Finding Errorless Pessiland in Error-Prone Heuristica

Shuichi Hirahara
National Institute of Informatics, Tokyo, Japan

Mikito Nanashima
Tokyo Institute of Technology, Japan

Abstract

Average-case complexity has two standard formulations, i.e., *errorless* complexity and *error-prone* complexity. In average-case complexity, a critical topic of research is to show the equivalence between these formulations, especially on the average-case complexity of NP.

In this study, we present a relativization barrier for such an equivalence. Specifically, we construct an oracle relative to which NP is easy on average in the error-prone setting (i.e., $\mathsf{DistNP} \subseteq \mathsf{HeurP}$) but hard on average in the errorless setting even by $2^{o(n/\log n)}$-size circuits (i.e., $\mathsf{DistNP} \not\subseteq \mathsf{AvgSIZE}[2^{o(n/\log n)}]$), which provides an answer to the open question posed by Impagliazzo (CCC 2011). Additionally, we show the following in the same relativized world:

Lower bound of meta-complexity $\mathsf{GapMINKT}^{\mathcal{O}} \notin \mathsf{pr\text{-}SIZE}^{\mathcal{O}}[2^{o(n/\log n)}]$ and $\mathsf{GapMCSP}^{\mathcal{O}} \notin \mathsf{pr\text{-}SIZE}^{\mathcal{O}}[2^{n^{\epsilon}}]$ for some $\epsilon > 0$.

Worst-case hardness of learning on uniform distributions $\mathsf{P/poly}$ is not weakly PAC learnable with membership queries on the uniform distribution by nonuniform $2^n/n^{\omega(1)}$-time algorithms.

Average-case hardness of distribution-free learning $\mathsf{P/poly}$ is not weakly PAC learnable on average by nonuniform $2^{o(n/\log n)}$-time algorithms.

Weak cryptographic primitives There exist a hitting set generator, an auxiliary-input one-way function, an auxiliary-input pseudorandom generator, and an auxiliary-input pseudorandom function against $\mathsf{SIZE}^{\mathcal{O}}[2^{o(n/\log n)}]$.

This provides considerable insights into Pessiland (i.e., the world in which no one-way function exists, and NP is hard on average), such as the relativized separation of the error-prone average-case hardness of NP and auxiliary-input cryptography. At the core of our oracle construction is a new notion of random restriction with masks.

2012 ACM Subject Classification Theory of computation → Computational complexity and cryptography

Keywords and phrases average-case complexity, oracle separation, relativization barrier, meta-complexity, learning, auxiliary-input cryptography

Digital Object Identifier 10.4230/LIPIcs.CCC.2022.25

Funding *Shuichi Hirahara*: JST, PRESTO Grant Number JPMJPR2024, Japan.
Mikito Nanashima: JST, ACT-X Grant Number JPMJAX190M, Japan.

Acknowledgements The authors would like to thank the anonymous reviewers for many helpful comments.

1 Introduction

Average-case complexity has been studied extensively in computational complexity theory. In the theory of average-case complexity, the computational cost of solving a distributional problem (L, D) well on average is investigated, where L is a language, and D is a polynomial-time samplable distribution on instances. Average-case complexity depends on the definition of "average-case easiness," and there are at least two natural ways to formulate this: *errorless*

37th Computational Complexity Conference (CCC 2022).
Editor: Shachar Lovett; Article No. 25; pp. 25:1–25:28
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

and *error-prone*[1] average-case easiness. In both formulations, an efficient algorithm needs to output a correct answer with high probability over a choice of random instances sampled from distribution D. The difference is in the requirement when the algorithm cannot solve an instance. In the errorless setting, the algorithm is not allowed to output a wrong answer; instead, it is allowed to output a special symbol $\perp$, which represents the failure of the algorithm. In the error-prone setting, an algorithm is allowed to output a wrong answer, provided that the error probability of the algorithm is small.

The difference between the two notions originates from two different motivations of studying average-case complexity. On one hand, Levin [29] laid the foundation of the theory of average-case complexity of NP and introduced the notion of *average-case polynomial-time*, which is equivalent to errorless heuristic schemes [24, 7]. The motivation of Levin is to clarify which distributional NP problems are hard, as some NP-complete problems are indeed easy on average with respect to natural distributions. Levin proved the distributional NP-completeness of a problem called the tiling problem. Although Levin's theory is applicable to both of the average-case notions, it is more natural to consider the notion of errorless average-case easiness in this context: Practical heuristic algorithms, such as SAT solvers, can be considered as errorless heuristics. A SAT solver is usually guaranteed to output the correct answer if it halts, but the solver may "fail" on some instances, i.e., may require a long time to halt on some instances. Levin's theory demonstrates that some distributional NP problems are hard and are unlikely to be solved by such heuristic algorithms. On the other hand, the errorless notion is not (necessarily) appropriate for discussing the security of cryptographic primitives. The foundational work of Blum and Micali [6] and Yao [44] demonstrated that error-prone average-case hardness of some distributional NP problems is useful to build cryptographic primitives. Closing the gap between the errorless and error-prone average-case notions would unify the two motivations of studying average-case complexity. In his influential paper, Impagliazzo [24] explicitly raised this question as an important research direction. The question can be formally stated as follows.

▶ **Question 1.** *Is* $\mathsf{DistNP} \subseteq \mathsf{HeurP}$ *equivalent to* $\mathsf{DistNP} \subseteq \mathsf{AvgP}$*?*

Here, AvgP (resp. HeurP) denotes the class of distributional problems solvable on average by a polynomial-time algorithm in the errorless (resp. error-prone) setting; see Section 3.1 for a formal definition. DistNP denotes the class of distributional NP problems, i.e., $\mathsf{DistNP} = \{(L, D) : L \in \mathsf{NP} \text{ and } D \text{ is a polynomial-time samplable distribution}\}$.

Giving an affirmative answer to Question 1 is necessary for basing the security of cryptography on the worst-case hardness of NP. An additional motivation was recently provided by Hirahara and Santhanam [22]: they identified a deep connection between the question of errorless versus error-prone average-case complexities and the question of constructing an instance checker for NP, which is another long-standing and important open question raised in the seminal work of Blum and Kannan [5].

Despite its importance, there does not seem to be an effective method for addressing this question, so it is natural to ask whether there is a technical barrier. This meta-approach is often considered in computational complexity theory and is useful for excluding hopeless proof techniques from consideration. For example, proof techniques that are captured by standard frameworks, such as relativization [4], natural proofs [38], and algebrization [1], are known to be incapable of resolving the P versus NP question. However, to the best of our knowledge,

[1] It is originally called the "heuristic" complexity, and the term "error-prone" is due to the follow-up work [22].

there is no barrier for the question of errorless versus error-prone average-case complexities. In fact, Impagliazzo [24, 25] raised the open question of presenting a relativization barrier to Question 1.

▶ **Question 2.** *Is there an oracle $\mathcal{O}$ such that $\mathsf{DistNP}^{\mathcal{O}} \not\subseteq \mathsf{AvgP}^{\mathcal{O}}$ and $\mathsf{DistNP}^{\mathcal{O}} \subseteq \mathsf{HeurP}^{\mathcal{O}}$?*

The main contribution of this study is to resolve this decade-old open question affirmatively. Before presenting the details of our results, we review the recent progress in complexity theory that demonstrates the notable power of the errorless average-case easiness of NP by *relativizing* proof techniques. Along the way, we provide additional questions related to errorless versus error-prone average-case complexities. We refer to the possible world in which $\mathsf{DistNP} \subseteq \mathsf{AvgP}$ (resp. $\mathsf{DistNP} \subseteq \mathsf{HeurP}$) but $\mathsf{P} \neq \mathsf{NP}$ as *errorless Heuristica* (resp. *error-prone Heuristica*). In any relativized errorless Heuristica, the following computational tasks regarding worst-case complexity are proved to be feasible.

Errorless Heuristica I: Approximating Complexity (Meta-Complexity)

Meta-complexity is a field that studies the computational complexity of determining computational complexity. One central meta-computational problem is MINKT; for an input $(x, t) \in \{0,1\}^n \times \mathbb{N}$, MINKT is the problem of determining the minimum description length of the program that prints x in t time, i.e., the t-time-bounded Kolmogorov complexity of x. Another well-studied problem is MCSP; for an input $x \in \{0,1\}^{2^n}$ (regarded as the truth table of a function), MCSP is the problem of determining the minimum size of the n-input circuit whose truth table corresponds to x, i.e., the circuit complexity of x.

Hirahara [16] revealed that the approximation versions of the aforementioned problems are efficiently solvable in the *worst case* based on the errorless average-case easiness. For every $\sigma \colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, let $\mathsf{Gap}_\sigma\mathsf{MINKT}$ denote the problem of approximating the t-time-bounded Kolmogorov complexity of $x \in \{0,1\}^n$ within an additive error term $\sigma(\cdot, n)$. For every $\epsilon \in [0,1]$, let $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ denote the problem of approximating the circuit complexity of $x \in \{0,1\}^{2^n}$ within a multiplicative approximation factor $2^{(1-\epsilon)n}$. The formal definitions of these problems are presented in Section 3.2, where they are defined as promise problems. Hirahara's theorem is stated as follows.

▶ **Theorem 1** ([16]). *If $\mathsf{DistNP} \subseteq \mathsf{AvgP}$, then there exist a function $\sigma(s,n) = \sqrt{s} \cdot \mathsf{polylog}(n)$ and a constant $\epsilon > 0$ such that $\mathsf{Gap}_\sigma\mathsf{MINKT} \in \mathsf{pr}\text{-}\mathsf{ZPP}$ and $\mathsf{Gap}_\epsilon\mathsf{MCSP} \in \mathsf{pr}\text{-}\mathsf{BPP}$. Furthermore, these results are relativized.*[2]

Errorless Heuristica II: PAC Learning

PAC (Probably Approximately Correct) learning is one of the well studied subjects in theoretical computer science, introduced by Valiant [40]. In the PAC learning model, a learner is required to learn *all* target functions f in the target class on *all* unknown example distributions D, i.e., the learner constructs a good approximator for f from passively collected data of the form $(x, f(x))$, where each x (called an example) is selected according to D. In other words, the performance of the learner is measured by the worst-case analysis on target functions and example distributions, and this task is not directly captured as a

[2] A subsequent result [18] improved the approximation errors using a potentially non-relativizing proof technique of [9].

distributional problem. Nevertheless, Hirahara and Nanashima [20] revealed that these worst-case requirements in PAC learning are performed based on only the average-case easiness of NP under a natural computational assumption on example distributions.

▶ **Theorem 2** ([20]). *If* DistNP ⊆ AvgP*, then* P/poly *is PAC learnable in polynomial time on all unknown* P/poly*-samplable example distributions. Furthermore, this result is relativized.*

Errorless Heuristica III: No Auxiliary-Input Cryptography

The aforementioned results are sufficient to break the security of any efficiently computable *auxiliary-input* cryptographic primitive, as observed in [3, 21], which is yet another notable consequence of DistNP ⊆ AvgP. An auxiliary-input primitive, introduced in [36, 37], is defined as a family of primitives and has the weak security condition that at least one primitive in the family is required to be secure depending on each adversary. In other words, an adversary for an auxiliary-input primitive needs to succeed in breaking *all* primitives in the family, and this task is not captured directly as a distributional NP problem. Nevertheless, we can efficiently break any auxiliary-input cryptographic primitive in errorless Heuristica.

▶ **Theorem 3.** *If* DistNP ⊆ AvgP*, then there is no auxiliary-input one-way function. Furthermore, this result is relativized.*

The three theorems mentioned above demonstrate that several fascinating tasks concerning worst-case requirements can be performed in errorless Heuristica. By contrast, there is no result which shows the feasibility of a similar task in error-prone Heuristica. Thus, there are two possibilities: the errorless condition is essential in the aforementioned results, or they can be extended by similar (especially, relativizing) proof techniques. Determining which is correct is important to understand the capability and limitation of the technique for the worst-case to average-case reduction within NP developed by Hirahara [16]. Particularly, a significant line of work [26, 30, 2, 23, 31, 32] shows the characterization of a one-way function (OWF) based on the error-prone average-case hardness of several central problems in meta-complexity, including GapMINKT and GapMCSP. Therefore, if Hirahara's reduction can be extended to error-prone average-case analogues of these problems, then OWFs is characterized by the worst-case hardness of meta-computational problems. Despite many efforts, however, extending Hirahara's reduction is currently open. Proving Theorems 1, 2, and 3 in error-prone Heuristica is one natural and necessary approach for this research direction, where we consider the stronger assumption that DistNP ⊆ HeurP (instead of the non-existence of OWFs) and attempt to solve easier problems such as breaking auxiliary-input cryptography.

▶ **Question 3.** *Do Theorems 1, 2, and 3 also hold in error-prone Heuristica, i.e., under the assumption that* DistNP ⊆ HeurP*? Or, is there any barrier for such research directions?*

In this study, we address these questions and study the difference between the errorless average-case complexity and the error-prone average-case complexity from the perspective of relativization.

1.1 Our results

Our main contribution is the oracle construction for separating the error-prone average-case hardness and the errorless average-case hardness for distributional NP problems. Furthermore, the proposed oracle also separates the error-prone average-case hardness and (i) the hardness of approximating complexity (i.e., the lower bound of meta-complexity), (ii) the hardness of PAC learning, and (iii) the existence of auxiliary-input cryptographic primitives. Therefore, the proposed oracle exhibits the relativization barrier for Question 3.

We remark several points before presenting the result. When we consider the adversary defined as (a family of) circuits for some cryptographic primitives (e.g., auxiliary-input primitives and hitting set generators), we regard a size function $s(n)$ of an adversary as a function in the length of a hidden seed instead of output of the primitive for simplicity. In addition, we regard a time-bound function of a learning algorithm as a function in the length of examples, i.e., the input size to the target function.

Now, we present the main theorem. The formal definition of each notion in the statement is presented in Section 3.

▶ **Theorem 4.** *For any constant $a > 0$, there exists an oracle $\mathcal{O}$ relative to which the following hold:*

- *(Error-prone average-case easiness of* NP*)* $\mathsf{DistNP}^{\mathcal{O}} \subseteq \mathsf{HeurP}^{\mathcal{O}}$*.*
- *(Errorless average-case hardness of* NP*)* $\mathsf{DistNP}^{\mathcal{O}} \not\subseteq \mathsf{AvgSIZE}^{\mathcal{O}}[2^{an/\log n}]$*.*
- *(Lower bound of meta-complexity)* $\mathsf{Gap}_\sigma\mathsf{MINKT}^{\mathcal{O}} \notin \mathsf{pr\text{-}SIZE}^{\mathcal{O}}[2^{an/\log n}]$ *for any* $\sigma(s,n) = o(s) \cdot \mathsf{polylog}(n)$*. In addition, for each* $\epsilon \in [0,1]$*, there exists* $\delta \in (0,1)$ *such that* $\mathsf{Gap}_\epsilon\mathsf{MCSP}^{\mathcal{O}} \notin \mathsf{pr\text{-}SIZE}^{\mathcal{O}}[2^{n^\delta}]$*.*
- *(Worst-case hardness of learning on uniform distributions)* $\mathsf{SIZE}^{\mathcal{O}}[n]$ *is not weakly PAC learnable with membership queries (MQ) on the uniform distribution by nonuniform* $O(2^{an/\log n})$*-time algorithms. Furthermore, there exists a polynomial* $s(n)$ *such that* $\mathsf{SIZE}^{\mathcal{O}}[s(n)]$ *is not weakly PAC learnable with MQ on the uniform distribution by nonuniform* $2^n/n^{\omega(1)}$*-time algorithms.*
- *(Average-case hardness of distribution-free learning) There exists a polynomial* $s(n)$ *such that* $\mathsf{SIZE}^{\mathcal{O}}[s(n)]$ *is not weakly PAC learnable on average by nonuniform* $O(2^{an/\log n})$*-time algorithms. Furthermore,* $\mathsf{SIZE}^{\mathcal{O}}[n]$ *is not weakly PAC learnable on average by nonuniform* $O(2^{n^\epsilon})$*-time algorithms for some constant* $\epsilon > 0$*.*
- *(Relaxed cryptographic primitives) There exist a hitting set generator (HSG), an auxiliary-input one-way function (AIOWF), an auxiliary-input pseudorandom generator (AIPRG), and an auxiliary-input pseudorandom function (AIPRF) against* $\mathsf{SIZE}^{\mathcal{O}}[2^{an/\log n}]$*.*

The lower bound in the oracle separation is considerably stronger than the polynomial lower bound and holds for the nonuniform computation model.

Wee [42] constructed an oracle relative to which $\mathsf{DistNP} \not\subseteq \mathsf{HeurP}$, and no AIOWF exists against $\mathsf{P/poly}$, which is the opposite separation of one of our results. Combined with Wee's result, our results show that auxiliary-input cryptography and the error-prone average-case hardness of NP are incomparable by any relativizing proof.

1.2 Related Work

The study of oracle separations is initiated by Baker, Gill, and, Solovay [4] to identify the barrier for resolving the P versus NP problem. The study of the average-case complexity is initiated by Levin [29], and later it was brushed up by Impagliazzo [24], where he introduced the notion of five worlds. In the same paper, Impagliazzo first addressed the question on the difference between the errorless complexity and the error-prone complexity. Each relativized world in Impagliazzo's five worlds is found in [4, 25, 42, 27, 8]. Specifically, Impagliazzo found a relativized heuristica in which $\mathsf{DistNP} \subseteq \mathsf{AvgP}$ but $\mathsf{NP} \not\subseteq \mathsf{SIZE}[2^{n^\epsilon}]$ for some $\epsilon > 0$, and Wee found a relativized pessiland in which $\mathsf{DistNP} \not\subseteq \mathsf{HeurP}$, but neither AIOWF nor OWF exists. Watson [41] also constructed a relativized world in which there is no black-box

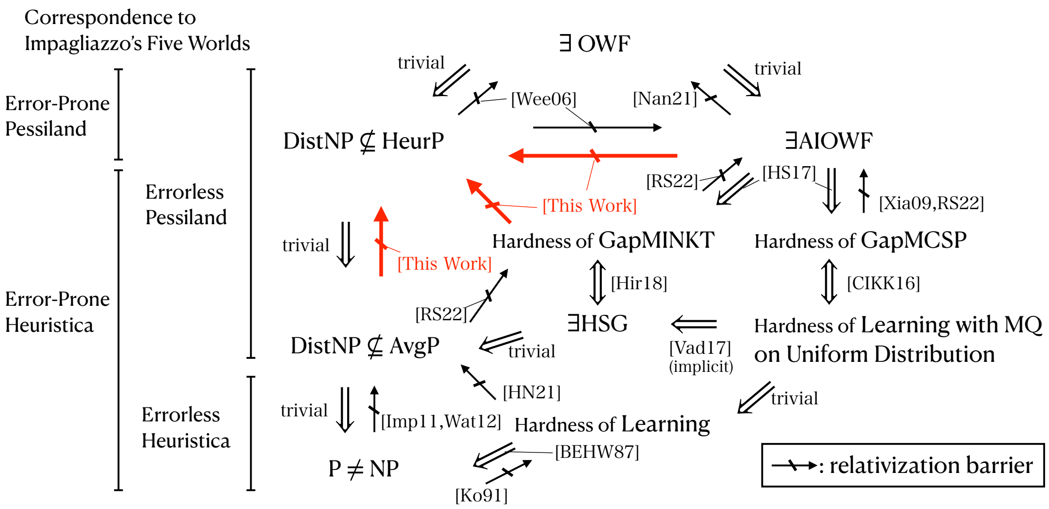

Figure 1 relativization barriers in heuristica and pessiland.

worst-case to average-case reduction for NP, but the reduction presented by Hirahara [16, 18] is non-black-box and overcomes the barrier against black-box reductions. Hirahara and Nanashima [20] improved the oracle construction proposed by Impagliazzo to the tight worst-case hardness of NP and also presented the relativized world in which DistNP $\subseteq$ AvgP, but PAC learning P/poly with MQ is sub-exponentially hard. Ko [28] showed the relativized world in which P $\neq$ NP, but a gap variant of the problem called MINLT is efficiently solvable, which is sufficient for PAC learning P/poly. Xiao [43] found the relativized world in which PAC learning P/poly with MQ is hard, but there is no AIOWF. Ren and Santhanam [39] presented various relativization barriers on the problems in meta-complexity, including the relativized world in which there is no efficient and robust reduction from distributional NP problems to the GapMINKT oracle. They also found the relativized world in which no AIOWF exists but GapMCSP and GapMINKT are hard (even in the error-prone average case). The oracle separation between AIOWF and OWF was discussed in [34]. The relationships among these oracle separation results is visualized in Figure 1.

Hirahara and Santhanam [22] also addressed the errorless complexity versus error-prone complexity problem, and they showed that the equivalence between a non-adaptive errorless to error-prone reduction for NP and an average-case instance checker for NP. They also discussed Question 1 for other classes of distributional problems such as DistPH and Dist(UP $\cap$ coUP) and showed that Dist(UP $\cap$ coUP) $\subseteq$ AvgP if and only if Dist(UP $\cap$ coUP) $\subseteq$ HeurP, i.e., they resolved Question 1 for the subclass UP $\cap$ coUP of NP.

2 Proof Techniques

We present ideas behind our oracle separation. The oracle construction is based on the one presented by Impagliazzo [25], in which the worst-case hardness and the errorless average-case easiness are separated for NP. First, we briefly review the idea and subsequently present its adjustment for the separation between the errorless average-case hardness and the error-prone average-case hardness for NP. For simplicity, we only consider the uniform distribution as the distribution over instances (instead of all sampleable distributions) and a lower bound for P/poly (instead of SIZE$[2^{an/\log n}]$) in this section.

Oracle Separation between NP $\not\subseteq$ P/poly and DistNP $\subseteq$ AvgP

The oracle construction by Impagliazzo [25] is based on the following observation: For a hidden random function $f\colon \{0,1\}^n \to \{0,1\}^{m(n)}$, the answers to most NP computations involving in f are determined by a random restriction of the truth-table of f. Therefore, by providing access to a restrictive NP oracle $\mathcal{A}$ that answers correctly only if the random restriction determines the answer (otherwise, $\mathcal{A}$ answers $\bot$), NP problems become easy on average. By contrast, we require all the information of f to perform all NP computations involving in f. Thus, NP problems remain hard in the worst-case sense in the presence of $\mathcal{A}$. We review how this idea can be implemented.

The oracle in [25] consists of two oracles $\mathcal{V}$ and $\mathcal{A}$ and a hidden internal random function[3] $f\colon \{0,1\}^n \to \{0,1\}^{m(n)}$, where $\mathcal{V}$ represents a verification oracle for the NP relation $R(x, f(x))$ which makes NP worst-case hard, and $\mathcal{A}$ represents a restrictive NP oracle which makes NP average-case easy while retaining the worst-case hardness. The NP oracle $\mathcal{A}$ is given a description of a nondeterministic oracle machine M, an input x, and a time bound T (of the form 1^{T^4} to prevent the circular call for $\mathcal{A}$), simulates $M^{\mathcal{V},\mathcal{A}}(x)$ in T time, and returns the answer, where we allow $\mathcal{A}$ to use only partial values of f on randomly selected positions. If the execution is determined only by the partial information, then $\mathcal{A}$ returns the result; otherwise, $\mathcal{A}$ returns $\bot$.

The average-case easiness of NP follows from the switching lemma for DNFs, where we regard each $f(y)_i$ as a binary variable for each input y and position i (assigned in the random selection of f) and $M^{\mathcal{V},\mathcal{A}}(x)$ (executed in T time) as a $m(n)\cdot T$-DNF formula[4]. If there is no query access to $\mathcal{A}$ by M, then the switching lemma implies that $M^{\mathcal{V}}(x)$ is determined only by the partial information of f for a large fraction of inputs x. In general cases, however, we need to take the recursive query access to $\mathcal{A}$ into account. To address this issue, we introduce a structure in f by multiple applications of random restrictions in the selection of f. Then, for a given time bound 1^{T^4}, we only apply from the first to $i_T := 2^{-1}\log\log T$-th random restrictions. If M queries $(M', x', 1^{T'^4})$ to $\mathcal{A}$ in T time, then $(T')^4 \leq T$ holds. Because $i_{T'} = 2^{-1}\log\log T' \leq i_T - 1$, the answer to the query to $\mathcal{A}$ is determined only by up to the $i_T - 1$-th random restriction. Thus, under an arbitrary condition on up to the $(i_T - 1)$-th random restriction, all the answers from $\mathcal{A}$ (for executing $M^{\mathcal{V},\mathcal{A}}(x)$ in T time) are determined by the condition, and a certain DNF formula is determined regardless of query access to $\mathcal{A}$. Then, the average-case easiness follows from the switching lemma for the i_T-th random restriction (conditioned on up to the $(i_T - 1)$-th random restriction). To apply the switching lemma for the average-case easiness, the parameter for the random restriction (i.e., the unset probability) is set to at most $n^{-\omega(1)}$ (we require a subexponentially small parameter for the subexponential lower bound for NP, as discussed in [20]).

By contrast, the worst-case hardness is shown by considering the $\mathsf{NP}^{\mathcal{V},\mathcal{A}}$ problem $L = \{\langle x, i\rangle : \exists y \text{ s.t. } \mathcal{V}(x,y) = 1 \text{ and } y_i = 1\}$ (in fact, $L \in \mathsf{UP}^{\mathcal{V},\mathcal{A}} \cap \mathsf{coUP}^{\mathcal{V},\mathcal{A}}$). Any polynomial-size circuit C can only access up to the $2^{-1}\log\log \mathsf{poly}(n) = O(\log\log n)$-th random restriction. Intuitively, if there still remain many unassigned values in the $O(\log\log n)$-th random restriction, then C should guess such values at random to find the witness $f(x)$ for L, which implies the worst-case hardness.

[3] This is a slightly modified analog of the original construction discussed in [20] for applying the standard switching lemma for DNFs in the proof instead of the switching lemma on matching variables.

[4] Specifically, the top-most $\vee$ is taken over a nondeterministic configuration path π for M and a choice of f, and each term corresponds to one choice of (π, f) such that $M^{\mathcal{V}}(x)$ accepts, where $\wedge$ in the term is applied to verify the consistency of the values of f at the (at most $m(n)\cdot T$) points $M^{\mathcal{V}}(x)$ queries.

The aforementioned oracle yields the *errorless* average-case easiness because $\mathcal{A}$ returns $\perp$ in the case in which the simulation of the given nondeterministic machine is not determined by the random restrictions. Therefore, a natural idea to separate the errorless and error-prone complexities is that we make $\mathcal{A}$ return a wrong answer in such cases. To implement this idea, the following concerns should be addressed. First, how should the answer from $\mathcal{A}$ be determined in such cases? Note that $\mathcal{A}$ cannot use the values of f assigned at higher levels in the structure to identify the wrong answer because it causes a circular problem, i.e., the DNF representing $M^{\mathcal{V},\mathcal{A}}(x)$ is not determined only by up to the $(i_T - 1)$-th random restriction anymore. Second, how should a distributional problem be determined for the errorless average-case hardness? Particularly, Hirahara and Santhanam [22] showed the equivalence between the errorless average-case easiness and the error-prone average-case easiness of $\mathsf{UP} \cap \mathsf{coUP}$ by relativizing proof techniques. Thus, we cannot hope to prove the errorless average-case hardness for the same $\mathsf{UP} \cap \mathsf{coUP}$ problem L under the error-prone average-case easiness of NP.

First Attempt for DistNP $\not\subseteq$ AvgP/poly and DistNP $\subseteq$ HeurP

The answer to the first question is relatively simple: we make $\mathcal{A}$ always answer 0. The intuition behind this is that an oracle machine given 1 as an answer from $\mathcal{A}$ (for some NP-type statement) can also obtain the witness for this assertion by the self-reducibility of NP; otherwise, the oracle machine can detect the error of $\mathcal{A}$ and output $\perp$. Thus, any error-prone algorithm can be translated into an errorless algorithm when $\mathcal{A}$ answers 1 as a wrong answer at some stage. By contrast, if $\mathcal{A}$ answers 0, i.e., declares "no witness," then there seems no efficient way to detect this error. Thus, we let $\mathcal{A}$ always answer 0, and this choice is indeed crucial in the proof.

By contrast, the answer to the second question is less obvious. Our approach is to construct a hitting set generator (HSG) instead of determining a distributional problem directly. A HSG (against $\mathsf{P/poly}$) is a (family of) efficiently computable function $G\colon \{0,1\}^n \to \{0,1\}^{m(n)}$ which stretches the seed (i.e., $m(n) > n$) and hits any language recognized by a polynomial-size circuit. Specifically, if a polynomial-size circuit C accepts more than half of the strings in $\{0,1\}^{m(n)}$, then C also accepts $G(x)$ for some $x \in \{0,1\}^n$ (for infinitely many $n \in \mathbb{N}$). Constructing a HSG for the errorless average-case hardness is a natural approach because it immediately yields a natural distributional NP problem $(\mathrm{Im}G, \mathrm{Uniform})$ that is hard on average in the errorless setting, and Hirahara [17] demonstrated the equivalence between the errorless average-case hardness of PH and the existence of PH-computable HSGs.

A first attempt to construct a HSG is that we regard the random function $f\colon \{0,1\}^n \to \{0,1\}^{m(n)}$ as a generator, where we let $m(n) > n$. Now, we replace the verification oracle $\mathcal{V}$ with $\mathcal{F}$ defined as $\mathcal{F}(x,i) = f(x)_i$ because the generator requires direct access to f for computing its values. Then, we define the candidate $G^{\mathcal{F},\mathcal{A}}\colon \{0,1\}^n \to \{0,1\}^{m(n)}$ for a HSG as $G^{\mathcal{F},\mathcal{A}}(x) = \mathcal{F}(x,1) \circ \cdots \circ \mathcal{F}(x, m(n)) \, (= f(x))$. However, this generator G is not a HSG, and G can be broken efficiently by using the partial information of f efficiently obtained from $\mathcal{A}$, informally as follows: For each random restriction, an expected fraction of unassigned values in f is $n^{-\omega(1)}$. Thus, for a given string $y \in \{0,1\}^{m(n)}$, we can easily detect the case of $y = G(x)$ for a large fraction of $x \in \{0,1\}^n$ by asking an NP-type query to $\mathcal{A}$ such as "Is there $x \in \{0,1\}^n$ such that $G(x)$ is *partially* consistent with y?" because the answer tends to be fixed to 1 only by the random restriction in $\mathcal{A}$ if such an x exists. After applying the random restrictions $\omega(1)$ times, the aforementioned strategy is sufficient for detecting all the cases of $y \in \mathrm{Im}G$. Thus, some different approach is required.

We remark that we can now regard executing a nondeterministic $M^{\mathcal{F},\mathcal{A}}(x)$ in T time as a T-DNF formula (instead of an $m(n) \cdot t$-DNF) because $\mathcal{F}$ accesses only one entry in f for each query.

Our Construction: Random Restriction with Masks

To construct a HSG, we introduce a new type of random restrictions, *random restriction with masks*, which is crucial to solve Question 2. A random restriction with masks to $f\colon \{0,1\}^n \to \{0,1\}^{m(n)}$ with parameter $p \in [0,1]$ (i.e., the unset probability) is performed as follows: First, we select a random subset $S_1 \subseteq \{0,1\}^n$ of size $p \cdot 2^n$ and then apply a standard random restriction with unset probability p to a variable set $\{f(x)_i : x \in \{0,1\}^n \setminus S_1 \text{ and } i \in [m(n)]\}$, i.e., the random set S_1 performs as a "mask" that prevents restriction. This variant of random restriction is extended to multiple applications inductively as follows: Let S_i be the random subset (i.e., the mask) selected in the i-th random restriction with masks to f. Next, the $(i+1)$-th restriction (with parameter p) is performed by selecting a random subset $S_{i+1} \subseteq S_i$ of size $p \cdot |S_i|$ and applying random restriction to variables except for S_{i+1}.

We consider a modified oracle in which the oracle construction is the same as previously mentioned except that we apply random restrictions with masks instead of the standard random restrictions. For now, we select the unset probability $p(n) = n^{-\log n}$. This choice is sufficient for a HSG against $\mathsf{P/poly}$ and the statement that $\mathsf{DistNP} \not\subseteq \mathsf{AvgP/poly}$. Note that $p(n)$ should be selected more carefully according to the size complexity of the adversary in the formal argument (for the detail, see Section 4).

Specifically, we randomly select the aforementioned oracles $\mathcal{F}$ and $\mathcal{A}$ by selecting the internal random function $f\colon \{0,1\}^n \to \{0,1\}^{m(n)}$ with $\log n$ applications of random restrictions with masks for each $n \in \mathbb{N}$ (after applying random restrictions, we also select the remaining values of f at random). For each $n \in \mathbb{N}$, let $S_{n,\log n} \subseteq \{0,1\}^n$ be the random mask selected in the $\log n$-th restriction. Then, we have $|S_{n,\log n}| = p(n)^{\log n} \cdot 2^n = 2^{n-(\log n)^3}$. Thus, there exist exponentially many $z \in S_{n,\log n} \subseteq \{0,1\}^n$ (we call these hard indices) such that no value in $f(z)$ is assigned by the $\log n$-th restriction. Remember that for a query $(M, x, 1^{T^4})$, the oracle $\mathcal{A}$ applies only up to the $i_T := 2^{-1}\log\log T$-th random restriction. Since any polynomial-size adversary C can make a query only with $T = \mathsf{poly}(n)$, C can only access up to the $O(\log\log n)$-th restrictions. Therefore, any polynomial-size adversary cannot obtain any information about $f(z)$ from $\mathcal{A}$ for each hard index z, and oracle access to $\mathcal{F}(z,i) = f(z)_i$ is indistinguishable from access to a random function for such adversaries.

The aforementioned argument is sufficient for constructing a HSG. In fact, by defining the generator G as $G^{\mathcal{F},\mathcal{A}}(x) = \mathcal{F}(x,1) \circ \cdots \circ \mathcal{F}(x, m(n))$, we can show that G is a HSG against $\mathsf{P/poly}$ by a similar argument as in [20]. Furthermore, the random restriction method with masks has another advantage: even if we select exponentially large $m(n)$, it still provides hard indices z such that $\mathcal{A}$ does not reveal any information of $f(z)$ to polynomial-size adversaries. Specifically, by letting $m(n) = 2^n \cdot n$ (i.e., the length of the truth table of a mapping from n-bit to n-bit), we can prepare an auxiliary-input oracle $\mathcal{F}\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ such that $\mathcal{F}(z,\cdot)$ is (computationally) indistinguishable from a random oracle for subexponentially many hard z's. Therefore, $F(\cdot,\cdot)$ is an AIOWF because it is known that a random oracle is also a OWF with probability 1 over the choice of the random oracle (cf. [27, 11]). Furthermore, by the technique presented in [45], we can construct an AIPRG based on the auxiliary-input analog of a random oracle with less security loss than the general methods to convert OWFs into PRGs (e.g., [15]). In the formal proof of Theorem 4, we first construct such an AIPRG and subsequently show the related hardness notion (e.g., HSGs and the hardness of learning) to prevent security loss. Note that the aforementioned argument does not yield standard cryptographic primitives such as OWFs because the set of hard indices are selected at random, and there is no efficient sampling algorithm that selects a hard index with high probability.

A random restriction with masks assigns fewer variables than a standard random restriction. Therefore, the remaining problem is whether the error-prone average-case easiness is preserved in the modified oracle construction. This issue can be addressed by the choice of the answer (i.e., 0) from $\mathcal{A}$ when the simulation is not determined by random restrictions. The proof is outlined as follows (for the formal proof, see Section 5).

For convenience, we regard that a random restriction with masks is performed as follows: (i) a standard random restriction is applied to remaining variables at the stage, (ii) the random subset $S_{i+1} \subseteq S_i$ is selected in the same manner, and (iii) the values in $f(z)$ are returned to unassigned for each $z \in S_{i+1}$. Let us call the first step (resp. the second and third steps) a *restriction* (resp. *reverse*) step. The random restriction in the restriction step is merely a standard one. Thus, by the standard switching lemma, we can show that the value of a T-DNF ϕ (representing the execution of a nondeterministic machine in T time) is determined with high probability at this stage. Therefore, it is sufficient to show that the answer from $\mathcal{A}$ rarely changes in the reverse step.

For simplicity, we use the notation $*$ to refer to the cases in which the DNF ϕ is not fixed by the random restriction. Then, there are $3 \times 3 = 9$ possibilities about the change in the state on the restricted ϕ, i.e., from $\{0, 1, *\}$ (in the restriction step) to $\{0, 1, *\}$ (in the reverse step). Obviously, we do not need to consider the following 3 cases: $\{0\} \to \{0\}$, $\{1\} \to \{1\}$, and $\{*\} \to \{*\}$. Since we cancel some assignments in the reverse step, the following 4 cases do not occur: $\{*\} \to \{0, 1\}$, $\{0\} \to \{1\}$, and $\{1\} \to \{0\}$. Furthermore, because $\mathcal{A}$ answers 0 in the case of $*$, we do not need to consider the case of $\{0\} \to \{*\}$. Therefore, the remaining case is only $\{1\} \to \{*\}$.

We show that the case of $\{1\} \to \{*\}$ rarely occurs as follows. Since the T-DNF formula ϕ is satisfied in the restriction step, there must exist a satisfied term τ of size T. If ϕ becomes unfixed in the reverse step, then τ is also unfixed. This event occurs only if there exists $z \in \{0, 1\}^n$ such that some variable $f(z)_i$ is contained in τ (for some i), and z is selected on the choice of the random subset in the reverse step. Since τ covers at most T indices z in literals, this probability is at most $T \cdot p(n) = T \cdot n^{-\log n}$. Particularly, for solving a NP problem by $\mathcal{A}$, we only need to simulate a nondeterministic machine in $\mathsf{poly}(n)$ times, so we can let $T = \mathsf{poly}(n)$. Therefore, the error probability that the answer from $\mathcal{A}$ is changed in the reverse step is negligible.

Limitations of Our Technique and Future Direction

We remark that the aforementioned argument above heavily relies on the characteristics of DNFs (i.e., nondeterministic machines). Currently, it is unclear whether the proposed argument can be extended to a general case of constant-depth circuits, even for depth-3 $\wedge$-$\vee$-$\wedge$-circuits (which corresponds to Π_2^p). By contrast, the oracle separation between the worst-case hardness and the errorless average-case easiness for NP in [25] is naturally extended for PH, as explicitly discussed in [20] by considering the switching lemma for constant-depth circuits. Therefore, we pose the following open question for the further research on the difference between the errorless and error-prone average-case complexity.

▶ **Question 4.** *Is there any oracle $\mathcal{O}$ relative to which* $\mathsf{DistNP}^{\mathcal{O}} \not\subseteq \mathsf{AvgP}^{\mathcal{O}}/\mathsf{poly}$ *and* $\mathsf{DistPH}^{\mathcal{O}} \subseteq \mathsf{HeurP}^{\mathcal{O}}$*? Or, is there a relativizing proof which shows that* $\mathsf{DistPH} \subseteq \mathsf{HeurP} \implies \mathsf{DistNP} \subseteq \mathsf{AvgP}/\mathsf{poly}$*?*

The fact that we failed to extend our results to PH might suggest the feasibility of proving $\mathsf{DistPH} \subseteq \mathsf{HeurP} \implies \mathsf{DistNP} \subseteq \mathsf{AvgP/poly}$. Furthermore, we failed to improve our lower bound $2^{o(n/\log n)}$ on the time complexity of errorless average-case algorithms to $2^{o(n)}$. In this light, we conjecture that the worst-case-to-average-case connection of Hirahara [19], which shows that $\mathsf{DistNP} \subseteq \mathsf{AvgP} \implies \mathsf{UP} \subseteq \mathsf{DTIME}(2^{O(n/\log n)})$, can be extended to the error-prone average-case complexity by using a relativizing proof.

▶ **Conjecture 5.** *For every oracle $\mathcal{O}$, if $\mathsf{DistNP}^{\mathcal{O}} \subseteq \mathsf{HeurP}^{\mathcal{O}}$, then $\mathsf{UP}^{\mathcal{O}} \subseteq \mathsf{BPTIME}^{\mathcal{O}}[2^{O(n/\log n)}]$.*

3 Preliminaries

For each $n \in \mathbb{N}$, let $[n] = \{1, \dots, n\}$. For each $x \in \{0,1\}^n$ and $i \in [n]$, we let x_i denote the i-th bit of x and $x_{\leq i}$ denote $x_1 \circ \cdots \circ x_i$. For a distribution D, we write $x \leftarrow D$ to refer to a random sampling x according to D. For a finite set S, we also use the notation $x \leftarrow_u S$ to denote the uniform sampling from S. For each $n \in \mathbb{N}$, we let U_n denote the uniform distribution over $\{0,1\}^n$ or a random variable selected uniformly at random from $\{0,1\}^n$ in context. We use the notation negl to represent a certain negligible function, i.e., for any polynomial $p(n)$, $\mathsf{negl}(n) < 1/p(n)$ for sufficiently large $n \in \mathbb{N}$. For a randomized algorithm A using $r(n)$ random bits on an n-bit input, we use $A(x; s)$ to refer to the execution of $A(x)$ with a random tape s for $x \in \{0,1\}^n$ and $s \in \{0,1\}^{r(n)}$.

For any oracles $\mathcal{O}_0$ and $\mathcal{O}_1$, we let $\mathcal{O}_0 + \mathcal{O}_1$ denote the combination, i.e., for any $b \in \{0,1\}$ and any $x \in \{0,1\}^*$, $(\mathcal{O}_0 + \mathcal{O}_1)(b \circ x) = \mathcal{O}_b(x)$.

In this paper, we assume the basic knowledge of probability theory, including the union bound, Markov's inequality, Hoeffding's inequality, and the Borel–Cantelli lemma.

For each $p \in [0,1]$ and set S of variables taking binary values, we define a p-random restriction ρ to S as a partial assignment $\rho\colon S \to \{0,1,*\}$ (where $*$ represents "unassigned") randomly selected as follows: for each $x \in S$,

$$\rho(x) = \begin{cases} * & \text{with probability } p \\ 0 & \text{with probability } (1-p)/2 \\ 1 & \text{with probability } (1-p)/2. \end{cases}$$

For every restriction ρ to S and function f defined on S, we let $f|_\rho$ denote the restricted function obtained by applying a partial assignment to f according to ρ.

3.1 Average-Case Complexity

We present the notions in average-case complexity theory. Further backgrounds can be found in a survey [7].

We say that a family $D = \{D_n\}_{n \in \mathbb{N}}$ of distributions, where each D_n is a distribution on $\{0,1\}^n$, is (polynomial-time) samplable if there exists a randomized sampling algorithm S such that the distribution of $S(1^n)$ is identical to D_n for each $n \in \mathbb{N}$. We consider a family of distributions as a single distribution on instances. We define a distributional problem as a pair of a language $L \subseteq \{0,1\}^*$ and a distribution $D = \{D_n\}_{n \in \mathbb{N}}$ on instances. For a standard complexity class $\mathcal{C}$ (e.g., NP), we define its average-case extension $\mathsf{Dist}\mathcal{C}$ as $\mathsf{Dist}\mathcal{C} = \{(L, D) : L \in \mathcal{C}, D \text{ is samplable}\}$.

We present the *errorless* average-case easiness. We say that a distributional problem (L, D) has an *errorless* heuristic algorithm A with failure probability $\epsilon\colon \mathbb{N} \to (0,1)$ if (1) A outputs $L(x)$ $(:= \mathbb{1}\{x \in L\})$ or $\perp$ (which represents "failure") for every $x \in \mathrm{supp}(D)$, and (2) the failure probability that $A(x)$ outputs $\perp$ over the choice of $x \leftarrow D$ is bounded above by $\epsilon(n)$ for each $n \in \mathbb{N}$. Note that an errorless heuristic algorithm never outputs an incorrect value $\neg L(x)$ for any $x \in \mathrm{supp}(D)$. Then, for every $\epsilon : \mathbb{N} \to (0,1)$, we define a class $\mathsf{Avg}_\epsilon\mathsf{P}$ as a class of distributional problems that have a polynomial-time errorless heuristic algorithm with failure probability $\epsilon(n)$. Furthermore, we say that a distributional problem (L, D) has an *errorless* heuristic scheme A if A is given an instance $x \in \mathrm{supp}(D)$ and $\epsilon \in (0,1)$ as input and satisfies the condition of an errorless heuristic algorithm with failure probability ϵ. We define a class AvgP as a class of distributional problems that have a polynomial-time errorless heuristic scheme. It is not hard to verify that $\mathsf{AvgP} \subseteq \mathsf{Avg}_{1/p(n)}\mathsf{P}$ for any polynomial $p(n)$.

Next, we present the *error-prone* average-case easiness. We say that a distributional problem (L, D) has an *error-prone* heuristic algorithm A with failure probability $\epsilon \colon \mathbb{N} \to (0,1)$ if the failure probability that $A(x) \neq L(x)$ over the choice of $x \leftarrow D$ is bounded above by $\epsilon(n)$ for each $n \in \mathbb{N}$. Note that an error-prone heuristic algorithm may output an incorrect value $\neg L(x)$, but the error probability is bounded above by $\epsilon(n)$. Then, for every $\epsilon : \mathbb{N} \to (0,1)$, we define a class $\mathsf{Heur}_\epsilon\mathsf{P}$ as a class of distributional problems that have a polynomial-time error-prone heuristic algorithm with failure probability $\epsilon(n)$. We also define an *error-prone* heuristic scheme and the class HeurP in the same manner as the errorless case.

We also define classes $\mathsf{AvgP/poly}$, $\mathsf{HeurP/poly}$, $\mathsf{AvgSIZE}[s(n)]$, and $\mathsf{HeurSIZE}[s(n)]$ for each size parameter $s(n)$ in the same manner as above.

3.2 Meta-Complexity

Next, we define problems $\mathsf{GapMINKT}$ and $\mathsf{GapMCSP}$ formally. In this study, we fix a universal Turing machine U arbitrarily to specify the Kolmogorov complexity.

▶ **Definition 6** (Kolmogorov complexity, GapMINKT). *For each $t \in \mathbb{N}$ and $x \in \{0,1\}^*$, we define the t-time-bounded Kolmogorov complexity $\mathsf{K}^t(x)$ of x as*

$$\mathsf{K}^t(x) = \min_{p \in \{0,1\}^*} \{|p| : U(p) \text{ outputs } x \text{ in } t \text{ time}\}.$$

We also define $\mathsf{K}(x)$ by $\mathsf{K}(x) = \lim_{t\to\infty} \mathsf{K}^t(x)$.

For a function $\sigma \colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, $\mathsf{Gap}_\sigma\mathsf{MINKT}$ is a promise problem (Π_Y, Π_N) defined as $\Pi_Y = \left\{(x, 1^s, 1^t) : \mathsf{K}^t(x) \le s\right\}$ and $\Pi_N = \{(x, 1^s, 1^t) : \mathsf{K}(x) > s + \sigma(s, |x|)\}$[5].

▶ **Definition 7** (Circuit complexity, GapMCSP). *For each $n \in \mathbb{N}$ and $x \in \{0,1\}^{2^n}$, we define the circuit complexity $\mathsf{cc}(x)$ of x as the minimum size of an n-input circuit whose truth table corresponds to x.*

For a constant $\epsilon \in [0,1]$, $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ is a promise problem (Π_Y, Π_N) defined as $\Pi_Y = \{(x, 1^s) : n \in \mathbb{N}, x \in \{0,1\}^{2^n}, \mathsf{cc}(x) \le s\}$ and $\Pi_N = \{(x, 1^s) : n \in \mathbb{N}, x \in \{0,1\}^{2^n}, \mathsf{cc}(x) > 2^{(1-\epsilon)n} \cdot s\}$.

3.3 Learning

We define a concept class as a subset of $\{f \colon \{0,1\}^n \to \{0,1\} : n \in \mathbb{N}\}$. For any concept class $\mathcal{C}$ and $n \in \mathbb{N}$, we let $\mathcal{C}_n$ represent $\mathcal{C} \cap \{f \colon \{0,1\}^n \to \{0,1\}\}$. Then, the (weak) PAC learning model with membership queries (MQ) is defined as follows. We refer to a family $D = \{D_n\}_{n\in\mathbb{N}}$, where each D_n is a distribution on $\{0,1\}^n$, as an example distribution in the learning context.

[5] Note that this formulation of $\mathsf{GapMINKT}$ has a relaxed (i.e., easier) requirement than one discussed in [16, 18], in the sense that we do not consider the time-bound in "no" cases. In other words, we only need to distinguish efficiently generated strings from strings no short program can generate even in time-unbounded settings.

▶ **Definition 8** (PAC learning [40]). *Let $\mathcal{C}$ be a concept class, D be an example distribution, and $t\colon \mathbb{N} \to \mathbb{N}$ be a time-bound function. We say that a (possibly nonuniform) randomized oracle machine L, referred to as a weak learner, (weakly) learns $\mathcal{C}$ with MQ on D in time $t(n)$ if L satisfies the following conditions for some polynomial $p(n)$:*

1. *L is given $n \in \mathbb{N}$ as the input and oracle access to $\mathsf{EX}_{f,D}$ (called an example oracle) and MQ_f (called a membership query oracle), which are determined by a target function $f \in \mathcal{C}_n$ and the example distribution D.*
2. *For each access (with no input), $\mathsf{EX}_{f,D}()$ returns an example of the form $(x, f(x))$, where x is selected identically and independently according to D_n. Furthermore, for each access with input $x \in \{0,1\}^n$, $\mathsf{MQ}_f(x)$ returns $f(x)$.*
3. *For each $n \in \mathbb{N}$ and target function $f\colon \{0,1\}^n \to \{0,1\}$, L outputs a circuit $h\colon \{0,1\}^n \to \{0,1\}$ that is $(\frac{1}{2} - \frac{1}{p(n)})$-close to f under D with probability at least $2/3$, i.e., L satisfies the following condition:*

$$\Pr_{L,\mathsf{EX}}\left[L^{\mathsf{EX}_{f,D},\mathsf{MQ}_f}(n) \text{ outputs } h \text{ such that } \Pr_{x \leftarrow D}[h(x) \neq f(x)] \leq \frac{1}{2} - \frac{1}{p(n)}\right] \geq \frac{2}{3}.$$

4. *$L^{\mathsf{EX}_{f,D},\mathsf{MQ}_f}(n)$ halts in time $t(n)$ for each $n \in \mathbb{N}$.*

We say that a concept class $\mathcal{C}$ is weakly PAC learnable with MQ on D in $t(n)$ time if there exists a $t(n)$-time weak learner for $\mathcal{C}$.

When the example distribution is uniform, a randomized learner can simulate EX based on MQ and its randomness with no loss of time complexity. Thus, we ignore EX in learning on the uniform distribution without loss of generality.

We also define an average-case analog of PAC learning, in which a target function is randomly selected according to some fixed distribution F (called a target distribution). By contrast, we consider the distribution-free setting on example distributions, formally, as follows:

▶ **Definition 9** (learning on average). *Let $\mathcal{C}$ be a concept class, $t\colon \mathbb{N} \to \mathbb{N}$ be a time-bound function, and $F = \{F_n\}_{n \in \mathbb{N}}$ be a target distribution, where each F_n is a distribution on $\mathcal{C}_n$. We say that a (possibly nonuniform) randomized oracle machine L, referred to as a weak learner, (weakly) learns $\mathcal{C}$ on average with respect to F in time $t(n)$ if L satisfies the following condition for some polynomial $p(n)$: For any $n \in \mathbb{N}$ and example distribution D_n on $\{0,1\}^n$, $L^{\mathsf{EX}_{f,D_n}}(n)$ halts in time $t(n)$ (for each $f \in \mathrm{supp}(F_n)$) and*

$$\Pr_{f \leftarrow F_n}\left[\Pr_{L,\mathsf{EX}_{f,D}}\left[L^{\mathsf{EX}_{f,D}}(n) \text{ outputs a circuit } h \text{ such that } \Pr_{x \leftarrow D}[h(x) \neq f(x)] \leq \frac{1}{2} - \frac{1}{p(n)}\right] \geq \frac{2}{3}\right] \geq \frac{1}{p(n)}.$$

We say that a concept class $\mathcal{C}$ is not weakly PAC learnable on average in $t(n)$ time if there exists a polynomial-time samplable target distribution F such that there is no $t(n)$-time weak learner that satisfies the condition above with respect to F.

3.4 Cryptography

We introduce cryptographic primitives. Let $\mathcal{C}$ be a complexity class of adversaries (e.g., $\mathsf{P/poly}$). We regard the complexity parameter (e.g., time and size) on $\mathcal{C}$ as a function in the size of a hidden seed for primitives.

▶ **Definition 10** (auxiliary-input one-way function). *Let $n, m \colon \mathbb{N} \to \mathbb{N}$ be polynomials. We say that $f = \{f_z \colon \{0,1\}^{n(|z|)} \to \{0,1\}^{m(|z|)}\}_{z \in \{0,1\}^*}$ is an auxiliary-input one-way function (AIOWF) against $\mathcal{C}$ if each $f_z(x)$ is polynomial-time computable from (z, x), and for any adversary A in $\mathcal{C}$, there exists an infinite subset $Z_A \subseteq \{0,1\}^*$ such that for every $z \in Z_A$,*

$$\Pr\left[f_z(A(z, f_z(U_{n(|z|)}))) = f_z(U_{n(|z|)})\right] < \mathsf{negl}(|z|).$$

▶ **Definition 11** (auxiliary-input pseudorandom generator). *Let $n, m \colon \mathbb{N} \to \mathbb{N}$ be polynomials. We say that $G = \{G_z \colon \{0,1\}^{n(|z|)} \to \{0,1\}^{m(|z|)}\}_{z \in \{0,1\}^*}$ is an auxiliary-input pseudorandom generator (AIPRG) against $\mathcal{C}$ if each $G_z(x)$ is polynomial-time computable from (z, x), $n(\ell) < m(\ell)$ holds for any $\ell \in \mathbb{N}$, and for any adversary A in $\mathcal{C}$, there exists an infinite subset $Z_A \subseteq \{0,1\}^*$ such that for every $z \in Z_A$,*

$$\left|\Pr\left[A(z, G_z(U_{n(|z|)})) = 1\right] - \Pr\left[A(z, U_{m(|z|)}) = 1\right]\right| < \mathsf{negl}(|z|).$$

▶ **Definition 12** (auxiliary-input pseudorandom function). *We say that $F = \{F_z \colon \{0,1\}^{|z|} \times \{0,1\}^{|z|} \to \{0,1\}\}_{z \in \{0,1\}^*}$ is an auxiliary-input pseudorandom function (AIPRF) against $\mathcal{C}$ if each F_z is polynomial-time computable from z and its input, and for any adversary $A^?$ in (an oracle machine analog of) $\mathcal{C}$, there exists an infinite subset $Z_A \subseteq \{0,1\}^*$ such that for every $z \in Z_A$,*

$$\left|\Pr_{A, u \sim \{0,1\}^{|z|}}\left[A^{F_z(u,\cdot)}(z) = 1\right] - \Pr_{A, \phi_{|z|}}\left[A^{\phi_{|z|}(\cdot)}(z) = 1\right]\right| < \mathsf{negl}(|z|),$$

where $\phi_{|z|} \colon \{0,1\}^{|z|} \to \{0,1\}$ denotes a truly random function.

When the auxiliary-input z is obvious in context, we write $n(|z|)$ and $m(|z|)$ as n and m, respectively.

▶ **Definition 13** (hitting set generator). *Let $\ell, m \colon \mathbb{N} \to \mathbb{N}$ be polynomials. We say that $G = \{G_n\}_{n \in \mathbb{N}}$, where $G_n \colon \{0,1\}^{\ell(n)} \to \{0,1\}^{m(n)}$ is a hitting set generator (HSG) against $\mathcal{C}$ if each G_n is polynomial-time computable, $\ell(n) < m(n)$ holds for each $n \in \mathbb{N}$, and G hits any language recognized by adversaries in $\mathcal{C}$ in the following sense: For any adversary A in $\mathcal{C}$, let $L_A = \{L_{A,n}\}_{n \in \mathbb{N}}$ be a language recognized by A, where $L_{A,n} \subseteq \{0,1\}^n$ for each $n \in \mathbb{N}$. Then, for infinitely many $n \in \mathbb{N}$, the following holds:*

$$|L_{A,m(n)}| > 2^{m(n)-1} \Longrightarrow L_{A,m(n)} \cap \mathrm{Im} G_n \neq \emptyset.$$

4 Oracle Construction

In this section, we formally present the proposed oracle construction. Let $a > 0$ be a parameter.

▶ **Construction.** *$\mathcal{O}_a = \mathcal{F} + \mathcal{A}$, where each oracle is randomly selected according to the following process:*

1. *Let $t(n) = 2^{an/\log n}$ be the upper bound on the time of nonuniform adversaries and $c = 7a$.*
2. *Define functions p and $i_{\max}$ as $p(n) = t(n)^{-6}$ and $i_{\max}(n) = \frac{1}{c} \log\log t(n)$. Here, p is a parameter of random restriction, and $i_{\max}$ is the number of applications of random restrictions.*
3. *For each $n \in \mathbb{N}$, define a set $V_{n,0}$ of variables taking binary values as follows:*

$$V_{n,0} = \{F_{z,x,\ell} : z, x \in \{0,1\}^n, \ell \in [n]\}.$$

4. *For each $n \in \mathbb{N}$, let $S_{n,0} = \{0,1\}^n$.*
5. *For each $n \in \mathbb{N}$ and $i \in [i_{\max}(n)-1]$, we inductively (on i) select a $p(n)$-random restriction $\rho^*_{n,i}$ to $V_{n,i-1}$ and a random subset $S_{n,i} \subseteq S_{n,i-1}$ of size $p(n) \cdot |S_{n,i-1}|$. Then, we define a restriction $\rho_{n,i}$ to $V_{n,i-1}$ and a subset $V_{n,i} \subseteq V_{n,i-1}$ as follows:*

$$\rho_{n,i}(z,x,\ell) = \begin{cases} * & \text{if } z \in S_{n,i} \\ \rho^*_{n,i}(z,x,\ell) & \text{otherwise} \end{cases}$$

$$V_{n,i} = \rho_{n,i}^{-1}(*) \ \left(= {\rho^*_{n,i}}^{-1}(*) \cup \{F_{z,x,\ell} : z \in S_{n,i}\}\right).$$

   We also define $\rho_{n,i_{\max}(n)}$ as a full assignment to $V_{n,i_{\max}(n)-1}$ selected uniformly at random. Let $\rho_{n,i} \equiv \rho_{n,i_{\max}(n)}$ for each $i \geq i_{\max}(n)+1$. We use the notation $\rho_{n,\leq i}$ to represent the composite restriction $\rho_{n,1}\cdots\rho_{n,i}$ to $V_{n,0}$ for each n and i.
6. *Define $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$, where $\mathcal{F}_n\colon \{0,1\}^n \times \{0,1\}^n \times [n] \to \{0,1\}$, as $\mathcal{F}_n(z,x,\ell) = \rho_{n,\leq i_{\max}(n)}(z,x,\ell)$.*
7. *Define $\mathcal{A}$ as follows: On input $(M, x, 1^{T^{2^c}})$, where M is a nondeterministic oracle machine, $x \in \{0,1\}^*$, and $T \in \mathbb{N}$, the oracle $\mathcal{A}(M,x,1^{T^{2^c}})$ returns 0 or 1 according to the following procedure:*

```
1: Let $i_T := \frac{1}{c}\log\log T$.
2: Construct a $T$-DNF $\phi$ on variables in $V_{n,0}$ representing the execution of $M^{\mathcal{F}+\mathcal{A}}(x)$ in $T$ steps, where the top-most OR corresponds to the nondeterminism on a possible choice of $\mathcal{F}$ (say, $\mathcal{F}'$) and an accepting path of $M^{\mathcal{F}'+\mathcal{A}}(x)$, and each term performs verification whether $M$'s at most $T$ queries (say, $(z_1,x_1,\ell_1),\ldots,(z_q,x_q,\ell_q)$ for some $q \leq T$) are consistent with the actual choices of $\mathcal{F}$, i.e., for each $i \in [q]$, the term contains $F_{z_i,x_i,\ell_i}$ if $\mathcal{F}'(z_i,x_i,\ell_i) = 1$; otherwise, $\neg F_{z_i,x_i,\ell_i}$ as a literal.
3: If $\phi|_{\rho_{1,\leq i_T},\ldots,\rho_{T,\leq i_T}} \equiv b$ for some $b \in \{0,1\}$, then return $b$, otherwise, return 0.
```

We can verify that $\mathcal{A}$ is well-defined (i.e., not circular on recursive calls for $\mathcal{A}$) as follows.

▶ **Proposition 14.** *For each input, the value of $\mathcal{A}(M,x,1^{T^{2^c}})$ is determined only by $\rho_{n,j}$ (equivalently, $\rho^*_{n,j}$ and $S_{n,j}$) for $n \leq T$ and $j \leq i_T$ (remember that $i_T = \frac{1}{c}\log\log T$).*

Proof. We show the proposition by induction on T. Remember that, on input $(M,x,1^{T^{2^c}})$, the oracle $\mathcal{A}$ first makes a T-DNF ϕ based on M independently of the values of $\mathcal{F}$.

Suppose that M makes some query $(M',x',1^{T'^{2^c}})$ to $\mathcal{A}$ for constructing ϕ. Since the length of such a query is at most T, we have $T'^{2^c} \leq T$ and

$$i_{T'} = \frac{1}{c}\log\log T' \leq \frac{1}{c}\log\log T^{\frac{1}{2^c}} = \frac{1}{c}\log\log T - 1 = i_T - 1.$$

By the induction hypothesis, the answer of $\mathcal{A}(M',x',1^{T'^{2^c}})$ is determined by only $\rho_{n,j}$ for $n \leq T'$ and $j \leq i_T - 1$, and so is ϕ. Then, $\mathcal{A}$ determines the answer by restricting ϕ by $\rho_{n,j}$ for $n \leq T$ and $j \leq i_T$. Therefore, $\mathcal{A}(M,x,1^{T^{2^c}})$ is determined only by $\rho_{n,j}$ for $n \leq T$ and $j \leq i_T$. ◀

When the parameter a is clear from the context, we omit the subscript a from $\mathcal{O}_a$.

5 Error-Prone Average-Case Easiness of NP

In this section, we show the error-prone average-case easiness of NP.

▶ **Theorem 15.** *With probability 1 over the choice of $\mathcal{O}_a$, $\mathsf{DistNP}^{\mathcal{O}_a} \subseteq \mathsf{HeurP}^{\mathcal{O}_a}$ holds.*

First, we introduce several notations. For each choice of $\mathcal{O}$, we define the oracle $\mathcal{A}^*$ in the same manner as $\mathcal{A}$ except we apply $\rho_{1,\leq i_T-1}\rho^*_{1,i_T}, \dots, \rho_{T,\leq i_T-1}\rho^*_{T,i_T}$ to ϕ instead of $\rho_{1,\leq i_T}, \dots, \rho_{T,\leq i_T}$. Note that $\mathcal{A}^*$ executes a given nondeterministic machine M with access to $\mathcal{A}$ (rather than $\mathcal{A}^*$) to construct the corresponding DNF ϕ. We can verify that $\mathcal{A}^*$ is well-defined (i.e., not circular) in the same manner as Proposition 14.

Now, we show that $\mathcal{A}$ and $\mathcal{A}^*$ do not differ considerably.

▶ **Lemma 16.** *For each input $(M, x, 1^{T^{2c}})$ to $\mathcal{A}$, we have that*

$$\Pr_{\mathcal{O}}\left[\mathcal{A}(M, x, 1^{T^{2c}}) \neq \mathcal{A}^*(M, x, 1^{T^{2c}})\right] = O(T^{-4}).$$

Proof. Let $i := i_T = (1/c)\log\log T$. For all $n \leq T$ and $j \leq i-1$, we fix $\rho^*_{n,j}$, $S_{n,j}$, and $\rho^*_{n,i}$ arbitrarily; let C_T denote this condition. Notice that the DNF formula ϕ_{C_T} constructed by $\mathcal{A}$ and $\mathcal{A}^*$ is determined only by C_T because all answers to recursive calls for $\mathcal{A}$ are determined by C_T as in Proposition 14. Let $\phi'_{C_T} = \phi_{C_T}|_{\rho_{1,\leq i-1},\dots,\rho_{T,\leq i-1}}$. Then, the value of $\mathcal{A}(M, x, 1^{T^{2c}})$ (resp. $\mathcal{A}^*(M, x, 1^{T^{2c}})$) is determined by $\phi'_{C_T}|_{\rho_{1,i},\dots,\rho_{T,i}}$ (resp. $\phi'_{C_T}|_{\rho^*_{1,i},\dots,\rho^*_{T,i}}$).

For any DNF formula ϕ and a restriction ρ, there are the following three cases: (i) $\phi|_\rho \equiv 0$, (ii) $\phi|_\rho \equiv 1$, or (iii) $\phi|_\rho$ does not become a constant (we write this case as $\phi|_\rho \equiv *$). Following this case analysis, there exist $3^2 = 9$ cases on $(\phi'_{C_T}|_{\rho^*_{1,i},\dots,\rho^*_{T,i}}, \phi'_{C_T}|_{\rho_{1,i},\dots,\rho_{T,i}})$. However, since each $\rho_{n,i}$ is a subrestriction of $\rho^*_{n,i}$ (i.e., $\rho_{n,i}$ assigns values only to variables that are also assigned by $\rho^*_{n,i}$), the following 4 cases do not occur: $(0,1), (1,0), (*,0)$, and $(*,1)$. Further, in the cases of $(0,0), (1,1), (*,*)$, and $(*,0)$, the answers by $\mathcal{A}^*$ and $\mathcal{A}$ do not differ because $\mathcal{A}^*$ and $\mathcal{A}$ return 0 in the case of $*$. Thus, we only need to consider the case of $(1,*)$.

By the aforementioned argument, the probability in the lemma is expressed as follows:

$$\Pr_{\mathcal{O}}\left[\mathcal{A}(M, x, 1^{T^{2c}}) \neq \mathcal{A}^*(M, x, 1^{T^{2c}})\right] = \mathrm{Exp}_{C_T}\left[\Pr_{S_{1,i},\dots,S_{T,i}}\left[\phi'_{C_T}|_{\rho_{1,i},\dots,\rho_{T,i}} \equiv * \middle| C_T, \phi'_{C_T}|_{\rho^*_{1,i},\dots,\rho^*_{T,i}} \equiv 1\right]\right].$$

To bound the probability in the right-hand side for each condition, we consider the case in which $\phi'_{C_T}|_{\rho^*_{1,i},\dots,\rho^*_{T,i}} \equiv 1$. Then, we can select a term τ in ϕ'_{C_T} such that $\tau|_{\rho^*_{1,i},\dots,\rho^*_{T,i}} \equiv 1$. For each $n \leq T$, define $Z_n \subseteq \{0,1\}^n$ as

$$Z_n = \{z \in \{0,1\}^n : \exists (x, \ell) \in \{0,1\}^n \times [n] \text{ s.t. a variable } F_{z,x,\ell} \text{ is contained in } \tau\}.$$

Since ϕ'_{C_T} is a T-DNF, $|Z_n| \leq T$ for each $n \leq T$. Furthermore, for any $n \in \mathbb{N}$ such that $i_{\max}(n) \leq i-1$, we have that $Z_n = \emptyset$ because all such variables must be assigned by $\rho_{n,\leq i-1}$. If $\phi'_{C_T}|_{\rho_{1,i},\dots,\rho_{T,i}} \equiv *$, then $\tau|_{\rho_{1,i},\dots,\rho_{T,i}} \equiv *$ must hold. This event occurs only if $\bigcup_{n\leq T}(S_{n,i} \cap Z_n) \neq \emptyset$ holds. Since each $S_{n,i}$ is selected from $S_{n,i-1}$ uniformly at random, this probability is bounded above by

$$\begin{aligned}
\Pr\left[\bigcup_{n\leq T}(S_{n,i} \cap Z_n) \neq \emptyset\right] &\leq \sum_{n\leq T} \Pr\left[S_{n,i} \cap Z_n \neq \emptyset\right] \\
&\leq \sum_{n\leq T : i_{\max}(n)\geq i} |Z_n| \cdot |S_{n,i}|/|S_{n,i-1}| \\
&\leq O(T) \cdot \sum_{n : t^{-1}(T)\leq n\leq T} p(n) \\
&\leq O(T^2 \cdot t(t^{-1}(T))^{-6}) \\
&= O(T^{-4}).
\end{aligned}$$

Thus, we conclude that

$$\Pr_{\mathcal{O}}\left[\mathcal{A}(M,x,1^{T^{2^c}}) \neq \mathcal{A}^*(M,x,1^{T^{2^c}})\right] = \mathrm{Exp}_{C_T}\left[\Pr_{S_{1,i},\ldots,S_{T,i}}\left[\phi'_{C_T}|_{\rho_{1,i},\ldots,\rho_{T,i}} \equiv * \middle| C_T, \phi'_{C_T}|_{\rho^*_{1,i},\ldots,\rho^*_{T,i}} \equiv 1\right]\right] \leq O(T^{-4}).$$ ◀

Furthermore, we can show the average-case easiness of NP under the oracle access to $\mathcal{A}^*$ (instead of $\mathcal{A}$). This part is similar to in [25, 20]. Thus, we defer the proof to Appendix A.

▶ **Lemma 17** ([25, 20]). *Let M be a $t_M(n)$-time nondeterministic oracle machine and S be a randomized polynomial-time oracle sampling machine. We assume that $S(1^n)$ takes at most $t_S(n)$ time to generate an instance of length n. Then, the following event occurs with probability 1 over the choice of $\mathcal{O}$: for any $n \in \mathbb{N}$,*

$$\Pr_{x \leftarrow S^{\mathcal{O}}(1^n)}\left[M^{\mathcal{O}}(x) \neq \mathcal{A}^*(M,x,1^{T^{2^c}})\right] \leq O(n^{-4}),$$

where $T = \max\{n^{2^c}, t_M(n)^{2^c}, t_S(n)^{2^c}\}$.

Now, we derive the average-case easiness of NP from Lemmas 16 and 17 .

Proof of Theorem 15. We consider an arbitrary distributional NP problem (L,D) and assume that L is specified by a $t_M(n)$-time nondeterministic oracle machine M, and D is specified by a randomized $t_S(n)$-time oracle sampling machine S. Let $T = \max\{n^{2^c}, t_M(n)^{2^c}, t_S(n)^{2^c}\}$ as in Lemma 17.

We construct an error-prone heuristic algorithm B for (L,D) as follows: On input $x \in \{0,1\}^n$, B queries $(M,x,1^{T^{2^c}})$ to $\mathcal{A}$ and returns the same answer. In the following, we will verify that the error probability of B (over the choice of $\mathcal{O}$ and $S^{\mathcal{O}}(1^n)$) is bounded above by $O(n^{-4})$ for each input size n. Then, by applying Markov's inequality and the Borel–Cantelli lemma, the error probability of B is bounded above by n^{-2} for all sufficiently large n with probability 1 over the choice of $\mathcal{O}$. Since the number of tuples (M,S) is countable, we can conclude that all distributional NP problems have an error-prone heuristic algorithm with error probability at most n^{-2} with probability 1 over the choice of $\mathcal{O}$. Based on the argument in [24, Proposition 3], this is sufficient for the statement that $\mathsf{DistNP}^{\mathcal{O}} \subseteq \mathsf{HeurP}^{\mathcal{O}}$.

Therefore, it is sufficient to show that, for any $n \in \mathbb{N}$,

$$\Pr_{\mathcal{O}, x \leftarrow S^{\mathcal{O}}(1^n)}\left[M^{\mathcal{O}}(x) \neq \mathcal{A}(M,x,1^{T^{2^c}})\right] \leq O(n^{-4}).$$

Obviously, this event occurs only if (i)$\mathcal{A}(M,x,1^{T^{2^c}}) \neq \mathcal{A}^*(M,x,1^{T^{2^c}})$ or (ii) $M^{\mathcal{O}}(x) \neq \mathcal{A}^*(M,x,1^{T^{2^c}})$ occur. By Lemmas 16 and 17 and the union bound, we have

$$\begin{aligned}
&\Pr_{\mathcal{O}, x \leftarrow S^{\mathcal{O}}(1^n)}\left[M^{\mathcal{O}}(x) \neq \mathcal{A}(M,x,1^{T^{2^c}})\right] \\
&\quad \leq \Pr_{\mathcal{O}, x \leftarrow S^{\mathcal{O}}(1^n)}\left[\mathcal{A}(M,x,1^{T^{2^c}}) \neq \mathcal{A}^*(M,x,1^{T^{2^c}})\right] + \Pr_{\mathcal{O}, x \leftarrow S^{\mathcal{O}}(1^n)}\left[M^{\mathcal{O}}(x) \neq \mathcal{A}^*(M,x,1^{T^{2^c}})\right] \\
&\quad = O(T^{-4}) + O(n^{-4}) = O(n^{-4}) + O(n^{-4}) = O(n^{-4}).
\end{aligned}$$ ◀

6 Errorless Average-Case Hardness of NP

In this section, we show the hardness part of our oracle separation. First, we show the existence of AIPRG relative to $\mathcal{O}_a$. Then we show the other hardness results, including the errorless average-case hardness for NP, as corollaries.

We use the following theorem, which shows the existence of PRGs based on a random oracle.

▶ **Theorem 18** ([45]). *For each $n \in \mathbb{N}$, let $\mathcal{R}_n \colon \{0,1\}^n \to \{0,1\}^n$ be a random function oracle, i.e., $\mathcal{R}_n$ is selected uniformly at random from $\{f \colon \{0,1\}^n \to \{0,1\}^n\}$.*

There exist a polynomial-time deterministic oracle machine $G^?$ and constants $c \geq 1$ and $b, \epsilon > 0$ satisfying the following: For any $n \in \mathbb{N}$ and $x \in \{0,1\}^{cn}$, $G^{\mathcal{R}_n}(x)$ generates a binary string of length $4cn$, and all oracle circuits $C^?$ of size 2^{bn} satisfy that

$$\left| \Pr_{U_{cn}} \left[C^{\mathcal{R}_n}(G^{\mathcal{R}_n}(U_{cn})) = 1 \right] - \Pr_{U_{4cn}} \left[C^{\mathcal{R}_n}(U_{4cn}) = 1 \right] \right| \leq 2^{-\epsilon n},$$

with probability at least $1 - 2^n$ over the choice of $\mathcal{R}_n$.

Furthermore, the result above is relativized, i.e., the above holds in the presence of an arbitrary oracle $\mathcal{O}$ independent of the choice of R_n.

Now, we show the existence of AIPRG relative to $\mathcal{O}_a$.

▶ **Theorem 19.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, there exists an AIPRG $G^{\mathcal{O}_a} = \{G_z^{\mathcal{O}_a}\}_{z \in \{0,1\}^*}$ against $\mathsf{SIZE}^{\mathcal{O}_a}[2^{\epsilon a n/\log n}]$ for some absolute constant $\epsilon > 0$, where $G_z^{\mathcal{O}_a} \colon \{0,1\}^{|z|} \to \{0,1\}^{3|z|}$ for each $z \in \{0,1\}^*$.*

Proof. Let $G^?$ and c be the oracle machine and the constant in Theorem 18, respectively. Then, we define an AIPRG $G' = \{G'_z\}_{z \in \{0,1\}^*}$ as $G'^{\mathcal{O}}_z(x) = G^{\mathcal{F}_{z'}}(x')_{\leq 3|z|}$, where $x \in \{0,1\}^{|z|}$, $z' = z_{\leq \lfloor |z|/c \rfloor}$, $x' = x_{\leq c|z'|}$, and $\mathcal{F}_{z'} \colon \{0,1\}^{|z'|} \to \{0,1\}^{|z'|}$ is defined as $\mathcal{F}_{z'}(y) = \mathcal{F}(z', y, 1) \circ \cdots \circ \mathcal{F}(z', y, |z'|)$. The validity of the truncation is verified as that $|x'| = c|z'| \leq c \cdot |z|/c = |z| = |x|$ and $|G^{\mathcal{F}_{z'}}(x')| = 4|x'| = 4c|z'| \geq 4c(|z|/c - 1) \geq 4|z| - 4c \geq 3|z|$ for any z with $|z| \geq 4c$.

Let $\epsilon = 1/2c$ and $s(n) = 2^{\epsilon a n/\log n}$. We show that G' above is an AIPRG against $\mathsf{SIZE}^{\mathcal{O}}[s(n)]$. Suppose there exists a family $C = \{C_n\}_{n \in \mathbb{N}}$ of oracle circuits of size $s(n)$ that breaks G', i.e., for any sufficiently large $n \in \mathbb{N}$ and any $z \in \{0,1\}^n$,

$$\left| \Pr_{U_n} \left[C_n^{\mathcal{O}}(z, G'^{\mathcal{O}}_z(U_n)) = 1 \right] - \Pr_{U_{3n}} \left[C_n^{\mathcal{O}}(z, U_{3n}) = 1 \right] \right| > \frac{1}{\mathsf{poly}(n)}.$$

Fix $n \in \mathbb{N}$ arbitrarily, and let $n' = \lfloor n/c \rfloor$. We consider an arbitrary choice of $\mathcal{O}$ except for the values of $\rho_{n', i_{\max}(n')}$ (we write this condition as R for convenience). Fix $z \in \{0,1\}^n$ such that $z' = z_{\leq n'} \in S_{n', i_{\max}(n')-1}$ arbitrary (where $S_{\cdot,\cdot}$ is the random set in the oracle construction). We refer to such z as a hard index.

Since the size of C_n is at most $s(n) = 2^{\frac{an}{2c \log n}} \leq 2^{\frac{a(\lfloor n/c \rfloor)}{\log(\lfloor n/c \rfloor)}} = t(n')$ for sufficiently large n (where t is the time-bound function in the oracle construction), the answers to queries by C_n of the form $\mathcal{A}(M, y, 1^{T^{2^c}})$ do not depend on the values of $\mathcal{F}_{z'}$ and are determined by condition R because they are determined only by the restrictions $\rho_{\cdot,j}$ for $j \leq c^{-1} \log\log T \leq c^{-1} \log\log s(n)^{1/2^c} \leq c \log\log t(n') - 1 < i_{\max}(n')$. Now, we consider an arbitrary choice of $\rho_{n', i_{\max}(n')}$ except for the values of $\mathcal{F}_{z'}$ and denote this condition by R'. It is not hard to verify that $\mathcal{F}_{z'}$ is selected uniformly at random even under the conditions R and R'. Therefore, under the conditions R and R', we can identify the query access to $\mathcal{O}$ by C with the query access to another oracle $\mathcal{O}'$ (determined only by R and R') and a random function oracle $\mathcal{F}_{z'}$ that are selected independently of $\mathcal{O}'$.

For any $n \in \mathbb{N}$, let E_n be an event (over the choice of $\mathcal{F}_{z'}$) that there exists a circuit C' of size $2^{bn'}$, where b represents the constant in Theorem 18, such that

$$\left| \Pr_{U_n} \left[C'^{\mathcal{O}}(G'^{\mathcal{F}_{z'}}_z(U_n)) = 1 \right] - \Pr_{U_{3n}} \left[C'^{\mathcal{O}}(U_{3n}) = 1 \right] \right|$$
$$= \left| \Pr_{U_n} \left[C'^{\mathcal{O}', \mathcal{F}_{z'}}(G^{\mathcal{F}_{z'}}(U_n)) = 1 \right] - \Pr_{U_{3n}} \left[C'^{\mathcal{O}', \mathcal{F}_{z'}}(U_{3n}) = 1 \right] \right| > \frac{1}{\mathsf{poly}(n)}.$$

Then, by Theorem 18 (relative to $\mathcal{O}'$), we have $\Pr_{\mathcal{O}}[E_n|R,R'] \leq 2^{-\Omega(n)}$. By the Borel–Cantelli lemma, E_n occurs only for finitely many n's with probability 1 over the choice of $\mathcal{O}$ conditioned on R, R' (i.e., the choice of $\mathcal{F}_{z'}$). By taking the expectation over R, R', we can show that, with probability 1 over the choice of $\mathcal{O}$, there is no family C' of $2^{bn'}$-size circuits satisfying that for any sufficiently large $n \in \mathbb{N}$, there exists a hard index $z \in \{0,1\}^n$ such that

$$\left|\Pr_{U_n}\left[C'^{\mathcal{O}}(G'^{\mathcal{F}_{z'}}_z(U_n)) = 1\right] - \Pr_{U_{3n}}\left[C'^{\mathcal{O}}(U_{3n}) = 1\right]\right| > \frac{1}{\mathsf{poly}(n)}.$$

However, the circuit C in the assumption violates this statement by embedding a hard index z as auxiliary-input because the size is at most $n + s(n) = O(2^{(a/2c)\cdot n/\log n}) = o(2^{bn'})$. Therefore, we conclude that, with probability 1 over the choice of $\mathcal{O}$, there is no such a circuit C of size $s(n)$, and G' is an AIPRG against $\mathsf{SIZE}[s(n)]$. ◀

Now, we present the consequences of the existence of AIPRG. First, it is the well-established that any PRG is also OWF (cf. [12, Proposition 3.3.8]). This result is trivially extended to the case of auxiliary-input primitives, and the following holds.

▶ **Corollary 20.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, there exists an AIOWF against $\mathsf{SIZE}^{\mathcal{O}_a}[2^{\epsilon an/\log n}]$ relative to $\mathcal{O}_a$ for some absolute constant $\epsilon > 0$.*

Next, AIPRG implies HSG by regarding the auxiliary-input as a part of the hidden input to HSG. It is not hard to verify the security (refer to [34, Lemma 18] for the formal proof).

▶ **Corollary 21.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, there exists an HSG $G^{\mathcal{O}_a} = \{G_n^{\mathcal{O}_a}\}_{n\in\mathbb{N}}$ against $\mathsf{SIZE}^{\mathcal{O}_a}[2^{\epsilon an/\log n}]$ for some absolute constant $\epsilon > 0$, where $G_n^{\mathcal{O}_a} \colon \{0,1\}^{2n} \to \{0,1\}^{3n}$ for each $n \in \mathbb{N}$.*

Furthermore, the existence of HSGs implies the errorless average-case hardness of NP, as observed in [21] in the context of natural proofs and the average-case complexity of the minimum circuit size problem.

▶ **Corollary 22.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, $\mathsf{DistNP}^{\mathcal{O}_a} \not\subseteq \mathsf{AvgSIZE}^{\mathcal{O}_a}[2^{\frac{\epsilon an}{\log n}}]$ for some absolute constant $\epsilon > 0$.*

Proof. Let $G^{\mathcal{O}}$ and ϵ be the HSG and the constant in Corollary 21, respectively. Then, we define the language $L^{\mathcal{O}}$ as $L^{\mathcal{O}} := \mathrm{Im} G^{\mathcal{O}}$. Obviously, $L^{\mathcal{O}} \in \mathsf{NP}^{\mathcal{O}}$ and $(L^{\mathcal{O}}, \{U_n\}_{n\in\mathbb{N}}) \in \mathsf{DistNP}^{\mathcal{O}}$. Thus, it is sufficient to show that $(L^{\mathcal{O}}, \{U_n\}_{n\in\mathbb{N}}) \notin \mathsf{Avg}_{1/4}\mathsf{SIZE}^{\mathcal{O}}[2^{\frac{\epsilon an}{4\log n}}]$.

For contradiction, we assume that $(L^{\mathcal{O}}, \{U_n\}_{n\in\mathbb{N}}) \in \mathsf{Avg}_{1/4}\mathsf{SIZE}^{\mathcal{O}}[2^{\frac{\epsilon an}{4\log n}}]$. Then, there exists a family $C = \{C_n\}_{n\in\mathbb{N}}$ of $O(2^{\frac{\epsilon an}{4\log n}})$-size oracle circuits for $(L^{\mathcal{O}}, \{U_n\}_{n\in\mathbb{N}})$, i.e., for any sufficiently large $n \in \mathbb{N}$,

$$\Pr_{y \leftarrow_u \{0,1\}^{3n}}\left[C_{3n}^{\mathcal{O}}(y) = \bot\right] \leq 1/4$$

and for each $y \in \{0,1\}^{3n}$,

$$C_{3n}^{\mathcal{O}}(y) \in \{\mathbb{1}(y \in L^{\mathcal{O}}), \bot\}.$$

Note that the size of C_{3n} is at most $O(2^{\frac{\epsilon a\cdot 3n}{4\log 3n}})$.

Next, we define an adversary C' for $G^{\mathcal{O}}$ as follows: for a given $y \in \{0,1\}^{3n}$ (i.e., the length of seed is $2n$), $C'^{\mathcal{O}}_{2n}$ simulates $C^{\mathcal{O}}_{3n}(y)$, and if C_{3n} returns 1 or $\bot$, then C'_{2n} outputs 0; otherwise (i.e., if C_{3n} returns 0), C'_{2n} outputs 1. Then, based on the aforementioned inequalities, it is not hard to verify that for any sufficiently large $n \in \mathbb{N}$,

$$\Pr_{y \leftarrow_u \{0,1\}^{3n}}\left[C'^{\mathcal{O}}_{2n}(y) = 0\right] \leq \frac{1}{4} + \frac{|\{G^{\mathcal{O}}(x) : x \in \{0,1\}^{2n}\}|}{|\{0,1\}^{3n}|} \leq \frac{1}{4} + \frac{2^{2n}}{2^{3n}} < \frac{1}{2},$$

and for any $x \in \{0,1\}^{2n}$, we have $C^{\mathcal{O}}_{3n}(G^{\mathcal{O}}(x)) \in \{1, \bot\}$ and $C'^{\mathcal{O}}_{2n}(G^{\mathcal{O}}(x)) = 0$.

Therefore, C' succeeds in avoiding $\mathrm{Im}G^{\mathcal{O}}$, and the size is bounded above by $O(2^{\frac{\epsilon a n}{\log n}}) = O(2^{\frac{\epsilon a(2n)}{\log(2n)}})$. This contradicts Corollary 21. Thus, we conclude that $(L^{\mathcal{O}}, \{U_n\}_{n\in\mathbb{N}}) \notin \mathsf{Avg}_{1/4}\mathsf{SIZE}^{\mathcal{O}}[2^{\frac{\epsilon a n}{4\log n}}]$. ◀

Furthermore, based on the GGM construction in [13], we can translate AIPRGs into AIPRFs. In the security proof, the seed length is preserved, and an adversary of size $s(n)$ for the PRF is translated into an adversary of size $s(n) \cdot \mathsf{poly}(n)$ for the original PRG, where poly is a polynomial depending on the computational cost of the PRG. This observation implies the following:

▶ **Corollary 23.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, there exists an AIPRF $f^{\mathcal{O}_a} = \{f^{\mathcal{O}_a}_z\}_{z\in\{0,1\}^*}$ against $\mathsf{SIZE}^{\mathcal{O}_a}[2^{\epsilon a n/\log n}]$ for some absolute constant $\epsilon > 0$, where $f^{\mathcal{O}_a}_z : \{0,1\}^{|z|} \times \{0,1\}^{|z|} \to \{0,1\}$ for each $z \in \{0,1\}^*$.*

Nanashima [33] observed that the existence of AIPRF implies the average-case hardness of distribution-free learning, where the complexity of the concept class depends on the complexity of computing AIPRF. Thus, we obtain the following, where we apply the standard transformation from nonuniform Turing machines to circuit families.

▶ **Corollary 24.** *There exist a polynomial $s(n)$ and a constant $\epsilon > 0$ such that for any $a > 0$, $\mathsf{SIZE}[s(n)]$ is not PAC learnable on average by nonuniform $O(2^{\epsilon a n/\log n})$-time algorithms with probability 1 over the choice of $\mathcal{O}_a$.*

Without loss of generality, we can let $s(n) = n^b$ in above for some $b > 0$. By the simple padding argument, where we stretch an n-bit example into an $s(n)$-bit example, the size complexity of the target function becomes $O(n)$ (for the input length $s(n)$) in above. Since $2^{s(n)^{1/(b+1)}} = o(2^{\epsilon a n/\log n})$, we have the following:

▶ **Corollary 25.** *There exists $\epsilon > 0$ such that for any $a > 0$, $\mathsf{SIZE}[n]$ is not PAC learnable on average by nonuniform $O(2^{n^\epsilon})$-time algorithms with probability 1 over the choice of $\mathcal{O}_a$.*

The existence of AIPRF also implies the (worst-case) hardness of PAC learning $\mathsf{SIZE}[s(n)]$ on the uniform distribution, as observed in [3], where $s(n)$ is a polynomial depending on the complexity of computing the AIPRF. In our case, we can directly construct a hard-to-learn class and show the hardness of learning $\mathsf{SIZE}[n]$.

▶ **Theorem 26.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, $\mathsf{SIZE}^{\mathcal{O}_a}[n]$ is not weakly PAC learnable with MQ on the uniform distribution by nonuniform $O(2^{\epsilon a n/\log n})$-time algorithms relative to $\mathcal{O}^a$ for some absolute constant $\epsilon > 0$.*

The proof of Theorem 26 is an analog of the proof in [20]. For completeness, we present the formal proof in Appendix B.

Furthermore, Oliveira and Santhanam [35] showed the speedup phenomena in PAC learning with MQ on the uniform distribution. One of their results is stated below.

▶ **Theorem 27** (speedup lemma [35])**.** *For any polynomial $s(n)$ and constant $\epsilon > 0$, there exists a polynomial $s'(n)$ such that if $\mathsf{SIZE}[s(n)]$ is not weakly PAC learnable with MQ on uniform distribution by nonuniform $O(2^{n^{\epsilon}})$-time algorithms, then $\mathsf{SIZE}[s'(n)]$ is not weakly PAC learnable with MQ on uniform distribution by nonuniform $2^n/n^{\omega(1)}$-time algorithms. Furthermore, this result is relativized.*

Theorems 26 and 27 immediately imply the following.

▶ **Corollary 28.** *There exists a polynomial $s(n)$ such that for any $a > 0$, $\mathsf{SIZE}[s(n)]$ is not PAC learnable with MQ on the uniform distribution by nonuniform $2^n/n^{\omega(1)}$-time algorithms with probability 1 over the choice of $\mathcal{O}_a$.*

Finally, we mention the hardness of approximation problems in meta-complexity. The hardness of $\mathsf{GapMINKT}$ follows from the existence of HSG in the same manner as Corollary 22.

▶ **Corollary 29.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, $\mathsf{Gap}_\sigma\mathsf{MINKT}^{\mathcal{O}_a} \notin \mathsf{pr\text{-}SIZE}^{\mathcal{O}_a}[2^{\epsilon an/\log n}]$ for any $\sigma(s,n) = o(s) \cdot \mathsf{polylog}(n)$, where $\epsilon > 0$ is an absolute constant.*

Proof sketch. Suppose that $\mathsf{Gap}_\sigma\mathsf{MINKT}^{\mathcal{O}} \in \mathsf{pr\text{-}SIZE}^{\mathcal{O}}[2^{(\epsilon/4)an/\log n}]$ for some $\sigma(s,n) = o(s) \cdot \mathsf{polylog}(n)$, where $\epsilon > 0$ is the constant in Corollary 21. Then, there exists an $O(2^{(\epsilon/4)an/\log n})$-size oracle circuit C for $\mathsf{Gap}_\sigma\mathsf{MINKT}^{\mathcal{O}}$. Based on C, we can construct an adversary for an arbitrary HSG $G^{\mathcal{O}} \colon \{0,1\}^{2n} \to \{0,1\}^{3n}$ because, for each $n \in \mathbb{N}$ and $x \in \{0,1\}^{2n}$, it holds that $\mathsf{K}^{t,\mathcal{O}}(G^{\mathcal{O}}(x)) \le 2n + O(1)$ for a proper choice of $t = \mathsf{poly}(n)$ and $\Pr_{y \leftarrow_u \{0,1\}^{3n}}[\mathsf{K}^{\mathcal{O}}(y) \ge 3n - 2\ (> 2n + O(1) + \sigma(2n + O(1), n))] \ge 3/4$. It is not hard to verify that the size of the adversary based on C is at most $O(2^{(\epsilon/4)3an/\log 3n})$. Thus, this contradicts Corollary 21. ◀

Furthermore, Carmosino, Impagliazzo, Kabanets, and Kolokolova [10] constructed a PAC learning algorithm for $\mathsf{P/poly}$ with MQ on the uniform distribution based on an algorithm for $\mathsf{GapMCSP}$ (originally, the existence of natural proofs). As the contraposition, we obtain the following from Corollary 28.

▶ **Corollary 30.** *Let $a > 0$ be an arbitrary constant. With probability 1 over the choice of $\mathcal{O}_a$, for each $\epsilon > 0$, there exists $\delta > 0$ such that $\mathsf{Gap}_\epsilon\mathsf{MCSP}^{\mathcal{O}_a} \notin \mathsf{pr\text{-}SIZE}^{\mathcal{O}_a}[2^{n^{\delta}}]$.*

Theorem 4 follows from Theorems 15, 19, and 26 and Corollaries 20– 30 by selecting an appropriately large parameter in the oracle construction according to $a > 0$ in the statement of Theorem 4.

References

1 S Aaronson and A Wigderson. Algebrization: A New Barrier in Complexity Theory. *ACM Trans. Comput. Theory*, 1(1), February 2009.

2 E. Allender, M. Cheraghchi, D. Myrisiotis, H. Tirumala, and I. Volkovich. One-Way Functions and a Conditional Variant of MKTP. In *41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2021, December 15-17, 2021, Virtual Conference*, volume 213 of *LIPIcs*, pages 7:1–7:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

3 B. Applebaum, B. Barak, and D. Xiao. On Basing Lower-Bounds for Learning on Worst-Case Assumptions. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, FOCS'08, pages 211–220, 2008.

4 T. Baker, J. Gill, and R. Solovay. Relativizations of the $\mathcal{P} =? \mathcal{NP}$ Question. *SIAM Journal on Computing*, 4(4):431–442, 1975.

5 Manuel Blum and Sampath Kannan. Designing Programs that Check Their Work. *J. ACM*, 42(1):269–291, 1995. doi:10.1145/200836.200880.

6 Manuel Blum and Silvio Micali. How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits. *SIAM J. Comput.*, 13(4):850–864, 1984. doi:10.1137/0213053.

7 A. Bogdanov and L. Trevisan. Average-Case Complexity. *Foundations and Trends in Theoretical Computer Science*, 2(1):1–106, 2006.

8 G. Brassard. Relativized Cryptography. *IEEE Transactions on Information Theory*, 29(6):877–894, 1983.

9 Harry Buhrman, Lance Fortnow, and Aduri Pavan. Some Results on Derandomization. *Theory Comput. Syst.*, 38(2):211–227, 2005. doi:10.1007/s00224-004-1194-y.

10 M. Carmosino, R. Impagliazzo, V. Kabanets, and A. Kolokolova. Learning Algorithms from Natural Proofs. In *Proceedings of the 31st Conference on Computational Complexity*, CCC'16. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016.

11 R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 305–313, 2000.

12 O. Goldreich. *Foundations of Cryptography: Volume 1.* Cambridge University Press, New York, NY, USA, 2006.

13 O. Goldreich, S. Goldwasser, and S. Micali. How to Construct Random Functions. *J. ACM*, 33(4):792–807, August 1986.

14 J. Håstad. *Computational Limitations of Small-depth Circuits.* MIT Press, Cambridge, MA, USA, 1987.

15 J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A Pseudorandom Generator from Any One-way Function. *SIAM J. Comput.*, 28(4):1364–1396, March 1999.

16 S. Hirahara. Non-Black-Box Worst-Case to Average-Case Reductions within NP. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 247–258, 2018.

17 S Hirahara. Characterizing Average-Case Complexity of PH by Worst-Case Meta-Complexity. In *IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS 2020)*, pages 50–60, 2020.

18 S. Hirahara. Non-Disjoint Promise Problems from Meta-Computational View of Pseudorandom Generator Constructions. In *35th Computational Complexity Conference (CCC 2020)*, volume 169 of *LIPIcs*, pages 20:1–20:47, Dagstuhl, Germany, 2020.

19 S. Hirahara. Average-Case Hardness of NP from Exponential Worst-Case Hardness Assumptions. In *53rd Annual ACM Symposium on Theory of Computing (STOC 2021)*, 2021.

20 S. Hirahara and M. Nanashima. On Worst-Case Learning in Relativized Heuristica. In *62nd Annual IEEE Symposium on Foundations of Computer Science (FOCS 2021)*, 2021.

21 S. Hirahara and R. Santhanam. On the Average-Case Complexity of MCSP and Its Variants. In Ryan O'Donnell, editor, *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, volume 79 of *LIPIcs*, pages 7:1–7:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

22 S. Hirahara and R. Santhanam. Errorless versus Error-prone Average-case Complexity. In *The 13th Innovations in Theoretical Computer Science (ITCS 2022)*, 2022.

23 R. Ilango, H. Ren, and R. Santhanam. Hardness on any Samplable Distribution Suffices: New Characterizations of One-Way Functions by Meta-Complexity. *Electron. Colloquium Comput. Complex.*, page 82, 2021.

24 R. Impagliazzo. A personal view of average-case complexity. In *Proceedings of IEEE Tenth Annual Conference on Structure in Complexity Theory*, pages 134–147, 1995.

25 R. Impagliazzo. Relativized Separations of Worst-Case and Average-Case Complexities for NP. In *2011 IEEE 26th Annual Conference on Computational Complexity*, pages 104–114, 2011.

26 R. Impagliazzo and L. Levin. No better ways to generate hard NP instances than picking uniformly at random. In *Proceedings of the 31st Annual Symposium on Foundations of Computer Science*, FOCS'90, pages 812–821, 1990.

27 R. Impagliazzo and S. Rudich. Limits on the Provable Consequences of One-Way Permutations. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, pages 44–61, New York, NY, USA, 1989. ACM.

28 K. Ko. On the Complexity of Learning Minimum Time-Bounded Turing Machines. *SIAM Journal on Computing*, 20(5):962–986, 1991.

29 L. A Levin. Average Case Complete Problems. *SIAM J. Comput.*, 15(1):285–286, February 1986.

30 Y. Liu and R. Pass. On One-way Functions and Kolmogorov Complexity. In *IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS 2020)*, pages 1243–1254, 2020. doi:10.1109/FOCS46700.2020.00118.

31 Y. Liu and R. Pass. A Note on One-way Functions and Sparse Languages. *Electron. Colloquium Comput. Complex.*, page 92, 2021.

32 Y. Liu and R. Pass. On the Possibility of Basing Cryptography on EXP $\neq$ BPP. In *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part I*, volume 12825 of *Lecture Notes in Computer Science*, pages 11–40. Springer, 2021.

33 M. Nanashima. A Theory of Heuristic Learnability. In *Proceedings of the 34th Conference on Learning Theory, COLT'21*. PMLR, August 2021.

34 M. Nanashima. On Basing Auxiliary-Input Cryptography on NP-Hardness via Nonadaptive Black-Box Reductions. In James R. Lee, editor, *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, volume 185 of *LIPIcs*, pages 29:1–29:15, Dagstuhl, Germany, 2021. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

35 I. Oliveira and R. Santhanam. Conspiracies between Learning Algorithms, Circuit Lower Bounds, and Pseudorandomness. In *Proceedings of the 32nd Computational Complexity Conference*, CCC'17, Dagstuhl, DEU, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

36 R. Ostrovsky. One-way functions, hard on average problems, and statistical zero-knowledge proofs. In *Proceedings of the Sixth Annual Structure in Complexity Theory Conference*, pages 133–138, 1991.

37 R. Ostrovsky and A. Wigderson. One-way functions are essential for non-trivial zero-knowledge. In *Proceedings of the 2nd Israel Symposium on Theory and Computing Systems*, ISTCS'93, pages 3–17, June 1993.

38 A. A. Razborov and S. Rudich. Natural Proofs. *Journal of Computer and System Sciences*, 55(1):24–35, 1997.

39 H. Ren and R. Santhanam. A Relativization Perspective on Meta-Complexity. In *The 39th International Symposium on Theoretical Aspects of Computer Science (STACS 2022)*, 2022.

40 L. Valiant. A Theory of the Learnable. *Commun. ACM*, 27(11):1134–1142, 1984. doi:10.1145/1968.1972.

41 Thomas Watson. Relativized Worlds without Worst-Case to Average-Case Reductions for NP. *ACM Trans. Comput. Theory*, 4(3), September 2012.

42 H Wee. Finding Pessiland. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pages 429–442, Berlin, Heidelberg, 2006. Springer-Verlag.

43 D. Xiao. On basing ZK $\neq$ BPP on the hardness of PAC learning. In *Proceedings of the 24th Conference on Computational Complexity*, CCC'09, pages 304–315, 2009.

44 A. Yao. Theory and Application of Trapdoor Functions. In *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, FOCS'82, pages 80–91, November 1982.

45 M. Zimand. Efficient Privatization of Random Bits. In *IN WORKSHOP ON RANDOMIZED ALGORITHMS*, 1997.

A Proof of Lemma 17

Fix $n \in \mathbb{N}$ arbitrarily. Remember that $T = \max\{n^{2^c}, t_M(n)^{2^c}, t_S(n)^{2^c}\}$. Let $i_T = c^{-1}\log\log T$.

We first show that the instance $x \in \{0,1\}^n$ generated by $S^{\mathcal{O}}(1^n)$ is determined by only $\rho_{n',j}$ for $n' \leq T$ and $j \leq i_T - 1$ with probability at least $1 - O(n^{-5})$. Then, we show that $\mathcal{A}^*(M, x, 1^{T^{2^c}})$ returns $M^{\mathcal{O}}(x)$ with probability at least $1 - O(n^{-5})$ under the condition that x is determined by only $\rho_{n',j}$ for $n' \leq T$ and $j \leq i_T - 1$. If we assume these, then by the union bound, we have the lemma as

$$\Pr_{\mathcal{O}, x \leftarrow S^{\mathcal{O}}(1^n)}\left[M^{\mathcal{O}}(x) \neq \mathcal{A}^*(M, x, 1^{T^{2^c}})\right] \leq O(n^{-5}) + O(n^{-5}) = O(n^{-5}) \; (= O(n^{-4})).$$

Now, we show the first claim. Since $t_S(n) \leq T^{1/2^c}$, the answers of $\mathcal{A}$ to queries made by $S^{\mathcal{O}}(1^n)$ are determined by only $\rho_{n',j}$ for $n' \leq T^{1/2^c}$ and $j \leq i_T - 2$. Under an arbitrary condition on restrictions $\rho_{n',j}$ for $n' \leq T^{1/2^c}$ and $j \leq i_{t_S(n)^{1/2^c}} \leq i_T - 2$, the output $S^{\mathcal{O}}(1^n)$ is determined by only $\rho_{n',j}$ for $n' \leq T$ and $j \leq i_T - 1$ unless S queries $(z, x, \ell) \in \cup_{n' \leq T^{1/2^c}} \rho^*_{n', i_T - 1}{}^{-1}(*)$ to $\mathcal{F}$. Note that, if $n' \in \mathbb{N}$ satisfies $n' < t^{-1}(T^{1/2^c})$, then we have $i_{\max}(n') < c^{-1}\log\log(T^{1/2^c}) = c^{-1}\log\log T - 1 = i_T - 1$. Thus, $\rho^*_{n', i_T - 1}{}^{-1}(*) = \emptyset$. Otherwise, each element in $\rho^*_{n', i_T - 1}{}^{-1}(*)$ is selected from $V_{n', i_T - 2}$ independently with probability $p(n')$. Since $S^{\mathcal{O}}(1^n)$ accesses to $\mathcal{A}$ at most $T^{1/2^c}$ times, such a conditional probability is bounded above by

$$\begin{aligned} T^{1/2^c} \max_{t^{-1}(T^{1/2^c}) \leq n' \leq T^{1/2^c}} p(n') &= T^{1/2^c} p(t^{-1}(T^{1/2^c})) \\ &= T^{1/2^c} t(t^{-1}(T^{1/2^c}))^{-6} \\ &= (T^{1/2^c})^{-5} \\ &\leq n^{-5}, \end{aligned}$$

where the inequality holds because $T \geq n^{2^c}$.

Next, we show the second claim. Under the condition that the given instance $x \in \{0,1\}^n$ is determined by only $\rho_{n',j}$ for $n' \leq T$ and $j \leq i_T - 1$, the T-DNF formula ϕ constructed in $\mathcal{A}^*(M, x, 1^{T^{2^c}})$ is determined only by $\rho_{n',j}$ for $n' \leq T$ and $j \leq i_T - 1$. Then, applying the restriction $\rho_{n',j}$ for $n' \leq T$ and $j \leq i_T$ under this condition is regarded as a $p(n')$-random restriction to $V_{n', i_T - 1}$ for each $n' \leq T$, where we can ignore small n' such that $n' < t^{-1}(T)$ because $i_{\max}(n') < c^{-1}\log\log T = i_T$ for such n'. By applying the switching lemma (particularly, the baby switching lemma) for T-DNF [14], the probability that ϕ does not become a constant is at most

$$\begin{aligned} O\left(T \max_{t^{-1}(T) \leq n' \leq T} p(n')\right) &= O\left(T \cdot t(t^{-1}(T))^{-6}\right) \\ &= O(T^{-5}) \\ &= O(n^{-5}), \end{aligned}$$

where the last equation holds because $T \geq n^{2^c} \geq n$. Remember that $\mathcal{A}^*$ always returns the correct answer whenever ϕ becomes constant. Therefore, the second claim holds.

B Proof of Theorem 26

We first introduce the following lemma.

▶ **Lemma 31** ([11]). *Let $S, T \subseteq \{0,1\}^*$ be finite subsets of the same size N, and let $b\colon S \to T$ be a bijection. Let $A^?$ be a deterministic oracle machine that makes at most q queries to b. If $A^f(y) = f^{-1}(y)$ for all $y \in T$, then b has the representation of length at most $2\log\binom{N}{a} + \log((N-a)!)$ when A is given, where $a = N/(q+1)$.*

Now, we present the formal proof of Theorem 26.

Proof of Theorem 26. For every choice of $\mathcal{O}_a$, we define a concept class $\mathcal{C}^{\mathcal{O}_a}$ as

$$\mathcal{C}^{\mathcal{O}_a} = \{\mathcal{F}_{|z|}(z,\cdot,1) : z \in \{0,1\}^*\}.$$

Then, we show that $\mathcal{C}^{\mathcal{O}_a}$ is not weakly PAC learnable with MQ on the uniform distribution by nonuniform $t_L(n) = O(2^{(a/2)n/\log n})$-time algorithms with probability 1 over the choice of $\mathcal{O}_a$. Since $\mathcal{C}^{\mathcal{O}_a} \subseteq \mathsf{SIZE}^{\mathcal{O}_a}[n]$, the theorem also holds.

Let $\epsilon(n) = n^{-\log n}$. We fix $n \in \mathbb{N}$ arbitrarily. We consider an arbitrary nonuniform randomized oracle machine (i.e., a learner) L. For each $z \in \{0,1\}^n$, we define I_z as an event (over the choice of $\mathcal{O}$) that L succeeds in learning $f_z(x) \equiv F_n(z,x,1) \in \mathcal{C}^{\mathcal{O}}_n$ in $t_L(n)$ time with advantage $\epsilon(n)$, i.e.,

$$I_z = \left(\Pr_L\left[L^{\mathcal{O},\mathsf{MQ}_{f_z}}(n) \to h^{\mathcal{O}} \text{ s.t. } \Pr_x[h^{\mathcal{O}}(x) = f_z(x)] \geq 1/2 + \epsilon(n) \text{ in } t_L(n) \text{ time}\right] \geq 2/3\right).$$

We will show that $\Pr_{\mathcal{O}}[\wedge_{z\in\{0,1\}^n} I_z] \leq 2^{-2^{\Omega(n)}}$. For now, we assume this and show the hardness of learning $\mathcal{C}^{\mathcal{O}}$. Since any nonuniform $t_L(n)$-time oracle machine has a binary representation of length at most $O(t_L(n))$ (for each $n \in \mathbb{N}$), the event E_n that there exists a nonuniform $t_L(n)$-time oracle machine succeeds in learning $\mathcal{C}^{\mathcal{O}}_n$ is at most $2^{O(t_L(n))} \cdot 2^{-2^{\Omega(n)}} = \mathsf{negl}(n)$ by the union bound. By the Borel–Cantelli lemma, these events E_n occur only for finitely many $n \in \mathbb{N}$ with probability 1 over the choice of $\mathcal{O}$. In such cases, there is no nonuniform $t_L(n)$-time algorithm that succeeds in weak learning for $\mathcal{C}^{\mathcal{O}}$.

Now, we show that $\Pr_{\mathcal{O}}[\wedge_{z\in\{0,1\}^n} I_z] \leq 2^{-2^{\Omega(n)}}$. For any $z, x \in \{0,1\}^n$, we use a notation $L^{\mathcal{O}}(n)(x)$ to refer to the following procedure: We execute $L^{\mathcal{O},\mathsf{MQ}_{f_z}}(n)$ and if L outputs some hypothesis $h^?$ in $t_L(n)$ time, then we also execute $h^{\mathcal{O}}(x)$. For any $z \in \{0,1\}^n$, we define an event J_z as the event (over the choice of $\mathcal{O}$) that L or its hypothesis directly access a target function f_z by $\mathcal{F}$, i.e.,

$$J_z = \left(\Pr_{L,x\sim\{0,1\}^n}\left[\mathcal{F}(z,x',\ell) \text{ is queried for some } (x',\ell) \in \{0,1\}^n \times [n] \text{ during } L^{\mathcal{O}}(n)(x)\right] \geq \epsilon(n)^4\right).$$

We say that $z \in \{0,1\}^n$ is a hard index (relative to $\mathcal{O}$) if $z \in S_{n,i_{\max}(n)-1}$. Then, we have that

$$\begin{aligned}\Pr_{\mathcal{O}}\left[\bigwedge_{z\in\{0,1\}^n} I_z\right] &\leq \Pr_{\mathcal{O}}\left[\bigwedge_{z\in\{0,1\}^n:\text{hard}} I_z\right]\\ &\leq \Pr_{\mathcal{O}}\left[\bigwedge_{z\in\{0,1\}^n:\text{hard}} (I_z \vee J_z)\right]\\ &\leq \Pr_{\mathcal{O}}\left[\bigwedge_{z:\text{hard}} J_z\right] + \Pr_{\mathcal{O}}\left[\exists z : \text{hard s.t. } I_z \wedge \neg J_z\right].\end{aligned}$$

In the following, we show that each term is bounded above by $2^{-2^{\Omega(n)}}$, which implies the theorem.

▷ Claim 32. $\Pr_{\mathcal{O}}\left[\bigwedge_{z:\mathrm{hard}} J_z\right] \leq 2^{-2^{\Omega(n)}}$.

Proof. Let $N = |S_{n,i_{\max}(n)-1}|$. For any choice of $S_{n,i_{\max}(n)-1}$, we can divide a random selection of $\rho_{n,i_{\max}(n)}$ into the following two steps without loss of generality: (i) select N random functions $x_1, \dots, x_N \in \{0,1\}^{n2^n}$ uniformly at random (where we regard each x_j as a truth table of a mapping from n bits to n bits), and (ii) select a random bijection $b\colon S_{n,i_{\max}(n)-1} \to \{x_1, \dots, x_N\}$ to assign each value of $F(z,\cdot,\cdot)$ as $F(z,\cdot,\cdot) \equiv b(z)$ for each $z \in S_{n,i_{\max}(n)-1}$.

We consider an arbitrary choice of $\mathcal{O}$ except for the aforementioned bijection b and use the notation C to refer to such a partial choice of $\mathcal{O}$. We regard C as a condition on the choice of $\mathcal{O}$. We say that the partial choice C is bad if there are two distinct indices $j_1, j_2 \in [N]$ such that $x_{i_1} = x_{i_2}$. Since $x_1, \dots, x_N$ is uniformly and independently selected from 2^{n2^n} elements, by the union bound, we obtain that

$$\Pr_{\mathcal{O}}[C \text{ is bad}] \leq N^2 \cdot 2^{-n2^n} \leq 2^{2n} \cdot 2^{-n2^n} = 2^{-\Omega(2^n)}.$$

Thus, we have that

$$\begin{aligned}\Pr_{\mathcal{O}}\left[\bigwedge_{z:\mathrm{hard}} J_z\right] &= \mathrm{Exp}_C\left[\Pr_{\mathcal{O}}\left[\bigwedge_{z:\mathrm{hard}} J_z \middle| C\right]\right] \\ &\leq \mathrm{Exp}_C\left[\Pr_{\mathcal{O}}\left[\bigwedge_{z:\mathrm{hard}} J_z \middle| C\right] \middle| C \text{ is not bad}\right] + \Pr_{O}[C \text{ is bad}] \\ &\leq \mathrm{Exp}_C\left[\Pr_{b}\left[\bigwedge_{z:\mathrm{hard}} J_z \middle| C\right] \middle| C \text{ is not bad}\right] + 2^{-\Omega(2^n)}.\end{aligned}$$

Therefore, it is sufficient to show that for every not bad condition C,

$$\Pr_{b}\left[\bigwedge_{z:\mathrm{hard}} J_z \middle| C\right] = 2^{-2^{\Omega(n)}}.$$

To show the aforementioned bound, we assume that $\bigwedge_{z:\mathrm{hard}} J_z$ holds under a not bad condition C (notice that C determines all hard indices), i.e., for any hard index $z \in \{0,1\}^n$,

$$\Pr_{L,x\sim\{0,1\}^n}\left[\mathcal{F}(z,\cdot,\cdot) \text{ is queried during } L^{\mathcal{O}}(n)(x)\right] \geq \epsilon(n)^4$$

By the standard probabilistic argument, we can reduce the upper bound from $1-\epsilon(n)^4$ to 2^{-2n} on the probability that $\mathcal{F}(z,\cdot,\cdot)$ is not queried by L or its hypothesis by repeating $L^{\mathcal{O}}(n)(x)$ $2n/\epsilon(n)^4$ times. Then, by the union bound for all hard indices, there exists a random seed $r \in \{0,1\}^{t_L(n)\cdot 2n/\epsilon(n)^4}$ such that for any hard index z, $\mathcal{F}(z,\cdot,\cdot)$ is queried during at least one execution of $L^{\mathcal{O}}(n)(x)$ by using the randomness r. We remark that all queries to $\mathcal{O}$ by L or its hypothesis are determined by C except for $\mathcal{F}(z,\cdot,\cdot)$ for each hard index z. This is because L and its hypothesis are executed in time $O(t_L(n)) = O(2^{(a/2)n/\log n}) \leq 2^{an/\log n} = t(n)$ (for sufficiently large n), i.e., all answers from $\mathcal{A}$ depend on only $\rho_{\cdot,j}$ for $j \leq c^{-1}\log\log t(n)^{1/2^c} = c^{-1}\log\log t(n) - 1 < i_{\max}(n)$.

Therefore, by executing L with the randomness r and tracing queries to $\mathcal{F}$, we can obtain a deterministic inverter for b of the query complexity at most $t_L(n)\cdot 2n/\epsilon(n)^4$, where the inverter simulates membership queries by using its input and its own query access to b. Particularly, the inverter accesses b only for answering the queries of the form $\mathcal{F}(z',\cdot,\cdot)$

for some $z' \in S_{n,i_{\max}(n)-1}$. However, by Lemma 31, such a bijection b is represented by $2\log\binom{N}{a} + \log((N-a)!)$ bits, where $a = N/(t_L(n) \cdot 2n\epsilon(n)^{-4} + 1)$, when L and r are given. Thus, we obtain that

$$a \geq \frac{2^n \cdot p(n)^{i_{\max}(n)}}{O(t_L(n)n\epsilon(n)^{-4})} \geq \frac{2^n \cdot 2^{-\frac{6an}{\log n}\cdot\frac{1}{c}\log n}}{O(2^{(a/2)n/\log n}n^{4\log n+1})} \geq \frac{2^n \cdot 2^{-\frac{6}{7}n}}{2^{O(n/\log n)}} \geq 2^{\Omega(n)},$$

and

$$\begin{aligned}\Pr_b\left[\bigwedge_{z:\text{hard}} J_z \middle| C\right] &\leq \frac{\binom{N}{a}^2 \cdot (N-a)! \cdot 2^{O(t_L(n)\cdot 2n\epsilon(n)^{-4})}}{N!} \\ &\leq \binom{N}{a} \cdot \frac{1}{a!} \cdot 2^{O(2^{(a/2)n/\log n})} \\ &\leq \left(\frac{Ne}{a}\right)^a \cdot \frac{1}{\sqrt{2\pi a}}\left(\frac{e}{a}\right)^a \cdot 2^{2^{O(n/\log n)}} \\ &\leq \left(e(t_L(n)\cdot 2n\epsilon(n)^{-4}+1)\right)^a \cdot \left(\frac{e}{a}\right)^a \cdot 2^{2^{O(n/\log n)}} \\ &\leq \left(\frac{2^{O(n/\log n)}}{a}\right)^a \cdot 2^{2^{O(n/\log n)}} \\ &\leq 2^{-a} \cdot 2^{2^{O(n/\log n)}} = 2^{-2^{\Omega(n)}}.\end{aligned}$$

◁

▷ Claim 33. $\Pr_{\mathcal{O}}[\exists z : \text{hard s.t. } I_z \wedge \neg J_z] \leq 2^{-2^{\Omega(n)}}$.

Proof. We consider an arbitrary choice of $\mathcal{O}$ except for values of $\rho_{n,i_{\max}(n)}$ (we write this condition as C). Note that hard indices are determined by the condition C, and for any hard index $z \in \{0,1\}^n$, f_z is a random function even under the condition C.

Suppose that z is a hard index, and $\neg I_z \wedge J_z$ occurs. By Markov's inequality, we derive the following from $\neg I_z$:

$$\Pr_L\left[\Pr_x\left[\mathcal{F}(z,\cdot,\cdot) \text{ is queried during } L^{\mathcal{O},\mathsf{MQ}_{f_z}}(n)(x)\right] \leq 4\epsilon(n)^3\right] \geq 1 - \epsilon(n)/4.$$

Since J_z holds, we also have

$$\Pr_L\left[L^{\mathcal{O},\mathsf{MQ}_{f_z}}(n) \to h^{\mathcal{O}} \text{ s.t. } \Pr_x[h^{\mathcal{O}}(x) = f_z(x)] \geq 1/2 + \epsilon(n) \text{ in } t_L(n) \text{ time}\right] \geq \frac{2}{3}.$$

From the aforementioned two inequalities, there exists a random string r for L such that

- $L^{\mathcal{O},\mathsf{MQ}_{f_z}}(n; r)$ outputs some hypothesis $h^{\mathcal{O}}$ in $t_L(n)$ time without querying $(z,\cdot,\cdot)$ to $\mathcal{F}$;
- $\Pr_x[h^{\mathcal{O}}(x) \text{ queries } (z,\cdot,\cdot) \text{ to } \mathcal{F}] \leq 4\epsilon(n)^3$; and
- $\Pr_x[h^{\mathcal{O}}(x) = f_z(x)] \geq 1/2 + \epsilon(n)$.

Since L and h are only executed in $O(t_L(n)) \leq t(n)$ time (for sufficiently large n), any query $(M, x', 1^{T^{2^c}})$ to $\mathcal{A}$ by L and h satisfies that $T^{2^c} \leq t(n)$ and $i_T = c^{-1}\log\log T = c^{-1}\log\log t(n) - 1 < i_{\max}(n)$. Therefore, if L and h do not query $(z,\cdot,\cdot)$ to $\mathcal{F}$, then the answers from $\mathcal{O}$ do not depend on $\rho_{n,i_{\max}(n)}$, i.e., they are determined only by the condition C.

In this case, we show that a truth table of f_z has a short description under the condition C. This implies the upper bound on the probability of this case because a random function does not have such a short description with extremely high probability.

The short description of f_z is obtained as follows. Let $B_z \subseteq \{0,1\}^n$ be the subset consisting of x such that $h^{\mathcal{O}}(x)$ queries $\mathcal{F}(z,\cdot,\cdot)$. By the second property of the above, $|B_z| \leq 2^n \cdot 4\epsilon(n)^3$ holds. We execute $L^{\mathcal{O},\mathsf{MQ}_{f_z}}(n; r)$ to obtain $h^{\mathcal{O}}$, and we write down all answers from the membership query oracle MQ_{f_z} in Q, i.e., Q is a binary string of length at most $t_L(n)$. By the first property on r, the answers from $\mathcal{O}$ are determined by the condition C. Next, we execute the outputted hypothesis $h^{\mathcal{O}}(x)$ on each input $x \in \{0,1\}^n \setminus B_z$. From these predictions and auxiliary information $f_z(B_z) = \{(x, f_z(x)) : x \in B_z\}$, we obtain a function $\tilde{f} : \{0,1\}^n \to \{0,1\}$ defined as

$$\tilde{f}(x) = \begin{cases} h^{\mathcal{O}}(x) & \text{if } x \notin B_z \\ f_z(x) & \text{if } x \in B_z. \end{cases}$$

Then, by the third property, $\tilde{f}$ is $(1/2 - \epsilon(n))$-close to f_z. We define $e \in \{0,1\}^{2^n}$ as $e_{x+1} = f_z(x) \oplus \tilde{f}(x)$, where we identify $x \in \{0,1\}^n$ with an integer in $[0, 2^n - 1]$. Then, the Hamming weight of e is at most $2^n \cdot (1/2 - \epsilon(n))$, and e is represented by a binary string $\tilde{e}$ of length at most $(1 - \Omega(\epsilon(n)^2)) \cdot 2^n$ by lexicographic indexing among binary strings of the same weight. Obviously, f_z is reconstructed from $\tilde{f}$ and $\tilde{e}$. Therefore, based on the aforementioned constructions, f_z is represented only by $L, r, Q, f_z(B_z)$, and $\tilde{e}$ on the condition C. The total number of such representations is at most

$$\begin{aligned} &|L| + t_L(n) + t_L(n) + (n+1) \cdot |B_z| + (1 - \Omega(\epsilon(n)^2)) \cdot 2^n \\ &\qquad \le O(t_L(n)) + \left(1 + 4(n+1)\epsilon(n)^3 - \Omega(\epsilon(n)^2)\right) 2^n \\ &\qquad \le O(t_L(n)) + \left(1 - \Omega(\epsilon(n)^2)\right) \cdot 2^n. \end{aligned}$$

Therefore, we have that for any condition C and any hard index $z \in \{0,1\}^n$,

$$\begin{aligned} \Pr_{\mathcal{O}}\left[I_z \wedge \neg J_z | C\right] &\le \frac{2^{O(t_L(n)) + \left(1 - \Omega(\epsilon(n)^2)\right) \cdot 2^n}}{2^{2^n}} \\ &\le 2^{2^{O(n/\log n)} - \Omega(n^{-2\log n}) \cdot 2^n} \\ &\le 2^{-2^{\Omega(n)}}. \end{aligned}$$

Thus, we conclude that

$$\begin{aligned} \Pr_{\mathcal{O}}\left[\bigvee_{z:\text{hard}} I_z \wedge \neg J_z\right] &= \mathrm{Exp}_C\left[\Pr_{\mathcal{O}}\left[\bigvee_{z:\text{hard}} I_z \wedge \neg J_z \middle| C\right]\right] \\ &\le \mathrm{Exp}_C\left[\sum_{z \in \{0,1\}^n} \Pr_{\mathcal{O}}\left[z \text{ is hard and } I_z \wedge \neg J_z | C\right]\right] \\ &\le 2^n \cdot 2^{-2^{\Omega(n)}} = 2^{-2^{\Omega(n)}}. \end{aligned}$$

◁

◀