ of sets that is closed downward under the polynomial-time majority reducibility, it follows from the probability amplification lemma of Schöning [17] that $\text{BP} \cdot K = \text{BP} \cdot K$, where, for two sets $A$ and $B$, $A$ is polynomial-time majority reducible to $B$ if there exists a polynomial-time computable function $g$ such that for all $x$, $g(x) = y_1# \cdots #y_m$ ($m \geq 1$) and $x \in A$ if the majorities of $y_i$'s are in $B$. However, if $K$ is not closed under the reducibility, then the two classes might be different. It is currently known that $\text{PP}$ (resp., $\text{C}_{=\text{P}}$ and $\text{MOD}_k \text{P}$ with $k$ prime) are closed downward under the polynomial-time majority reducibility [4][6][11] (see [9] also). Thus, for such classes, we may replace the $\text{BP}$-operators in the above corollary by the $\text{BP}$-operators, without weakening the results. Nonetheless Theorem 2.4 and the result (3) above for the unrestricted case might be weakened when the $\text{BP}$-operators are replaced by the $\text{BP}$-operator.

Combining Corollary 2.5 with the following result due to Schöning [17], we can observe that $\text{C}_{=\text{P}}$ and $\text{MOD}_k \text{P}$, as well as $\text{PP}$ and $\text{PH}$, are computationally harder than $\text{PH}$ unless $\text{PH}$ collapses to a finite level.

**Theorem 2.6** [17] For every $k \geq 1$, if $\Pi_k^P \subseteq \text{BP} \cdot \Sigma_k^P$, then $\text{PH} = \Sigma_{k+1}^P$.

**Corollary 2.7** Let $Q$ be a subset of $\mathbb{Z}$ such that $Q \neq \emptyset$ and $Q \neq \mathbb{Z}$. Then, if $\Pi^P \subseteq \text{C}[Q, \text{GapP}]$, then $\text{PH}$ collapses to a finite level. Thus, for each $K$ of $\text{C}_{=\text{P}}$ and $\text{MOD}_k \text{P}$, $K \nsubseteq \text{PH}$ unless $\text{PH}$ collapses to a finite level.

**Proof.** We first show that $\text{C}[Q, \text{GapP}]$ has a polynomial-time many-one complete set. Let $\# \text{SAT}$ be the function that, given a Boolean formula, gives the number of satisfying assignments of the formula. It was observed by Valiant [24] that $\# \text{SAT} \in \# \text{P}$ and for all functions $F$ in $\# \text{P}$, there exists a polynomial-time computable function $f$ such that for all strings $x$, $F(x) = \# \text{SAT}(f(x))$. Now define $\text{DiS} \text{AT} \subseteq \{ \phi_1, \phi_2 \}$
Then we easily see that $\text{DiSAT}_Q$ is polynomial-time many-one complete for $C[Q,\text{GapP}]$ via the function $f$. Since $Q \neq \emptyset$ and $Q \neq \mathbb{Z}$, it is obvious that $\text{PH} \subseteq C[Q,\text{GapP}(PH)]$. The first statement of this corollary follows immediately from these facts, Theorem 2.4, and Theorem 2.6. The second is immediate from the first.

Below, we show some other consequences of Theorem 2.4 and Corollary 2.5. First we show that for each $K$ of $\text{PP}$, $C_P$ and $\text{MOD}_4P$, $K(\text{PH})$ and $K$ itself are interreducible to each other under the polynomial-time randomized reducibility. Next we observe that, when we consider nonuniform version of the classes in the sense of Karp and Lipton [12] and random analogues of those classes in the sense of Bennet and Gill [5], the inclusions in Corollary 2.5 become equalities. To show these results, we first prove some technical lemmas.

**Lemma 2.8** Let $K$ be any class of sets that is closed downward under the polynomial-time many-one reducibility. Then, $\overline{\overline{B}}^P \cdot \overline{B}^P \cdot K = \overline{B}^P \cdot K$.

**Proof.** Using a technique as in [17][21], we can easily show the lemma. We left the detail to the reader.

Given a class $K$ of sets, we denote by $\leq_m^P \cdot K$ the class of sets that are polynomial-time many-one reducible to sets in $K$. Note that for any class $K$, $\leq_m^P \cdot \leq_m^P \cdot K = \leq_m^P \cdot K$, i.e., $\leq_m^P \cdot K$ is closed downward under the polynomial-time many-one reducibility, because of the transitivity of the reducibility.

**Corollary 2.9** Let $Q$ be a subset of $\mathbb{Z}$. Then, $\overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}(PH)] = \overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}]$. $\overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}] \subseteq \overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}]$. Thus, we have, from Theorem 2.4 and Lemma 2.8, that $\overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}(PH)] \subseteq \overline{B}^P \cdot \leq_m^P \cdot \overline{B}^P \cdot \overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}] \subseteq \overline{B}^P \cdot \overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}] = \overline{B}^P \cdot \leq_m^P \cdot C[Q,\text{GapP}]$. The converse inclusion is obvious.

The following corollary is immediate from the above one.

**Corollary 2.10**

1. $\overline{B}^P \cdot \overline{B}^P (PH) = \overline{B}^P \cdot \overline{B}^P$.
2. $\overline{B}^P \cdot C_m^P (PH) = \overline{B}^P \cdot C_m^P$ and $\overline{B}^P \cdot \text{co-C}_m^P (PH) = \overline{B}^P \cdot \text{co-C}_m^P$.

3. For all integers $k \geq 2$, $\overline{B}^P \cdot \text{MOD}_k^P (PH) = \overline{B}^P \cdot \text{MOD}_k^P$ and $\overline{B}^P \cdot \text{co-MOD}_k^P (PH) = \overline{B}^P \cdot \text{co-MOD}_k^P$.

**Definition 2.11** [5][12] Let $K$ be any class of sets. A set $L$ is in $K/\text{poly}$ if there exist a set $A \in K$, an advice function $f$ from natural numbers to strings, and a polynomial $p$ such that $|f(n)| \leq p(n)$ for all $n$, and $L = \{ x : x \# f(|x|) \in A \}$. Let $K$ be a relativizable complexity class. A set $L$ is in almost-$K$ if for almost all oracle sets $X$, $L$ is in $K(X)$.

Recently, it is well understood that for any class $K$ of sets, if the class is closed downward under the polynomial-time majority reducibility, then $\text{BP} \cdot K \subseteq K/\text{poly}$ and $\text{BP} \cdot K \subseteq \text{almost-K}$; for the latter inclusion, we need some more assumptions on the class $K$ such as it being relativizable class and containing at most countably infinite sets. We can apply this understanding to the $\overline{B}P$-complexity classes, because in the previous works such as [17], the closure property of the class under the polynomial-time majority reducibility has been used only for amplifying the success probability in the $\overline{B}P$-operator; on the other hand, we have already had the amplified success probability in the $\overline{B}P$-operator.

**Lemma 2.12** For any class $K$, $\overline{B}P \cdot K \subseteq K/\text{poly}$.

For characterizing the $\overline{B}P$-complexity classes by means of random tally oracles, Tang and Watanabe [20] showed the following result. (Note that we mention their results by using our $\overline{B}P$-operator, though they showed it with the $\overline{B}P$-operator.)

**Theorem 2.13** [20] If any given class $K$ of sets contains at most countably many sets and
is closed under the polynomial-time majority reducibility, then a set \( L \) is in \( \text{BP-K} \) if and only if for almost all tally sets \( T, L \in \text{KoPF}(T) \), where \( \text{KoPF}(T) \) is the class of sets \( A \) such that for some \( B \in \text{K} \) and a function \( f \in \text{PF}(T) \), \( A = \{ x : f(x) \in B \} \).

In the proof of "only if" part of this theorem, we do not need to deal only with tally sets; we may consider any oracle sets. (In the "if" part, we have to be concerned only with tally sets when following their proof technique; but, we are not interested in the "if" part here.) Thus we can obtain the following lemma by using their technique.

**Lemma 2.14** Let \( \text{KoPF}(T) \) be any class that contains at most countably many sets. Let \( L \) be any set. Then, if \( L \in \text{BP-K} \), then for almost all sets \( X, L \in \text{KoPF}(X) \).

It is easy to see that for every oracle set \( X \) and every subset \( Q \) of \( \mathbb{Z} \), \( C[Q, \text{GapP}(X)] \subseteq C[Q, \text{GapP}(X)] \). Thus, from Lemma 2.12, Lemma 2.14, and Corollary 2.5, we obtain the following:

**Corollary 2.15** For all subsets \( Q \) of \( \mathbb{Z}, C[Q, \text{GapP}(PH)] \subseteq C[Q, \text{GapP}(PH)] \), and for all almost sets \( X, C[Q, \text{GapP}(PH)] \subseteq C[Q, \text{GapP}(PH)] \). Thus, for each \( \text{KoPF}(T) \), \( \text{KoPF}(T) \), and \( \text{co-MOD}_{kP} \),

(1) \( \text{KoPF}(T) \subseteq \text{KoPF}(PH) \cap \text{KoPF}(T) \), and

(2) \( \text{KoPF}(T) = \text{KoPF}(PH) \cap \text{KoPF}(T) \).

We would like to close this section by defining a natural extension of \( \text{MOD}_{kP} \) and by observing a result for the class similar to Corollary 2.5(4).

**Definition 2.16** Let \( g \) be a function from strings to natural numbers. We define \( \text{MOD}_{gP} \) to be the class of sets \( L \) for which there exists a function \( F \in \text{#P(X)} \) such that for all \( x, L \in \text{#P(X)} \) if and only if \( F(x) = 0 \) (mod \( g(x) \)).

**Corollary 2.17** For all polynomial-time computable functions \( g \) from strings to natural numbers such that for all \( x, g(x) \geq 2 \), \( \text{PH} \subseteq \text{MOD}_{gP} \).

**Proof.** For all sets \( L \in \text{PH} \), there exists a function \( F \in \text{GapP}(PH) \) such that for all \( x, F(x) = 1 \); otherwise, \( F(x) = 0 \). From this fact and Lemma 2.3, we have the corollary.

We note that \( \text{MOD}_{gP} \) cannot in general be defined in the way of Definition 2.2. Thus, at the present time, we do not know whether for all \( g, \text{MOD}_{gP}(PH) \subseteq \text{BP-MOD}_{gP} \), though we feel that it may be the case. In the final section, we will discuss the reason why our proof technique cannot be applied to the class.

## 3 Proof of Lemma 2.3

In this section, we prove Lemma 2.3. The lemma is obtained from the following lemma.

**Lemma 3.1** Let \( X \) be an oracle set, let \( F \) be a function in \( \text{#P(NP(X))} \), and let \( e \) be a polynomial. Then there exist a function \( H \in \text{GapP}(X) \) and a polynomial \( s \) such that for all strings \( x, \text{Pr}(w \in \{0,1\}^{l(x)} : H(x\#w) = F(x)) \geq 1 - 2^{-e(l(x))} \).

By applying this lemma inductively to each class in \( \text{PH} \), we immediately obtain Lemma 2.3. The detail is left to the reader. Thus, in this section, we concentrate to prove the above lemma.

**Proof of Lemma 3.1.** In this proof, we will use some more technical lemmas. For the sake of clarifying our main heart of this proof, the proofs of those lemmas will be given after completing this proof.

We first mention the following result on \( \text{#P(NP(X))} \) that allows us to change the definition of the class.

**Lemma 3.2** For every function \( F \in \text{#P(NP(X))} \), there exist a set \( A \in \text{co-NP(X)} \) and a polynomial \( p \) such that for all \( x, F(x) = ||\{y \in \{0,1\}^{l(x)} : x\#y \in A\}|| \).

In the remainder of this proof, when we write \( x\#y \), we assume \( |y| = p(|x|) \), for simplifying the argument. From the well-known characterization of \( \text{co-NP(X)} \) in [18][27], we have, for the set \( A \) above, a set \( B \in \text{P(X)} \) and a polynomial \( q \) such
that for all $x\#y, z\#y \in A$ if there is no string $z \in \{0,1\}^{|x\#y|}$ such that $z\#y\#z \in B$. By using the set $B$, we will later construct a function $H_1$ in $\text{GapP}(X)$ and a polynomial $s$ such that for all positive integers $n$,

$$(A) \quad \Pr\{w \in \{0,1\}^{|w|} :$$

- for all $x\#y \in A$ of length $n$, $H_1(x\#y\#w) = 1$,
- for all $x\#y \notin A$ of length $n$, $H_1(x\#y\#w) = 0 \} \geq 1 - 2^{-\epsilon(n)}$.

By using $H_1$, we define the required function $H$ as follows:

$$H(x\#w) = \sum_{y \in \{0,1\}^{|y|}} H_1(x\#y\#w).$$

Then we easily see that $H$ is in $\text{GapP}(X)$ (provided $H_1 \in \text{GapP}(X)$) and that $H$ and the polynomial $s$ above satisfy the condition of Lemma 3.1.

Now we show how to define the function $H_1$ and the polynomial $s$ above. To this end, we use a consequence of Valiant and Vazirani's result [25]. Following their paper, we shall view a string $w \in \{0,1\}^m$ as a vector in $GF[2]^m$. We denote by $u \cdot w$ the inner product of the vectors $u$ and $w$ in $GF[2]^m$. For a string $x\#y$ of length $n$ and a finite number of strings $w_1, \ldots, w_k \in \{0,1\}^{|y|}$, we define a finite set $B_{x\#y}$ and $B_{x\#y}(w_1, \ldots, w_k)$ by

$$B_{x\#y} = \{ z \in \{0,1\}^{|y|} : x\#y\#z \in B \},$$

$$(B) \quad B_{x\#y}(w_1, \ldots, w_k) = \{ z \in \{0,1\}^{|y|} : x\#y\#z \in B \text{ and } w_1 \cdot z = \cdots = w_k \cdot z = 0 \}.$$

 Furthermore, we use the following notation. Let $l, m$ be any positive integers. We denote by $\text{Mat}[l, m]$ the set of all $l \times m$ matrices whose components are strings in $\{0,1\}^m$. For any matrix $W \in \text{Mat}[l, m]$, we denote by $W_{j,k}$ the $(j,k)$-component of $W$. Below, we shall view a matrix $W \in \text{Mat}[l, m]$ as the string

$$W_{1,1} \cdots W_{1,m} W_{2,1} \cdots W_{2,m} \cdots W_{l,1} \cdots W_{l,m}$$

which is in $\{0,1\}^{l \cdot m^2}$. Then we have the following lemma (from Valiant and Vazirani's result).

**Lemma 3.3** Let $e$ be a polynomial. Then there exists a polynomial $r$ such that for all strings $x\#y$ of length $n$,

1. if $x\#y \in A$, then
   $$\Pr\{W : \|B_{x\#y}\| = 0 \text{ and } (\forall k,j,1 \leq k \leq r(n), 1 \leq j \leq q(n))$$
   $$[\|B_{x\#y}(W_{k,1}, \ldots, W_{k,j})\| = 0] \} = 1,$$

2. if $x\#y \notin A$, then
   $$\Pr\{W : \|B_{x\#y}\| = 1 \text{ or } (\exists k,j,1 \leq k \leq r(n), 1 \leq j \leq q(n))$$
   $$[\|B_{x\#y}(W_{k,1}, \ldots, W_{k,j})\| = 1] \} \geq 1 - 2^{-\epsilon(n)},$$

where $W$ is randomly chosen from $\text{Mat}[r(n), q(n)]$ under uniform distribution.

Since we can take an arbitrary polynomial $e$ in the above lemma, we obtain a stronger observation than the above one: for all positive integers $n$,

(B) $\Pr\{ W : \text{for all strings } x\#y \text{ of length } n,$

- $x\#y \in A \Rightarrow \|B_{x\#y}\| = 0$ and
  $$(\forall k,j,1 \leq k \leq r(n), 1 \leq j \leq q(n))$$
  $$[\|B_{x\#y}(W_{k,1}, \ldots, W_{k,j})\| = 0],$$

- $x\#y \notin A \Rightarrow \|B_{x\#y}\| = 1$ or
  $$(\exists k,j,1 \leq k \leq r(n), 1 \leq j \leq q(n))$$
  $$[\|B_{x\#y}(W_{k,1}, \ldots, W_{k,j})\| = 1] \} \geq 1 - 2^{-\epsilon(n)},$$

where $W$ is randomly chosen from $\text{Mat}[r(n), q(n)]$ under uniform distribution.

Now we define the function $H_1$ as follows:

- for all strings $x\#y$ of length $n$ and all $W \in \text{Mat}[r(n), q(n)]$,
  $$H_1(x\#y\#W) = \{ (G(x\#y\#\lambda) - 1) \cdot \prod (G(x\#y\#W_{k,1} \cdots W_{k,j}) - 1) \}^2$$

(for other strings $W$, we define the value of $H_1$ to be zero), where the product is taken over all $j,k$ with $1 \leq k \leq r(n)$ and $1 \leq j \leq q(n)$, and $G$ is defined as follows (which gives
Furthermore, we define the required polynomial $s$ by

$$s(n) = r(n) \cdot q^2(n).$$

Then we show that $H_1$ and $s$ satisfy (A) mentioned previously. (In what follows, recall that we are viewing a matrix $W \in \text{Mat}[r(n), q(n)]$ as a string in $\{0, 1\}^{s(n)}$.)

Let $z#y$ be a string of length $n$ and let $W \in \text{Mat}[r(n), q(n)]$. If $z#y \in A$, then $H_1(z#y#W) = 0$ and for all $1 \leq k \leq r(n)$ and all $1 \leq j < q(n)$,

$$||B_{z#y}(W_{k,1}, \ldots, W_{k,j})|| = 0.$$  

Hence, in this case, we have $H_1(z#y#W) = 1$. Otherwise, $||B_{z#y}(W_{k,1}, \ldots, W_{k,j})|| = 1$ for some $k, j$. Hence, in this case, we have $H_1(z#y#W) = 0$. From these observations, we see that the condition in (B) implies the condition in (A), and hence see that the probability in (A) is greater than or equal to the probability in (B). Thus $H_1$ and $s$ satisfy (A).

Obviously, $G$ is in $\#P(X)$. Then, the following lemma tells us the membership of $H_1$ in GapP($X$). The lemma draws out an essential idea developed independently by Gundermann, Nasser and Wechsung [11] and by Ogiwara [14]. Below, we shall view a finite multiset of strings as a list of strings in the multiset and consider a list of strings as an element of the set $\Sigma^*(\#\Sigma^*)^*$. Hence we shall view $\Sigma^*(\#\Sigma^*)^*$ as the class of all finite multisets. We also use the ordinary set-theoretical notations for multisets.

**Lemma 3.4** [11][14] Let $F, G$ be functions in $\#P(X)$ and let $f : \Sigma^* \to \Sigma^*(\#\Sigma^*)^*$ be a polynomial-time computable function. Then, the function

$$H(x) = \prod_{y \in f(x)} (F(y) - G(y))$$

is in GapP($X$).

This completes the proof of Lemma 3.1, remaining to prove the above technical lemmas.

(End-of-Lemma 3.1)

In the rest of this section, we prove the technical lemmas used above. In those proofs, we will use the notations defined in the proof of Lemma 3.1.

**Proof of Lemma 3.2** Let $M$ and $C$ be a polynomial-time bounded NOTM and an oracle set from NP($X$), respectively, that witness a function $F$ being in $\#P(NP(X))$. Let $N$ be a polynomial-time bounded NOTM that accepts $C$ relative to $X$. We below assume, without loss of generality, that all possible computation paths of $M$ together with possible oracle answers and those paths of $N_x$ are encoded into binary strings in a usual manner. Let $w$ be a computation path of $M$ on a given input $x$, which includes possible oracle answers. Then we denote by $YES_M(x, w)$ (resp., $NO_M(x, w)$) the set of query strings that are made by $M$ along path $w$ and whose corresponding oracle answer in $w$ is “yes” (resp., “no”). Now we define a set $A$ as follows. A string $x#y#y_1#\cdots#y_m$ is in $A$ if it satisfies the following conditions:

- $w$ is a computation path of $M$ on input $x$,
- $m \geq 0$,
- $YES_M(x, w)$ contains $m$ strings, say $z_1, \ldots, z_m$,
- each $y_i$ is the lexicographically smallest accepting computation path of $N_x$ on input $z_i$, and
- $NO_M(x, w) \subseteq \overline{C}$.

By a standard padding argument, we can so easily adjust the definition of $A$ (without changing its complexity) that for some polynomial $p$ and any strings $x$ and $u$, if $x#u \in A$, then $|u| = p(|x|)$. Then, we see that $A$ is in co-NP($X$) and for every string $x$,

$$F(x) = \|\{u \in \Sigma^{p(|x|)} : x#u \in A\}\|.$$  

(End-of-Lemma 3.2)
Proof of Lemma 3.3. Let $A$ and $B$ be the same sets as in Lemma 3.1. In their paper [25], Valiant and Vazirani showed the following claim. (Note that in the claim, we modify their original result by using our present notations).

Claim 1 [25] For all strings $x#y$ of length $n$,

1. if $x#y \in A$, then
$$\Pr(w_1, \ldots, w_q(n)) = \left\lVert B_x \otimes y(w_1, \ldots, w_k) \right\rVert = 0 \right\rVert = 1,$$
and
2. if $x#y \notin A$, then
$$\Pr(w_1, \ldots, w_q(n)) = \left\lVert B_x \otimes y(W_1, \ldots, W_k) \right\rVert = 1 \right\rVert = 1/4,$$
where each $w_i$ is randomly chosen from $\{0,1\}^{q(n)}$ under uniform distribution. (Recall that $q$ is the polynomial mentioned in Lemma 3.1.)

(End-of-Claim 1)

Repeating the random process in Claim 1 above and taking the disjunction of the outcomes, we can amplify the probability in 2 without changing the probability in 1. To state this more precisely, if we repeat the random process $r(n)$ times where $r$ is an arbitrary polynomial, then we have the following:

1'. if $x#y \in A$, then
$$\Pr\{W : ||B_x \otimes y|| = 0 \text{ and } (\forall k, j, 1 \leq k \leq r(n), 1 \leq j \leq q(n)) \left\lVert B_x \otimes y(W_k, \ldots, W_j) \right\rVert = 0 \right\rVert = 1,$$
and
2'. if $x#y \notin A$, then
$$\Pr\{W : ||B_x \otimes y|| = 1 \text{ or } (\exists k, j, 1 \leq k \leq r(n), 1 \leq j \leq q(n)) \left\lVert B_x \otimes y(W_k, \ldots, W_j) \right\rVert = 1 \right\rVert \geq 1 - (3/4)^{r(n)},$$
where $W$ is randomly chosen from Mat[r(n), q(n)] under uniform distribution. Thus, when we take the polynomial $r$ such that $(3/4)^{r(n)} \leq 2^{-e(n)}$ for all $n > 0$, we have this lemma.

(End-of-Lemma 3.3)

Proof of Lemma 3.4. Taking the expansion of $H(x)$, we can express $H(x)$ by

$$H(x) = \sum_{S \subseteq f(x)} \left( \prod_{y \in S} F(y) \right) \left( -1 \right)^{||f(x) \setminus S||} \left( \prod_{y \in f(x) \setminus S} G(y) \right),$$
where we define $\prod_{y \in \emptyset} F(y) = \prod_{y \in \emptyset} G(y) = 1$. For each additive term of the above expression, if $||f(x) \setminus S||$ is odd, then the term is negative; otherwise, it is positive. So, by separating both cases from each other, we can express $H(x)$ by $H(x)$

It is easy to see that each summation in the last expression is realized as a #P(X) function. Thus $H$ is in GapP(X).

(End-of-Lemma 3.4)

4 Concluding remarks

In this paper, we showed that all the known counting classes are at least as hard as the polynomial-time hierarchy; that is, all sets in the polynomial-time hierarchy is randomly reducible to sets in the counting classes with exponentially small error probability. A crucial point in showing these results is that every counting class concerned in this paper can be characterized in terms of a simple predicate over #P functions (alternatively, GapP functions). In fact, we showed, as a more general result, that the same relationship as above holds for all counting classes that can be defined in the way of Definition 2.2 by using GapP functions.

Nonetheless one can obtain a broader family of counting classes, as Fenner, Fortnow and Kurtz have defined the gap-definable classes in [8]. As an immediate question related to this work, one may ask whether our main result remains true for all of the gap-definable classes. Unfortunately we have not been able to settle the question. But, in our intuition, it seems unlikely to be the case. As a more concrete example to our intuition, we here consider SPP [8] that is the class of sets $L$ for which there exists a function in GapP such that for every $x$, $x \in L$ iff $F(x) = 1$ and $x \notin L$ iff $F(x) = 0$. In the definition of SPP, there exists a promise, as well as in the case of UP [23] and Few [7], that the GapP function never takes the value other than 0 or 1. Because of the existence
of such a promise, we conjecture that SPP is not as computationally hard as the polynomial-time hierarchy (at least in the sense of the present work). In [8], SPP has been shown to be a gap-definable class. Thus we think that our main result cannot be extended to the gap-definable classes.

As mentioned at the end of Section 2, MOD,P is out of our family and is a typical example to which our proof technique could not be applied. In order to establish the relationship MOD,P(PH) ⊆ BP-MOD,P, we must probably show, for all L in MOD,P(PH), the existence of H₁ in GapP and a polynomial s such that for all x, x ∈ L iff for (intuitively) almost all w of \{0, 1\}^{|x|}, H₁(x¬w) ≠ 0 (mod g(x¬w)). By using our argument, we can find a function H₂ ∈ GapP such that for all x, x ∈ L iff for almost all w of \{0, 1\}^{|x|}, H₂(x¬w) ≠ 0 (mod g(x)), but in general, we cannot know how g(z) is related to g(x¬w) as well as how to construct the function H₁ from H₂. This is why we could not establish the relationship, while we conjecture it is the case.

We think it is still important to find much closer relationships between counting classes and classes in PH. As mentioned in Section 1, it seems very hard to show a new inclusion relationship. Thus our interest mainly concentrates on some reducibility questions. In particular, it is more important to know whether all sets in PH (or a class in PH) are polynomial-time reducible to sets in C=P or MOD₂P under deterministic reducibilities such as Turing or truth-table ones. Considering the current status on this question, we think it still nice to show that all sets in Σ₂P are polynomial-time Turing reducible to sets in C=P (or in MOD₂P), if it holds, or to find oracles separating those classes.

Acknowledgement This work was done while the authors were visiting the Department of Mathematics at the University of California at Santa Barbara. We would like to be very thankful to Prof. Ronald V. Book for his advice and hospitality during that time. We would like to thank so much to Richard Beigel, Lance Fortnow, and Jun Tarui for their valuable suggestions and discussions, and would also like to thank to the unknown referees for their suggestions for improving our presentation.

References


