

Quantum Pessiland

Boyang Chen¹, Tomoyuki Morimae², and Takashi Yamakawa^{3,2}

¹Department of Computer Science and Technology, Tsinghua University, Beijing, China
by-chen24@mails.tsinghua.edu.cn

²Yukawa Institute for Theoretical Physics, Kyoto University, Kyoto, Japan
tomoyuki.morimae@yukawa.kyoto-u.ac.jp

³NTT Social Informatics Laboratories, Tokyo, Japan
takashi.yamakawa@ntt.com

Abstract

Pessiland is a world where NP is hard on average but one-way functions (OWFs) do not exist [Impagliazzo 1995]. Because almost all classical cryptographic primitives imply OWFs [Impagliazzo and Luby 1989], there is almost no classical cryptography in Pessiland. On the other hand, quantum cryptography can exist even when OWFs do not [Kretschmer 2021; Morimae and Yamakawa 2022; Ananth, Qian and Yuen 2022]. Is there a quantum analogue of Pessiland where NP is hard on average but even quantum cryptography does not exist? In this paper, we show that such a miserable world, *Quantum Pessiland*, exists: there is a quantum oracle relative to which $UP \cap coUP$ is hard on average against quantum polynomial-time algorithms with quantum advice, yet auxiliary-input EFI pairs do not exist. We also show that there is a classical oracle relative to which $UP \cap coUP$ is hard on average against quantum polynomial-time algorithms with quantum advice, yet classically-secure auxiliary-input one-way puzzles (OWPuzzs) do not exist. Almost all quantum cryptographic primitives imply EFI pairs or OWPuzzs, and therefore these results mean that there is almost no quantum cryptography relative to these oracles. We further show that relative to the classical oracle, $SampBQP = SampBPP$, and therefore there is no sampling-based quantum advantage in Quantum Pessiland. Finally, because our average-case hardness of $UP \cap coUP$ implies $P^{\#P} \not\subseteq i.o.BQP/qpoly$, our result also implies that a non-relativizing proof technique is necessary to construct OWPuzzs solely from $P^{\#P} \not\subseteq i.o.BQP/qpoly$, which gives a partial negative answer to the open problem of [Khurana and Tomer 2025].

Contents

1	Introduction	3
1.1	Main Results	3
1.2	Implications	4
1.3	Technical Overview	5
1.4	Related Works	8
2	Preliminaries	9
2.1	Basic Notations	9
2.2	Average-Case Decision and Search Hardness	10
2.3	Auxiliary-Input Primitives	11
2.4	Sampling Complexity	12
3	Oracle	12
4	Average-Case Hard Problems	13
5	Patching Lemma	17
6	Proof of Theorem 1.1	22
7	Proof of Theorem 1.2	25

1 Introduction

Pessiland [Imp95] is a world where NP is hard on average but one-way functions (OWFs) do not exist. Because almost all classical cryptographic primitives imply OWFs [IL89], Pessiland is a miserable world where essentially no classical cryptography exists even though NP is hard on average.

In quantum cryptography, by contrast, OWFs are not necessarily the minimal assumption [Kre21, MY22, AQY22]. Several quantum analogues of OWFs, pseudorandom generators (PRGs), and pseudorandom functions (PRFs) have been introduced, such as pseudorandom unitaries (PRUs) [JLS18], pseudorandom state generators (PRSGs) [JLS18], one-way state generators (OWSGs) [MY22], one-way puzzles (OWPuzzs) [KT24], and EFI pairs [BCQ23]. They are potentially weaker than OWFs [Kre21, KQST23, KQT25, LMW24], yet they still suffice for many applications, including private-key quantum money [JLS18], secret-key encryption (SKE) [AQY22], digital signatures [MY22], commitments [MY22, AQY22, BCQ23], and multiparty computation [MY22, AQY22]. The fact that quantum cryptography may exist even without OWFs raises a natural question:

Is there “Quantum Pessiland” where NP is hard on average but even quantum cryptography does not exist?

1.1 Main Results

In this paper, we show that the answer is yes. We prove the following two theorems.

Theorem 1.1. *There exists a classical oracle $\mathcal{O}$ relative to which the following statements hold.*

1. *There exists a language $L^{\mathcal{O}} \in \text{UP}^{\mathcal{O}} \cap \text{coUP}^{\mathcal{O}}$ that is strongly hard on average against quantum polynomial-time algorithms that can query $\mathcal{O}$ and receive polynomial-size oracle-dependent quantum advice.¹*
2. *There exists a relation $R^{\mathcal{O}} \in \text{TFUP}^{\mathcal{O}}$ that is average-case hard against quantum polynomial-time algorithms that can query $\mathcal{O}$ and receive polynomial-size oracle-dependent quantum advice.*
3. *$\text{SampBQP}^{\mathcal{O}} = \text{SampBPP}^{\mathcal{O}}$. (And therefore $\text{BQP}^{\mathcal{O}} = \text{BPP}^{\mathcal{O}}$.)*
4. *Classically-secure auxiliary-input OWPuzzs do not exist.*

Theorem 1.2. *There exists a quantum unitary oracle $\mathcal{U}$ and a classical oracle $\mathcal{V}$ relative to which the following statements hold.*

1. *There exists a language $L^{\mathcal{V}} \in \text{UP}^{\mathcal{V}} \cap \text{coUP}^{\mathcal{V}}$ that is strongly hard on average against quantum polynomial-time algorithms that can query $(\mathcal{V}, \mathcal{U})$ and receive polynomial-size oracle-dependent quantum advice.*
2. *There exists a relation $R^{\mathcal{V}} \in \text{TFUP}^{\mathcal{V}}$ that is average-case hard against quantum polynomial-time algorithms that can query $(\mathcal{V}, \mathcal{U})$ and receive polynomial-size oracle-dependent quantum advice.*
3. *Auxiliary-input EFI pairs do not exist relative to $(\mathcal{V}, \mathcal{U})$.*

¹Because $\text{UP} \cap \text{coUP} \subseteq \text{NP} \cap \text{coNP}$, the same result also holds for $\text{NP} \cap \text{coNP}$.

An EFI pair (generator) [BCQ23] is a quantum polynomial-time (QPT) algorithm $\text{Gen}(1^\lambda, b) \rightarrow \rho_{\lambda,b}$ that takes the security parameter 1^λ and a bit $b \in \{0, 1\}$ as input and outputs a quantum state $\rho_{\lambda,b}$ such that $\{\rho_{\lambda,0}\}_{\lambda \in \mathbb{N}}$ and $\{\rho_{\lambda,1}\}_{\lambda \in \mathbb{N}}$ are statistically far but computationally indistinguishable. Almost all classical/quantum cryptographic primitives imply EFI pairs. Moreover, EFI pairs are existentially equivalent to quantum non-interactive bit commitments and multi-party computations [BCQ23].

Auxiliary-input EFI pairs are a weaker variant of EFI pairs. An auxiliary-input EFI pair is a uniform QPT algorithm $\text{Gen}(x, b) \rightarrow \xi_{b,x}$ that takes a classical auxiliary input $x \in \{0, 1\}^*$ and a bit $b \in \{0, 1\}$ as input and outputs a quantum state $\xi_{b,x}$ such that there exists an infinite set $S \subseteq \{0, 1\}^*$ on which $\{\xi_{0,x}\}_{x \in S}$ and $\{\xi_{1,x}\}_{x \in S}$ are statistically far but computationally indistinguishable. (For the precise definition, see Definition 2.7.) Therefore, Theorem 1.2 also rules out the standard EFI pairs. Almost all classical/quantum cryptographic primitives imply EFI pairs, and therefore non-existence of EFI pairs means that there is almost no classical/quantum cryptography.

OWPuzzs [KT24] are a quantum analogue of OWFs. A OWPuzz is a pair $(\text{Samp}, \text{Ver})$ of two algorithms. $\text{Samp}(1^\lambda) \rightarrow (\text{puzz}, \text{ans})$ is a QPT algorithm that takes the security parameter 1^λ as input and outputs two classical bit strings puzz and ans . $\text{Ver}(\text{puzz}, \text{ans}') \rightarrow \top/\perp$ is a not-necessarily-efficient algorithm that takes puzz and a classical bit string ans' and outputs $\top$ or $\perp$. The correctness requires that

$$\Pr[\top \leftarrow \text{Ver}(\text{puzz}, \text{ans}) : (\text{puzz}, \text{ans}) \leftarrow \text{Samp}(1^\lambda)] \geq 1 - \text{negl}(\lambda) \quad (1)$$

and the security requires that

$$\Pr[\top \leftarrow \text{Ver}(\text{puzz}, \text{ans}') : (\text{puzz}, \text{ans}) \leftarrow \text{Samp}(1^\lambda), \text{ans}' \leftarrow \mathcal{A}(1^\lambda, \text{puzz})] \leq \text{negl}(\lambda) \quad (2)$$

for any QPT adversary $\mathcal{A}$. We say that a OWPuzz is classically secure if the security is required to hold only against probabilistic polynomial-time (PPT) adversaries. Almost all classical/quantum cryptographic primitives imply OWPuzzs, and OWPuzzs imply EFI pairs [KT24].

Auxiliary-input OWPuzzs are a weaker variant of OWPuzzs. An auxiliary-input OWPuzz is a pair $(\text{Samp}, \text{Ver})$ of algorithms, where $\text{Samp}(x) \rightarrow (\text{puzz}, \text{ans})$ is a uniform QPT algorithm that takes a classical auxiliary input $x \in \{0, 1\}^*$, and $\text{Ver}(x, \text{puzz}, \text{ans}) \rightarrow \top/\perp$ is a not-necessarily-efficient algorithm. The security is required only on an infinite set of auxiliary inputs that may depend on the adversary: for every adversary $\mathcal{A}$, there exists an infinite set $I_{\mathcal{A}} \subseteq \{0, 1\}^*$ such that for every $x \in I_{\mathcal{A}}$, $\mathcal{A}$ finds a valid answer with only negligible probability. (For the precise definition, see Definition 2.4.) OWPuzzs imply classically-secure auxiliary-input OWPuzzs, and therefore Theorem 1.1 also rules out the standard OWPuzzs. Most classical/quantum cryptographic primitives imply OWPuzzs. Hence Theorem 1.1 also rules out most classical/quantum cryptographic primitives.

1.2 Implications

Because the average-case hardness of $\text{UP} \cap \text{coUP}$ in Theorem 1.1 implies that of PP , we also have the worst-case separation $\text{PP} \not\subseteq \text{i.o.BQP/qpoly}$. Moreover, $\text{PP} \subseteq \text{P}^{\#P}$. Therefore Theorem 1.1 gives the following corollary.

Corollary 1.3. *There exists a classical oracle relative to which PP is strongly hard on average against QPT algorithms with quantum advice (and therefore $\text{P}^{\#P} \not\subseteq \text{i.o.BQP/qpoly}$) but OWPuzzs do not exist.*

Khurana and Tomer [KT25] constructed OWPuzzs from $\text{P}^{\#P} \not\subseteq \text{i.o.BQP/qpoly}$ and an assumption that implies sampling-based quantum advantage (the so-called “average-case $\#P$ hardness assumption” [AA11, BMS16]). Whether OWPuzzs can be constructed solely from the average-case hardness of PP or the

worst-case separation $PP \neq BQP$ has remained open. Corollary 1.3 suggests that a non-relativizing technique would be required to resolve the open problem.

Morimae, Shirakawa, and Yamakawa [MSY25] showed the equivalence between classically-secure OW-Puzzs and inefficient-verifier proofs of quantumness (IV-PoQ), which are interactive protocols demonstrating quantum advantage [MY24].² Theorem 1.1 therefore also gives the following corollary, which suggests that IV-PoQ cannot be constructed in a black-box way from the average-case hardness of $UP \cap coUP$.

Corollary 1.4. *There exists a classical oracle relative to which $UP \cap coUP$ is strongly hard on average against QPT algorithms with quantum advice, but IV-PoQ do not exist.*

PoQ imply IV-PoQ, and therefore the above corollary also suggests that PoQ do not exist relative to the oracle. No relation is known between IV-PoQ and $SampBQP \neq SampBPP$. However, Theorem 1.1 also shows $SampBQP^{\mathcal{O}} = SampBPP^{\mathcal{O}}$ relative to a classical oracle $\mathcal{O}$. Therefore, our results indicate that there is essentially no quantum advantage in Quantum Pessiland.

1.3 Technical Overview

Our starting point is the classical construction of Pessiland by Wee [Wee06] (building on an unpublished result of Impagliazzo and Rudich). Let $\pi = (\pi_1, \pi_2, \dots)$, where each π_ℓ is a uniformly random permutation over $\{0, 1\}^\ell$, and let V^π be the *verification* oracle

$$V^\pi(\ell, u, v) := \begin{cases} 1 & |u| = |v| = \ell \text{ and } \pi_\ell(u) = v, \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

Let QBF be an oracle that solves any PSPACE-complete problem. Consider the oracle

$$\mathcal{O} := (V^\pi, QBF). \quad (4)$$

[Wee06] showed that, relative to $\mathcal{O}$, there is a language in $NP \cap coNP$ (in fact $UP \cap coUP$) that is hard on average, but OWFs do not exist.

The result is shown as follows. First, the inverse $\pi_n^{-1}(y)$ is information-theoretically determined by V^π but hard to find: an algorithm making polynomially many queries to V^π essentially never receives the answer 1 on a fresh query, so the oracle it sees is indistinguishable from the trivial oracle Z that always outputs zero. This yields an average-case hard language in $NP \cap coNP$ (in fact $UP \cap coUP$) via the Goldreich–Levin predicate $\langle \pi_n^{-1}(y), r \rangle$. On the other hand, the same observation shows that no candidate OWF $f^{V^\pi, QBF}$ can effectively use V^π , and therefore $f^{V^\pi, QBF}$ can be simulated by $f^{Z, QBF}$ and the latter can be inverted in polynomial time with the help of the QBF oracle.

Our goal is to carry out this program in the quantum setting, where the candidate primitives are quantum algorithms that query the oracles in superposition, where the hardness must hold even against QPT algorithms with quantum advice, and where the primitives to be ruled out are the *auxiliary-input* versions of OWPuzzs and EFI pairs.³ The hardness side follows from a recent tight bound for quantum permutation inversion with quantum advice [ABC⁺26] combined with Goldreich–Levin with quantum advice [KT24, AC02]. In this overview we focus on the non-existence side, where the main technical novelty of this paper lies.

²In [MSY25], the security of OWPuzzs only requires $1 - 1/poly$ security (see their Remark 2.15 and 2.16). However, our proof breaks even such weak OWPuzzs.

³[Wee06] also ruled out auxiliary-input OWFs, but his technique cannot be used for our quantum setting. For details, see Section 1.4.

The non-auxiliary-input case is easy. First, let us consider the easy case, namely, ruling out *standard* (not auxiliary-input) primitives. In this case, we can follow the template of [Wee06] which we have explained above. Let us consider OWPuzzs. To break OWPuzzs, we have only to break a distributional OWPuzz, $\text{Samp}^{V^\pi, \text{QBF}}$. The only input of Samp is 1^λ , which is independent of π . Hence V^π can simply be replaced by a trivial oracle Z that always outputs 0. More precisely, we can easily show

$$\mathbb{E}_\pi \text{SD} \left(\text{Samp}^{V^\pi, \text{QBF}}(1^\lambda), \text{Samp}^{Z, \text{QBF}}(1^\lambda) \right) \leq \frac{1}{p_1(\lambda)} \quad (5)$$

for any large polynomial p_1 . Take any polynomial r . Then, taking $p_1(\lambda) := r(\lambda)\lambda^2$, and applying Markov's inequality,

$$\Pr_\pi \left[\text{SD} \left(\text{Samp}^{V^\pi, \text{QBF}}(1^\lambda), \text{Samp}^{Z, \text{QBF}}(1^\lambda) \right) \geq \frac{1}{r(\lambda)} \right] \leq r(\lambda) \times \frac{1}{p_1(\lambda)} = \frac{1}{\lambda^2}. \quad (6)$$

Because the right-hand side is summable, from the Borel–Cantelli lemma, with probability one over π ,

$$\text{SD} \left(\text{Samp}^{V^\pi, \text{QBF}}(1^\lambda), \text{Samp}^{Z, \text{QBF}}(1^\lambda) \right) \leq \frac{1}{r(\lambda)} \quad (7)$$

for all sufficiently large λ . Fix any such π . We can easily construct a PPT algorithm $\mathcal{A}$ querying QBF such that

$$\text{SD} \left((\text{puzz}, \mathcal{A}^{\text{QBF}}(1^\lambda, \text{puzz}))_{\text{puzz} \leftarrow \text{Samp}^{Z, \text{QBF}}(1^\lambda)}, (\text{puzz}, \text{ans})_{(\text{puzz}, \text{ans}) \leftarrow \text{Samp}^{Z, \text{QBF}}(1^\lambda)} \right) \leq \text{negl}(\lambda). \quad (8)$$

Combining Equations (7) and (8) breaks the distributional OWPuzzs.

Why auxiliary inputs break this argument. The above argument does not work for auxiliary-input OWPuzzs, because of the following two reasons.

First, the auxiliary input x can depend on π , and therefore x could contain (ℓ, u, v) such that $V^\pi(\ell, u, v) = 1$. In other words, x can “tell” Samp some positive points of V^π . Then we can no longer replace V^π with the trivial oracle Z . To solve the issue, we have to find a “learning” QPT algorithm $\mathcal{E}^{V^\pi, \text{QBF}}$ that takes x as input, runs $\text{Samp}^{V^\pi, \text{QBF}}(x)$ several times, and outputs a “patch” Γ , which is a set of polynomially many (ℓ, u, v) , such that

$$\mathbb{E}_\Gamma \text{SD} \left(\text{Samp}^{V^\pi, \text{QBF}}(x), \text{Samp}^{\Gamma, \text{QBF}}(x) \right) \leq \frac{1}{\text{poly}(|x|)}, \quad (9)$$

where $\text{Samp}^{\Gamma, \text{QBF}}$ is the same as $\text{Samp}^{V^\pi, \text{QBF}}$ except that when the base algorithm queries (ℓ, u, v) to V^π , it queries the oracle Γ that returns 1 if $(\ell, u, v) \in \Gamma$ and 0 otherwise.

Second, to break the auxiliary-input security, for a fixed π , Equation (9) has to be satisfied *for every sufficiently long* x .

The patching lemma. Our main technical tool, the patching lemma (Lemma 5.2), overcomes both obstructions. It states that, with probability one over π , for every uniform QPT algorithm $\mathcal{A}^{V^\pi, \text{QBF}}(x)$ which takes $x \in \{0, 1\}^n$ as input and outputs a quantum state, and every accuracy polynomial r , there is a uniform QPT extractor $\mathcal{E}$ that receives x and outputs a polynomial-length patch Γ such that, for all sufficiently large n and *simultaneously for every* $x \in \{0, 1\}^n$,

$$\mathbb{E}_\Gamma \left[\text{TD} \left(\mathcal{A}^{V^\pi, \text{QBF}}(x), \mathcal{A}^{\Gamma, \text{QBF}}(x) \right) \right] \leq \frac{1}{r(n)}. \quad (10)$$

The extractor $\mathcal{E}$ addresses the first obstruction in the natural way. First, it classically queries V^π for all (ℓ, u, v) such that $\ell \leq m$ with $m = O(\log n)$, and constructs the complete truth table Γ_0 of V^π for all “short” (u, v) . Because $m = O(\log n)$, this is possible in polynomial time. Next, $\mathcal{E}$ runs $\mathcal{A}^{\Gamma_0, \text{QBF}}$ up to the i th query with random $i \in [T]$, where T is the number of queries to V^π made by $\mathcal{A}$, and measures the query register to get (ℓ, u, v) . $\mathcal{E}$ queries (ℓ, u, v) to V^π , and if the result is 1, (ℓ, u, v) is added to Γ_0 . In other words, $\Gamma_1 := \Gamma_0 \cup \{(\ell, u, v)\}$. $\mathcal{E}$ repeats this R times with an appropriate polynomial R , and gets the list $(\Gamma_0, \Gamma_1, \dots, \Gamma_R)$. Finally $\mathcal{E}$ outputs $\Gamma := \Gamma_J$ with random $J \leftarrow \{0, 1, \dots, R-1\}$. A BBBV-style hybrid [BBBV97] shows that

$$\mathbb{E}_\Gamma \left[\text{TD} \left(\mathcal{A}^{V^\pi, \text{QBF}}(x), \mathcal{A}^{\Gamma, \text{QBF}}(x) \right) \right] \leq 2T \sqrt{\frac{\kappa_\pi(x)}{R}}, \quad (11)$$

where $\kappa_\pi(x)$ is the expected number of stages at which the extractor discovers a new positive triple, i.e., adds an element to the patch.

The key observation is that each stage makes a discovery with probability at most $1/R$, no matter what has happened before: conditioned on the entire transcript of the extraction algorithm $\mathcal{E}$, the unrevealed part of each long block π_ℓ with $\ell \geq m$ is a uniformly random perfect matching consistent with the previously observed answers, and hence any fixed fresh triple is a true positive (i.e., a triple (ℓ, u, v) such that $V^\pi(\ell, u, v) = 1$) with probability at most $1/(2^m - R) \leq 1/R$, where the parameters are chosen so that $2^m \geq 2R$. Hence, $\mathbb{E}_\pi[\kappa_\pi(x)] \leq R \times \frac{1}{R} = 1$. Averaging over π on both sides of Equation (11), we have

$$\mathbb{E}_\pi \mathbb{E}_\Gamma \left[\text{TD} \left(\mathcal{A}^{V^\pi, \text{QBF}}(x), \mathcal{A}^{\Gamma, \text{QBF}}(x) \right) \right] \leq \mathbb{E}_\pi 2T \sqrt{\frac{\kappa_\pi(x)}{R}} \leq \frac{2T}{\sqrt{R}}. \quad (12)$$

Hence by Markov’s inequality, we obtain that

$$\mathbb{E}_\Gamma \left[\text{TD} \left(\mathcal{A}^{V^\pi, \text{QBF}}(x), \mathcal{A}^{\Gamma, \text{QBF}}(x) \right) \right] \leq \frac{1}{\text{poly}(n)} \quad (13)$$

except with probability δ , which is an inverse-polynomial, over the choice of π .

This is, however, far from sufficient: the security of auxiliary-input primitives only requires an infinite set of hard auxiliary inputs, so to break them we must succeed, for one fixed π , on all but finitely many x . Hence a union bound over the 2^n inputs of length n would require δ to be $2^{-\omega(n)}$ for each x , which Markov’s inequality cannot provide.

Solution. The exponential moment saves us. Since every stage makes a discovery with probability at most $1/R$ regardless of the transcript, the number K of discoveries during the R stages satisfies

$$\mathbb{E} \left[2^K \right] \leq \left(1 + \frac{1}{R} \right)^R \leq e, \quad (14)$$

where the expectation is over π and the internal randomness of the extraction algorithm $\mathcal{E}$, and, crucially, the bound is uniform in x . By Jensen’s inequality the same bound holds with K replaced by $\kappa_\pi(x)$, and bounding the maximum by the log-sum-exp,

$$\mathbb{E}_\pi \left[\max_{x \in \{0,1\}^n} \kappa_\pi(x) \right] \leq \log_2 \sum_{x \in \{0,1\}^n} \mathbb{E}_\pi \left[2^{\kappa_\pi(x)} \right] \leq \log_2 (2^n e) \leq n + 2. \quad (15)$$

In words, even an adversarially chosen input cannot force the extractor $\mathcal{E}$ to make more than roughly n discoveries on average: each discovery is a $1/R$ -unlikely event, so accumulating many more than n of them has probability far below 2^{-n} , and this is what beats the union bound over the 2^n inputs.

Choosing R to be a sufficiently large polynomial in n , T , and the accuracy polynomial r , we obtain $2T\sqrt{(n+2)/R} \leq 1/(2r(n)n^2)$, so that $\mathbb{E}_\pi \mathbb{E}_\Gamma [\text{TD}(\mathcal{A}^{V^\pi, \text{QBF}}(x), \mathcal{A}^{\Gamma, \text{QBF}}(x))]$ is bounded by $1/2r(n)n^2$ even for the worst x . Markov's inequality over π and the Borel–Cantelli lemma yield the final result that Equation (10) is true with probability one over π and for all x .

Ruling out EFI pairs. Breaking EFI pairs can be done in a similar way. However, instead of the QBF oracle, we have to use a quantum oracle, which we call the Helstrom oracle, because it is not known whether classical oracle is enough to break EFI pairs [LMW24]. The Helstrom oracle receives classical descriptions of two quantum circuits C_0 and C_1 together with a challenge state, and the oracle applies to the challenge state the optimal two-outcome (Helstrom) measurement that distinguishes the outputs of C_0 and C_1 . (Actually, we consider a unitary version of it so that our oracle is a unitary oracle.) Because the Helstrom measurement for two states generated by polynomial-size circuits can be implemented in quantum polynomial space up to an exponentially small error, the same proof goes through when the Helstrom oracle is replaced with a unitaryPSPACE oracle.

1.4 Related Works

Comparison with Aaronson and Chen [AC17]. Aaronson and Chen [AC17] constructed a classical oracle relative to which $\text{SampBQP} = \text{SampBPP}$ and yet the polynomial-time hierarchy is infinite. This implies that in order to show

$$\text{Polynomial-time hierarchy is infinite} \Rightarrow \text{SampBQP} \neq \text{SampBPP}, \quad (16)$$

which is a long-standing open problem in the field of quantum advantage, a non-relativizing technique is required. (Currently what is known is that the infiniteness of the polynomial-time hierarchy plus some ad hoc assumptions imply $\text{SampBQP} \neq \text{SampBPP}$ [AA11, BMS16], and to remove the latter is an open problem.)

Our Theorem 1.1 is of the same flavor: relative to a classical oracle there is no sampling-based quantum advantage, while strong average-case hardness survives. However, the two results strengthen the hardness side along incomparable axes. The hardness in [AC17] is a worst-case structural statement about the entire hierarchy, whereas ours concerns a single language: it is average-case rather than worst-case, it holds against QPT adversaries with polynomial-size quantum advice rather than against classical algorithms, and the hard language lies in $\text{UP} \cap \text{coUP}$, a subclass of NP. In these three respects our hardness statement is stronger. Moreover, our oracle rules out not only sampling-based quantum advantage but also inefficient-verifier proofs of quantumness. On the other hand, since our hard language lies in $\text{UP} \cap \text{coUP}$, our result says nothing about separating the levels of the hierarchy, and we do not know whether the polynomial-time hierarchy is infinite relative to our oracle; in this respect their result is stronger, making the two results incomparable overall.

Comparison with Wee [Wee06]. The second result of Wee [Wee06] is the construction of a relativized Pessiland that rules out auxiliary-input OWFs, and our patching lemma (Lemma 5.2) is similar in spirit to the approximation lemma at the core of his proof. Relative to an oracle consisting of the verification oracle of a uniformly random function f together with a PSPACE oracle, Wee showed that auxiliary-input OWFs do not exist, as follows. Since an auxiliary-input OWF is computed by a family of polynomial-size oracle circuits, it suffices to show that every oracle circuit $\mathcal{C}$ of polynomial size admits a small set Q of inputs to f , depending

on both $\mathcal{C}$ and f , such that hardwiring the values of f on Q into $\mathcal{C}$ and answering every query outside Q by 0 yields an oracle-free circuit that agrees with the original one on all but a small fraction of inputs; the oracle-free circuit can then be inverted with the help of the PSPACE oracle, and the inversion carries over to the original circuit by the agreement. The set Q is constructed iteratively over the query positions: for each i , one adds to Q all “heavy” points for the i -th query, namely those points x such that the i -th oracle query is of the form $(x, \cdot)$ on a noticeable fraction of the circuit’s inputs. The central difficulty is the same as ours: once the oracle is fixed, the approximation must hold *simultaneously* for all polynomial-size circuits, and a per-circuit failure probability of 2^{-n} , which is all that a naive replacement of the oracle by the all-zero oracle achieves, cannot survive a union bound over the $2^{\text{poly}(n)}$ circuits. Wee resolved this by showing, via the Hoeffding bound, that the light queries cause noticeable disagreement with probability only $e^{-\Omega(sn^2)}$ over f , which beats the number $2^{O(sn \log(sn))}$ of circuits of size s . This is precisely the role played in our proof by the exponential moment bound on the number of discoveries, which yields a failure probability small enough to beat the union bound over the 2^n auxiliary inputs of each length.

Beyond this analogy, however, his technique does not transfer to our setting, for three reasons. First, auxiliary-input OWFs are secure against nonuniform adversaries, so the set Q and the values of f on Q merely need to *exist*: they are hardwired into the inverter as nonuniform advice. In contrast, the security of auxiliary-input OWPuzzs and EFI pairs is defined with respect to uniform adversaries, together with an adversary-dependent infinite set of hard auxiliary inputs, so we must exhibit a single uniform QPT extractor that *efficiently learns* the patch by interacting with the oracle. Second, the heaviness of a query is defined in terms of the fraction of the circuit’s classical inputs on which the query takes a given value, a quantity that is no longer meaningful for algorithms that query the oracle in superposition; our extractor instead runs the algorithm up to a randomly chosen query, measures the query register, and tests the outcome against V^π , in the spirit of the heavy-query learners of Impagliazzo and Rudich [IR89] and Barak and Mahmoody [BM09], combined with a BBBV-style hybrid argument [BBBV97]. Third, the probabilistic structure is different: the values of a uniformly random function are independent, which is what permits a direct application of the Hoeffding bound, whereas our oracle encodes uniformly random permutations, so we instead argue that, conditioned on the entire transcript of the extraction, every fresh triple tested by the extractor is a true positive with probability at most the reciprocal of the number of extraction stages, and lift this martingale-type bound to the exponential moment estimate. For these reasons, we view the patching lemma as a uniform, quantum analogue of Wee’s approximation lemma rather than an application of it.

2 Preliminaries

2.1 Basic Notations

We use the standard notation of quantum information and cryptography. The security parameter is λ . $[n]$ is the set $\{1, 2, \dots, n\}$. For a set S , $x \leftarrow S$ means that an element x is sampled uniformly at random from S . For a bit string x , $|x|$ is the length of x . For an algorithm $\mathcal{A}$, $y \leftarrow \mathcal{A}(x)$ means that the algorithm $\mathcal{A}$ is run on input x and outputs y . For simplicity, we often denote $\mathcal{A}(x)$ by the output of the algorithm $\mathcal{A}$ on input x . (If $\mathcal{A}$ is a probabilistic algorithm that outputs a classical bit string, $\mathcal{A}(x)$ is a classical distribution. If $\mathcal{A}$ outputs a quantum state, $\mathcal{A}(x)$ is a quantum state.) TD is the trace distance. SD is the statistical distance. negl is a negligible function.

2.2 Average-Case Decision and Search Hardness

Throughout this subsection, we fix an oracle $\mathcal{O}$. For a polynomial S , an $S(N)$ -qubit oracle-dependent quantum advice family is a family of density operators $\{\rho_{N,\mathcal{O}}\}_{N \in \mathbb{N}}$, where $\rho_{N,\mathcal{O}}$ is an $S(N)$ -qubit state depending on the security parameter N and on the oracle $\mathcal{O}$.⁴

Definition 2.1 (Strong average-case hardness of languages). Let $L^\mathcal{O} \subseteq \{0, 1\}^*$ be a language relative to a fixed oracle $\mathcal{O}$. We say that $L^\mathcal{O}$ is strongly hard on average against uniform QPT algorithms that can query $\mathcal{O}$ if, for every uniform QPT oracle algorithm $\mathcal{A}$, there exists a negligible function negl such that

$$\Pr_{z \leftarrow \{0,1\}^N} [\mathcal{A}^\mathcal{O}(1^N, z) = L^\mathcal{O}(z)] \leq \frac{1}{2} + \text{negl}(N). \quad (17)$$

We say that $L^\mathcal{O}$ is strongly hard on average against QPT algorithms that can query $\mathcal{O}$ and can receive polynomial-size oracle-dependent quantum advice if, for every polynomial S , every uniform QPT oracle algorithm $\mathcal{A}$, and every $S(N)$ -qubit oracle-dependent quantum advice family $\{\rho_{N,\mathcal{O}}\}_{N \in \mathbb{N}}$, there exists a negligible function negl such that

$$\Pr_{z \leftarrow \{0,1\}^N} [\mathcal{A}^\mathcal{O}(1^N, z, \rho_{N,\mathcal{O}}) = L^\mathcal{O}(z)] \leq \frac{1}{2} + \text{negl}(N). \quad (18)$$

Remark 2.2. In the above definition, an instance z is sampled uniformly at random. We could define the average-case hardness in a more general way where an instance is sampled from some QPT or PPT samplers, but in this paper, we can show the hardness with the uniform distribution.

Definition 2.3 (Average-case hardness of total unique search). Let $R^\mathcal{O} \subseteq \{0, 1\}^* \times \{0, 1\}^*$ be a relation relative to a fixed oracle $\mathcal{O}$. The relation is polynomially balanced if there exists a polynomial p such that $|w| \leq p(|x|)$ for all $(x, w) \in R^\mathcal{O}$. We say that $R^\mathcal{O} \in \text{TFUP}^\mathcal{O}$ if the following three properties are satisfied:

1. $R^\mathcal{O}$ is polynomially balanced.
2. There exists a classical deterministic polynomial-time oracle algorithm $\mathcal{V}$ such that, for every x and w , $\mathcal{V}^\mathcal{O}(x, w) = R^\mathcal{O}(x, w)$.
3. For every instance x , $|\{w \in \{0, 1\}^* : R^\mathcal{O}(x, w) = 1\}| = 1$. In other words, every instance has exactly one polynomial-length valid solution.

We say that $R^\mathcal{O}$ is average-case hard against uniform QPT algorithms that can query $\mathcal{O}$ if, for every uniform QPT oracle algorithm $\mathcal{A}$, there exists a negligible function negl such that

$$\Pr [R^\mathcal{O}(x, \hat{w}) = 1 : x \leftarrow \{0, 1\}^N, \hat{w} \leftarrow \mathcal{A}^\mathcal{O}(1^N, x)] \leq \text{negl}(N). \quad (19)$$

We say that $R^\mathcal{O}$ is average-case hard against QPT algorithms that can query $\mathcal{O}$ and can receive polynomial-size oracle-dependent quantum advice if, for every polynomial S , every uniform QPT oracle algorithm $\mathcal{A}$, and every $S(N)$ -qubit oracle-dependent quantum advice family $\{\rho_{N,\mathcal{O}}\}_{N \in \mathbb{N}}$, there exists a negligible function negl such that

$$\Pr [R^\mathcal{O}(x, \hat{w}) = 1 : x \leftarrow \{0, 1\}^N, \hat{w} \leftarrow \mathcal{A}^\mathcal{O}(1^N, x, \rho_{N,\mathcal{O}})] \leq \text{negl}(N). \quad (20)$$

⁴When an oracle $\mathcal{O}$ is fixed, the advice states can of course depend on the oracle. Here, our intention is to write the oracle dependence explicitly for the later use.

2.3 Auxiliary-Input Primitives

Definition 2.4 (Auxiliary-input OWPuzzs). Fix an oracle $\mathcal{O}$. An auxiliary-input one-way puzzle (OWPuzz) relative to $\mathcal{O}$ is a pair $(\text{Samp}, \text{Ver})$ of oracle algorithms with the following syntax.

1. $\text{Samp}^{\mathcal{O}}(x) \rightarrow (\text{puzz}, \text{ans})$: It is a uniform QPT oracle algorithm that, on input a classical auxiliary input $x \in \{0, 1\}^*$, outputs a pair of classical strings $(\text{puzz}, \text{ans})$.
2. $\text{Ver}^{\mathcal{O}}(x, \text{puzz}, \text{ans}') \rightarrow \top / \perp$: It is a not-necessarily-efficient oracle algorithm that halts on every input and outputs $\top / \perp$.

We require the following properties.

Correctness. There exists a negligible function negl such that, for every $x \in \{0, 1\}^*$,

$$\Pr \left[\top \leftarrow \text{Ver}^{\mathcal{O}}(x, \text{puzz}, \text{ans}) : (\text{puzz}, \text{ans}) \leftarrow \text{Samp}^{\mathcal{O}}(x) \right] \geq 1 - \text{negl}(|x|). \quad (21)$$

Security. For every uniform QPT oracle adversary $\mathcal{A}$, there exist an infinite set $I_{\mathcal{A}} \subseteq \{0, 1\}^*$ and a negligible function negl such that, for every $x \in I_{\mathcal{A}}$,

$$\Pr \left[\top \leftarrow \text{Ver}^{\mathcal{O}}(x, \text{puzz}, \widehat{\text{ans}}) : \begin{array}{l} (\text{puzz}, \text{ans}) \leftarrow \text{Samp}^{\mathcal{O}}(x) \\ \widehat{\text{ans}} \leftarrow \mathcal{A}^{\mathcal{O}}(x, \text{puzz}) \end{array} \right] \leq \text{negl}(|x|). \quad (22)$$

An AI-OWPuzz is called classically secure if the security condition is required only for uniform PPT oracle adversaries $\mathcal{A}$.

Definition 2.5 (Auxiliary-input distributional OWPuzzs). Fix an oracle $\mathcal{O}$. An auxiliary-input distributional OWPuzz relative to $\mathcal{O}$ is a uniform QPT oracle algorithm $\text{Samp}^{\mathcal{O}}(x) \rightarrow (\text{puzz}, \text{ans})$ that, on input a classical auxiliary input $x \in \{0, 1\}^*$, outputs a pair of classical strings. We require the following security.

Security. There exists a polynomial p such that, for every uniform QPT oracle adversary $\mathcal{A}$, there exists an infinite set $I_{\mathcal{A}} \subseteq \{0, 1\}^*$ such that, for every $x \in I_{\mathcal{A}}$,

$$\text{SD} \left((\text{puzz}, \mathcal{A}^{\mathcal{O}}(x, \text{puzz}))_{\text{puzz} \leftarrow \text{Samp}^{\mathcal{O}}(x)}, (\text{puzz}, \text{ans})_{(\text{puzz}, \text{ans}) \leftarrow \text{Samp}^{\mathcal{O}}(x)} \right) \geq \frac{1}{p(|x|)}. \quad (23)$$

An auxiliary-input distributional OWPuzz is called classically secure if the security condition is required only for uniform PPT oracle adversaries $\mathcal{A}$.

The following lemma is straightforwardly obtained from the definitions.

Lemma 2.6 (OWPuzzs imply distributional OWPuzzs, auxiliary-input version). Fix an oracle $\mathcal{O}$. If $(\text{Samp}, \text{Ver})$ is an auxiliary-input (classically-secure) OWPuzz relative to $\mathcal{O}$ then Samp is an auxiliary-input (classically-secure) distributional OWPuzz relative to $\mathcal{O}$.

Definition 2.7 (Auxiliary-input EFI pairs). Fix an oracle $\mathcal{O}$. An auxiliary-input EFI pair (generator) relative to $\mathcal{O}$ is a uniform QPT oracle algorithm Gen that, on input a classical auxiliary input $x \in \{0, 1\}^*$ and a bit $b \in \{0, 1\}$, outputs a state $\xi_{b,x} \leftarrow \text{Gen}^{\mathcal{O}}(x, b)$. We require that there exists a positive polynomial p such that, for every uniform QPT oracle distinguisher $\mathcal{D}$, there exists an infinite set $S_{\mathcal{D}} \subseteq \{0, 1\}^*$ satisfying the following.

Statistical fairness. For every $x \in S_{\mathcal{D}}$,

$$\text{TD}(\xi_{0,x}, \xi_{1,x}) \geq \frac{1}{p(|x|)}. \quad (24)$$

Computational indistinguishability. There exists a negligible function negl such that for all $x \in S_{\mathcal{D}}$,

$$\left| \Pr \left[1 \leftarrow \mathcal{D}^{\mathcal{O}}(x, \xi_{0,x}) \right] - \Pr \left[1 \leftarrow \mathcal{D}^{\mathcal{O}}(x, \xi_{1,x}) \right] \right| \leq \text{negl}(|x|). \quad (25)$$

Remark 2.8. Auxiliary-input EFI pairs were defined in [BCQ23]. Their definition is stated in a different way from the above one, but is actually equivalent to the above one. (Their infinite set S is independent of the distinguisher, and they require that for every polynomial q there exists $x \in S$ such that the advantage is at most $1/q(|x|)$.)

2.4 Sampling Complexity

Definition 2.9 (Sampling Problems [Aar14, ABK24]). A (polynomially-bounded) sampling problem S is a collection of probability distributions $\{D_x\}_{x \in \{0,1\}^*}$, where D_x is a distribution over $\{0,1\}^{p(|x|)}$, for some fixed polynomial p .

Definition 2.10 (SampBPP and SampBQP [Aar14, ABK24]). SampBPP is the class of (polynomially-bounded) sampling problems $S = \{D_x\}_{x \in \{0,1\}^*}$ for which there exists a PPT algorithm $\mathcal{B}$ such that for all x and all $\epsilon > 0$,

$$\text{SD}(\mathcal{B}(x, 1^{\lfloor 1/\epsilon \rfloor}), D_x) \leq \epsilon, \quad (26)$$

where $\mathcal{B}(x, 1^{\lfloor 1/\epsilon \rfloor})$ is the output probability distribution of $\mathcal{B}$ on input $(x, 1^{\lfloor 1/\epsilon \rfloor})$. SampBQP is defined the same way, except that $\mathcal{B}$ is a QPT algorithm rather than a PPT one.

3 Oracle

For every $\ell \in \mathbb{N}$, let π_ℓ be a permutation over $\{0,1\}^\ell$ chosen uniformly at random and write $\pi := (\pi_1, \pi_2, \dots)$. For a fixed realization π , define the oracle V^π by

$$V^\pi(\ell, u, v) := \begin{cases} 1, & |u| = |v| = \ell \text{ and } \pi_\ell(u) = v, \\ 0, & \text{otherwise.} \end{cases} \quad (27)$$

We use QBF as a fixed PSPACE-complete Boolean oracle. We also use the following quantum oracle, which is fixed independently of π .

Definition 3.1 (Helstrom oracle). We define a set $\mathcal{H} = (\mathcal{H}^{(0)}, \mathcal{H}^{(1)}, \mathcal{H}^{(2)}, \dots)$ of quantum oracles as follows. The level zero oracle $\mathcal{H}^{(0)}$ acts as the identity. A query to the level- $(j+1)$ oracle $\mathcal{H}^{(j+1)}$ contains 1^{j+1} , classical descriptions of two quantum circuits C_0 and C_1 , a classical string z , a state $|\psi\rangle$, and a one-qubit answer register. The circuits C_0 and C_1 may query levels $\mathcal{H}^{(0)}, \dots, \mathcal{H}^{(j)}$, but they may not query V^π or any other oracle depending on π . Let $\sigma_{b,z}$ be the output density operator of $C_b(z)$, and let $M_{C_0, C_1, z}$ be the projector onto the positive eigenspace of $\sigma_{1,z} - \sigma_{0,z}$. On the state $|\psi\rangle$ and the answer register, the oracle applies

$$U_{C_0, C_1, z} := (I - M_{C_0, C_1, z}) \otimes I + M_{C_0, C_1, z} \otimes X. \quad (28)$$

Invalid descriptions act as the identity. This defines $\mathcal{H}$ inductively because a query to the level- $(j+1)$ oracle $\mathcal{H}^{(j+1)}$ refers only to already-defined lower levels. Note that the unary level encoding implies that a polynomial-time algorithm on n -bit inputs can access only levels $j = \text{poly}(n)$.

4 Average-Case Hard Problems

We use the following theorem.

Theorem 4.1 (Theorem 1 of [ABC⁺26]). *Let π be a uniformly random permutation over $[M]$. A quantum algorithm $\mathcal{A}$ can make at most T forward queries to π and can make queries to a side oracle w , which is independent of π . $\mathcal{A}$ receives a uniformly random $y \leftarrow [M]$ as the challenge input. $\mathcal{A}$ can also receive an S -qubit quantum advice state $\rho_{\pi, \mathcal{A}, w}$ that may depend arbitrarily on the permutation π , on the description of $\mathcal{A}$, and on the side oracle w , but it must be independent of the challenge y . Finally, $\mathcal{A}$ outputs x . Then*

$$\mathbb{E}_{\pi} \Pr [\pi(x) = y : y \leftarrow [M], x \leftarrow \mathcal{A}^{\pi, w}(y, \rho_{\pi, \mathcal{A}, w})] \leq C \left(\frac{ST}{M} + \frac{T^2}{M} \right), \quad (29)$$

where C is a universal constant.

Remark 4.2. Theorem 1 of [ABC⁺26] is stated without the side oracle w . Granting access to a side oracle w whose action is fixed independently of π is without loss of generality: the algorithm is constrained only in the number of queries to π and may apply arbitrary unitaries between the queries, so the action of w (and of its inverse) on the relevant registers can be absorbed into these unitaries, and the dependence of the advice state on w is subsumed by its dependence on the description of $\mathcal{A}$. We nevertheless state the side oracle explicitly for our later uses.

From Theorem 4.1, we obtain the following corollary.

Corollary 4.3. *Let $\pi = (\pi_1, \pi_2, \dots)$, where, independently for every $\ell \in \mathbb{N}$, π_{ℓ} is chosen uniformly from the permutations of $\{0, 1\}^{\ell}$. Fix any collection $\mathcal{B}$ of classical or quantum side oracles whose action is fixed independently of π . Then, with probability one over the choice of π , the following statement holds.*

For every uniform QPT oracle algorithm $\mathcal{A}$, every pair of polynomials S and T such that $\mathcal{A}$ makes at most $T(n)$ queries to V^{π} on inputs of length n , and every family of $S(n)$ -qubit advice states $\{\rho_{n, \pi, \mathcal{A}, \mathcal{B}}\}_{n \in \mathbb{N}}$, there exists a negligible function negl such that

$$\Pr \left[x = \pi_n^{-1}(y) : y \leftarrow \{0, 1\}^n, x \leftarrow \mathcal{A}^{V^{\pi}, \mathcal{B}}(1^n, y, \rho_{n, \pi, \mathcal{A}, \mathcal{B}}) \right] \leq \text{negl}(n). \quad (30)$$

The advice state may depend arbitrarily on the entire fixed permutation family π , on the description of $\mathcal{A}$, and on $\mathcal{B}$, but it must be independent of y .

Proof. Fix a uniform QPT oracle algorithm $\mathcal{A}$ and polynomials S and T as in the statement. We first prove a probability-one statement for this fixed triple $(\mathcal{A}, S, T)$ and then take a countable intersection over all such triples.

Fix an input length n . Let $\pi_{\neq n} := (\pi_{\ell})_{\ell \neq n}$ denote a fixed realization of all non-target permutation blocks, where each π_{ℓ} is a permutation of $\{0, 1\}^{\ell}$. For an $S(n)$ -qubit state ρ , define

$$\text{Succ}_n^{\mathcal{A}, \mathcal{B}}(\pi_n, \pi_{\neq n}; \rho) := \Pr \left[x = \pi_n^{-1}(y) : y \leftarrow \{0, 1\}^n, x \leftarrow \mathcal{A}^{V^{\pi}, \mathcal{B}}(1^n, y, \rho) \right]. \quad (31)$$

Define the best success probability obtainable from an $S(n)$ -qubit advice state by

$$\text{Opt}_n^{\mathcal{A}, S, \mathcal{B}}(\pi_n, \pi_{\neq n}) := \max_{\rho} \text{Succ}_n^{\mathcal{A}, \mathcal{B}}(\pi_n, \pi_{\neq n}; \rho). \quad (32)$$

We next bound the expectation of Equation (32) over a uniformly random π_n , while keeping $\pi_{\neq n}$ fixed. A query to V^π can be simulated by at most two forward queries to π_n . Applying Theorem 4.1, for every fixed $\pi_{\neq n}$,

$$\mathbb{E}_{\pi_n} \left[\text{Opt}_n^{\mathcal{A}, S, \mathcal{B}}(\pi_n, \pi_{\neq n}) \right] \leq C \left(\frac{2S(n)T(n)}{2^n} + \frac{4T(n)^2}{2^n} \right). \quad (33)$$

Here, we have used the fact that a query to V^π can be simulated by querying π twice. Hence the tower property of conditional expectation gives

$$\mathbb{E}_\pi \left[\text{Opt}_n^{\mathcal{A}, S, \mathcal{B}}(\pi_n, \pi_{\neq n}) \right] = \mathbb{E}_{\pi_{\neq n}} \left[\mathbb{E}_{\pi_n} \left[\text{Opt}_n^{\mathcal{A}, S, \mathcal{B}}(\pi_n, \pi_{\neq n}) \middle| \pi_{\neq n} \right] \right] \leq C \left(\frac{2S(n)T(n)}{2^n} + \frac{4T(n)^2}{2^n} \right). \quad (34)$$

Set $\eta(n) := 2^{-n/4}$ and define the bad event

$$\text{Bad}_n^{\mathcal{A}, S, T, \mathcal{B}} := \left\{ \pi : \text{Opt}_n^{\mathcal{A}, S, \mathcal{B}}(\pi_n, \pi_{\neq n}) > \eta(n) \right\}. \quad (35)$$

By Markov's inequality and Equation (34),

$$\Pr_\pi \left[\text{Bad}_n^{\mathcal{A}, S, T, \mathcal{B}} \right] \leq C \left(2S(n)T(n) + 4T(n)^2 \right) 2^{-3n/4}. \quad (36)$$

Since S and T are polynomials, the right-hand side of Equation (36) is summable in n . The first Borel–Cantelli lemma therefore implies that, for the fixed triple $(\mathcal{A}, S, T)$, with probability one over π only finitely many of the events $\text{Bad}_n^{\mathcal{A}, S, T, \mathcal{B}}$ occur. Equivalently, with probability one over π there exists an integer $n_0 = n_0(\mathcal{A}, S, T, \mathcal{B}, \pi)$ such that

$$\text{Opt}_n^{\mathcal{A}, S, \mathcal{B}}(\pi_n, \pi_{\neq n}) \leq 2^{-n/4} \quad (37)$$

for every $n \geq n_0$.

There are only countably many uniform oracle machines and only countably many polynomial bounds with finite descriptions. Taking the countable intersection of these events of probability one produces a set of complete permutation families that also has probability one and for which Equation (37) holds simultaneously for every triple $(\mathcal{A}, S, T)$. $\square$

Lemma 4.4 (Goldreich-Levin with quantum advice [KT24, AC02]). *There exists a QPT algorithm $\mathcal{E}$ with the following property. For every $n \in \mathbb{N}$, every QPT algorithm $\mathcal{D}$ that outputs a bit, every mixed state ρ , every string $a \in \{0, 1\}^n$, and every $\varepsilon > 0$, if*

$$\Pr_{r \leftarrow \{0, 1\}^n} [\mathcal{D}(\rho, r) = \langle a, r \rangle] \geq \frac{1}{2} + \varepsilon, \quad (38)$$

then

$$\Pr \left[\mathcal{E}^{\mathcal{D}}(\rho) = a \right] \geq 4\varepsilon^2. \quad (39)$$

Here, $\mathcal{E}^{\mathcal{D}}$ denotes the coherent black-box use of a unitary implementation of $\mathcal{D}$ and its inverse.

Corollary 4.5. Fix $n \in \mathbb{N}$ and an arbitrary family of strings $\{x_y \in \{0, 1\}^n\}_{y \in \{0, 1\}^n}$. Let $\mathcal{O}$ be any fixed collection of reversible oracle unitaries whose inverse queries are available, and let $\mathcal{P}^{\mathcal{O}}$ be a uniform QPT oracle algorithm that outputs a bit. Let ρ be an arbitrary mixed advice state that may depend on n , on $\mathcal{O}$, on the description of $\mathcal{P}$, and on the family $\{x_y\}_y$, but it may not depend on y and r . Suppose that, for some $\varepsilon > 0$,

$$\Pr_{\substack{y \leftarrow \{0, 1\}^n \\ r \leftarrow \{0, 1\}^n}} \left[\mathcal{P}^{\mathcal{O}}(\rho, y, r) = \langle x_y, r \rangle \right] \geq \frac{1}{2} + \varepsilon. \quad (40)$$

Then there exists a uniform QPT oracle algorithm $\mathcal{R}^{\mathcal{O}}$, whose code depends only on the code of $\mathcal{P}$, such that

$$\Pr_{y \leftarrow \{0, 1\}^n} \left[x_y \leftarrow \mathcal{R}^{\mathcal{O}}(\rho, y) \right] \geq 4\varepsilon^2. \quad (41)$$

Proof. For every $y \in \{0, 1\}^n$, define

$$\delta_y := \Pr_{r \leftarrow \{0, 1\}^n} \left[\mathcal{P}^{\mathcal{O}}(\rho, y, r) = \langle x_y, r \rangle \right] - \frac{1}{2}, \quad (42)$$

$$\delta_y^+ := \max\{\delta_y, 0\}. \quad (43)$$

Equation (40) is equivalent to

$$\mathbb{E}_{y \leftarrow \{0, 1\}^n} [\delta_y] \geq \varepsilon. \quad (44)$$

Fix $y \in \{0, 1\}^n$ and define the predictor $\mathcal{D}_y^{\mathcal{O}}$ by hardwiring y into $\mathcal{P}^{\mathcal{O}}$:

$$\mathcal{D}_y^{\mathcal{O}}(\rho, r) := \mathcal{P}^{\mathcal{O}}(\rho, y, r). \quad (45)$$

Define the recovery algorithm by

$$\mathcal{R}^{\mathcal{O}}(\rho, y) := \mathcal{E}^{\mathcal{D}_y^{\mathcal{O}}}(\rho), \quad (46)$$

where $\mathcal{E}$ is defined in Equation (39). If $\delta_y > 0$, then Lemma 4.4 applied with $a = x_y$ and $\varepsilon = \delta_y$, gives

$$\Pr \left[\mathcal{R}^{\mathcal{O}}(\rho, y) = x_y \right] \geq 4\delta_y^2. \quad (47)$$

If $\delta_y \leq 0$, the recovery probability is nonnegative. Consequently, for every $y \in \{0, 1\}^n$,

$$\Pr \left[\mathcal{R}^{\mathcal{O}}(\rho, y) = x_y \right] \geq 4 \left(\delta_y^+ \right)^2. \quad (48)$$

Averaging Equation (48) over a uniform y and applying Jensen's inequality to the convex function $t \mapsto t^2$ yields

$$\Pr_{\substack{y \leftarrow \{0, 1\}^n \\ \hat{x} \leftarrow \mathcal{R}^{\mathcal{O}}(\rho, y)}} \left[\hat{x} = x_y \right] \geq 4 \mathbb{E}_y \left[\left(\delta_y^+ \right)^2 \right] \geq 4 \left(\mathbb{E}_y \left[\delta_y^+ \right] \right)^2 \geq 4 \left(\mathbb{E}_y \left[\delta_y \right] \right)^2 \geq 4\varepsilon^2. \quad (49)$$

Here the final inequality follows from Equation (44). $\square$

Lemma 4.6 (Decision and search hardness). Let $\pi := (\pi_1, \pi_2, \dots)$, where π_ℓ is a permutation over $\{0, 1\}^\ell$. For a string $z \in \{0, 1\}^N$ with $N \geq 2$, set $n := \lfloor N/2 \rfloor$ and parse $z = y\|r\|s$, $|y| = |r| = n$, and $|s| = N - 2n$. Define the language

$$L^\pi(z) := \langle \pi_n^{-1}(y), r \rangle \bmod 2. \quad (50)$$

In other words, $z \in L^\pi$ if and only if $\langle \pi_n^{-1}(y), r \rangle = 1$. The three strings of length less than two, namely, the empty string, 0 and 1, may be assigned arbitrary values. Define also the relation

$$R^\pi(y, w) = 1 \iff |y| = |w| \text{ and } V^\pi(|y|, w, y) = 1. \quad (51)$$

Let $\mathcal{B}$ be any fixed collection of reversible oracle unitaries independent of π , and suppose that inverse queries to every oracle in $\mathcal{B}$ are available. Let π be a permutation family satisfying Corollary 4.3. Then, relative to V^π ,

$$R^\pi \in \text{TFUP}^{V^\pi}, \quad (52)$$

$$L^\pi \in \text{UP}^{V^\pi} \cap \text{coUP}^{V^\pi}. \quad (53)$$

Moreover, relative to $(V^\pi, \mathcal{B})$, R^π is average-case hard against QPT algorithms with polynomial-size oracle-dependent quantum advice, and L^π is strongly hard on average against QPT algorithms with polynomial-size oracle-dependent quantum advice.

Proof. For every $y \in \{0, 1\}^n$, the relation R^π has exactly one valid solution, namely $w = \pi_n^{-1}(y)$. Every valid solution satisfies $|w| = |y|$, so the relation is polynomially balanced. A deterministic polynomial-time oracle verifier checks whether $V^\pi(n, w, y) = 1$. This proves Equation (52). Furthermore, the average-case hardness of R^π follows directly from Corollary 4.3.

We next prove Equation (53). The unique witness for both L^π and its complement is $w = \pi_n^{-1}(y)$. The verifier first checks $V^\pi(n, w, y) = 1$, and then checks $\langle w, r \rangle = 1 \pmod{2}$ for membership in L^π , or $\langle w, r \rangle = 0 \pmod{2}$ for membership in its complement. Since π_n is a permutation, there is exactly one possible witness. This proves Equation (53).

It remains to prove the strong average-case hardness of L^π . Assume toward a contradiction that it fails. Then there exist a polynomial S , a uniform QPT oracle algorithm $\mathcal{A}$, an $S(N)$ -qubit oracle-dependent quantum advice family $\{\rho_N\}_{N \in \mathbb{N}}$ ⁵, and a positive polynomial q such that, for infinitely many input lengths N ,

$$\Pr_{z \leftarrow \{0,1\}^N} [\mathcal{A}^{V^\pi, \mathcal{B}}(1^N, z, \rho_N) = L^\pi(z)] \geq \frac{1}{2} + \frac{1}{q(N)}. \quad (54)$$

For $n \in \mathbb{N}$ and $\beta \in \{0, 1\}$, define $N_\beta(n) := 2n + \beta$. We define two uniform QPT predictors. On input (ρ, y, r) with $|y| = |r| = n$, let

$$\mathcal{P}_0^{V^\pi, \mathcal{B}}(\rho, y, r) := \mathcal{A}^{V^\pi, \mathcal{B}}(1^{2n}, y\|r, \rho). \quad (55)$$

The predictor $\mathcal{P}_1$ samples $s \leftarrow \{0, 1\}$ and outputs

$$\mathcal{P}_1^{V^\pi, \mathcal{B}}(\rho, y, r) := \mathcal{A}^{V^\pi, \mathcal{B}}(1^{2n+1}, y\|r\|s, \rho). \quad (56)$$

⁵For notational simplicity, we omit the dependence of ρ on V^π , $\mathcal{B}$ and $\mathcal{A}$.

The random bit in Equation (56) can be generated coherently and its measurement can be deferred. Since V^π is self-inverse and inverse queries to $\mathcal{B}$ are available, both predictors admit the coherent forward and inverse implementations required by Corollary 4.5.

For each $\beta \in \{0, 1\}$, define

$$\alpha_{n,\beta} := \Pr_{\substack{y \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n}} \left[\mathcal{P}_\beta^{V^\pi, \mathcal{B}}(\rho_{N_\beta(n)}, y, r) = \langle \pi_n^{-1}(y), r \rangle \right] - \frac{1}{2}. \quad (57)$$

Because L^π ignores the padding string s ,

$$\alpha_{n,\beta} = \Pr_{z \leftarrow \{0,1\}^{N_\beta(n)}} \left[\mathcal{A}^{V^\pi, \mathcal{B}}(1^{N_\beta(n)}, z, \rho_{N_\beta(n)}) = L^\pi(z) \right] - \frac{1}{2}. \quad (58)$$

For each $\beta \in \{0, 1\}$, apply Corollary 4.5 to $\mathcal{P}_\beta$. This gives a uniform QPT recovery algorithm $\mathcal{R}_\beta^{V^\pi, \mathcal{B}}$ such that, whenever $\alpha_{n,\beta} > 0$,

$$\Pr_{\substack{y \leftarrow \{0,1\}^n \\ \hat{w}_\beta \leftarrow \mathcal{R}_\beta^{V^\pi, \mathcal{B}}(\rho_{N_\beta(n)}, y)}} \left[\hat{w}_\beta = \pi_n^{-1}(y) \right] \geq 4\alpha_{n,\beta}^2. \quad (59)$$

Each invocation in Equation (59) uses one copy of the corresponding advice state.

Let $\mathcal{N} \subseteq \mathbb{N}$ be the infinite set of input lengths for which Equation (54) holds. For $\beta \in \{0, 1\}$, define $\mathcal{N}_\beta := \{n \in \mathbb{N} : 2n + \beta \in \mathcal{N}\}$. Since $\mathcal{N}$ is infinite, at least one of $\mathcal{N}_0$ and $\mathcal{N}_1$ is infinite. Fix $\beta^* \in \{0, 1\}$ such that $\mathcal{N}_{\beta^*}$ is infinite. For every $n \in \mathcal{N}_{\beta^*}$, Equation (54) and Equation (58) give $\alpha_{n,\beta^*} \geq \frac{1}{q(2n+\beta^*)}$. In particular, $\alpha_{n,\beta^*} > 0$, and hence Equation (59) implies

$$\Pr_{\substack{y \leftarrow \{0,1\}^n \\ \hat{w} \leftarrow \mathcal{R}_{\beta^*}^{V^\pi, \mathcal{B}}(\rho_{2n+\beta^*}, y)}} \left[\hat{w} = \pi_n^{-1}(y) \right] \geq \frac{4}{q(2n+\beta^*)^2} \quad (60)$$

for infinitely many n . This contradicts Corollary 4.3. $\square$

5 Patching Lemma

Definition 5.1 (Patches). A patch is a finite set Γ of triples (ℓ, u, v) with $|u| = |v| = \ell$. It defines the oracle Γ as

$$\Gamma(\ell, u, v) := \begin{cases} 1, & (\ell, u, v) \in \Gamma, \\ 0, & \text{otherwise.} \end{cases} \quad (61)$$

Lemma 5.2 (Patching lemma). Fix a side oracle $\mathcal{B}$ independent of π . With probability one over π , the following holds simultaneously for every uniform QPT oracle algorithm $\mathcal{A}$ and every positive polynomial r .

Suppose that, on input (x, b) with $x \in \{0, 1\}^n$ and $b \in \{0, 1\}$, $\mathcal{A}^{V^\pi, \mathcal{B}}$ outputs a mixed state. Then there is a uniform QPT oracle algorithm

$$\Gamma \leftarrow \mathcal{E}_{\mathcal{A}, r}^{V^\pi, \mathcal{B}}(x) \quad (62)$$

that receives x but not the bit b , and outputs a polynomial-length patch Γ . (Here, $\mathcal{E}$ queries V^π only classically.) If

$$\rho_{b,x} := \mathcal{A}^{V^\pi, \mathcal{B}}(x, b), \quad (63)$$

$$\sigma_{b,x}^\Gamma := \mathcal{A}^{\Gamma, \mathcal{B}}(x, b), \quad (64)$$

then, for all sufficiently large n , simultaneously for every $x \in \{0, 1\}^n$,

$$\frac{1}{2} \sum_{b \in \{0,1\}} \mathbb{E}_\Gamma \left[\text{TD} \left(\rho_{b,x}, \sigma_{b,x}^\Gamma \right) \right] \leq \frac{1}{r(n)}. \quad (65)$$

The same sampled patch is used for both values of b .

Proof. Fix $\mathcal{A}$, r , and an input length n . Let T be the number of queries $\mathcal{A}$ makes to V^π . Set

$$R := \left\lceil 16 T^2 (n+2) r^2 n^4 \right\rceil, \quad (66)$$

$$m := \lceil \log_2(2R) \rceil, \quad \text{so that } 2^m \geq 2R. \quad (67)$$

Note that $R = \text{poly}(n)$ and $2^m < 4R$, so all quantities below are polynomial in n .

The extractor. The extractor $\mathcal{E}$ first learns, by classical queries to V^π , the complete positive table of the *short* blocks,

$$\text{Short} := \{(\ell, u, v) : \ell < m, V^\pi(\ell, u, v) = 1\}; \quad (68)$$

this takes fewer than $2^{2m} < 16R^2$ queries. For a set D of triples in blocks $\ell \geq m$, write $\Gamma(D) := \text{Short} \cup D$.

Define the auxiliary sampler $\bar{\mathcal{A}}$ as follows:

1. On input a patch Γ and x , sample $b \leftarrow \{0, 1\}$ and $i \leftarrow [T]$ uniformly.
2. Run $\mathcal{A}^{\Gamma, \mathcal{B}}(x, b)$ until immediately before its i -th query.
3. Measure the query input register, and output the outcome.

The extractor $\mathcal{E}$ then proceeds as follows:

1. Initialize $D_0 := \emptyset$.
2. For $j = 0, \dots, R-1$: run $Q_j \leftarrow \bar{\mathcal{A}}^{\Gamma(D_j), \mathcal{B}}(x)$; if $Q_j = (\ell, u, v)$ with $\ell \geq m$ and $Q_j \notin D_j$, query V^π classically on Q_j and set $D_{j+1} := D_j \cup \{Q_j\}$ if the answer is 1 and $D_{j+1} := D_j$ otherwise; in all other cases set $D_{j+1} := D_j$.
3. Sample $J \leftarrow \{0, \dots, R-1\}$ and output

$$\Gamma := \Gamma(D_J). \quad (69)$$

The bits b sampled inside $\bar{\mathcal{A}}$ are internal randomness; $\mathcal{E}$ never receives the later challenge bit. The procedure is uniform QPT, every element of $\Gamma(D_j)$ is a true positive of V^π by construction, and the patch Γ has polynomial length.

Since $\Gamma(D)$ records all positives of the short blocks and only true positives of the long blocks, the oracles V^π and $\Gamma(D)$ differ exactly on the set of *undiscovered long positives*

$$\Delta(D) := \{(\ell, u, v) : \ell \geq m, V^\pi(\ell, u, v) = 1, (\ell, u, v) \notin D\}. \quad (70)$$

Define

$$p_x(D) := \Pr \left[\bar{\mathcal{A}}^{\Gamma(D), \mathcal{B}}(x) \in \Delta(D) \right], \quad (71)$$

$$d_x(D) := \frac{1}{2} \sum_{b \in \{0,1\}} \text{TD} \left(\rho_{b,x}, \sigma_{b,x}^{\Gamma(D)} \right). \quad (72)$$

The proof consists of four steps.

Step 1. We claim that, for every set $D \subseteq \{(\ell, u, v) : \ell \geq m, V^\pi(\ell, u, v) = 1\}$,

$$d_x(D) \leq 2T \sqrt{p_x(D)}. \quad (73)$$

This is shown based on the argument of [BBBV97]. Fix b , and for $i \in \{1, \dots, T\}$ let $q_{i,b}$ be the probability that measuring the query input register of $\bar{\mathcal{A}}^{\Gamma(D), \mathcal{B}}(x, b)$ immediately before its i -th query yields a point of $\Delta(D)$. For $i \in \{0, \dots, T\}$, consider the hybrid in which the first i queries use $\Gamma(D)$ and the rest use V^π . Adjacent hybrids share the same state before the changed query; the two query unitaries agree outside $\Delta(D)$, so the Euclidean distance created at the i -th query is at most $2\sqrt{q_{i,b}}$, and later operations and the partial trace do not increase it. The triangle inequality and Cauchy–Schwarz give

$$\text{TD} \left(\rho_{b,x}, \sigma_{b,x}^{\Gamma(D)} \right) \leq 2 \sum_{i=1}^T \sqrt{q_{i,b}} \leq 2T \sqrt{\frac{1}{T} \sum_{i=1}^T q_{i,b}}. \quad (74)$$

Averaging over b and using the concavity of the square root yields Equation (73), because the pair (b, i) sampled by $\bar{\mathcal{A}}$ is uniform and therefore $p_x(D) = \frac{1}{2} \sum_b \frac{1}{T} \sum_i q_{i,b}$.

Step 2. In this step, we show

$$\mathbb{E}_{\text{Ext}, J} [d_x(D_J) \mid \pi] \leq 2T \sqrt{\frac{\kappa_\pi(x)}{R}}. \quad (75)$$

Here,

$$\kappa_\pi(x) = \mathbb{E}_{\text{Ext}} [K \mid \pi], \quad (76)$$

$K := \sum_{j=0}^{R-1} X_j$, $X_j := |D_{j+1}| - |D_j| \in \{0, 1\}$, and Ext denotes the internal randomness and measurement outcomes of the extraction algorithm $\mathcal{E}$. By construction, $X_j = 1$ if and only if $Q_j \in \Delta(D_j)$, and conditioned on π and D_j , the outcome Q_j is distributed as $\bar{\mathcal{A}}^{\Gamma(D_j), \mathcal{B}}(x)$. Hence

$$\mathbb{E}_{\text{Ext}} [X_j \mid \pi] = \mathbb{E}_{\text{Ext}} [p_x(D_j) \mid \pi]. \quad (77)$$

Since J is uniform over $\{0, \dots, R-1\}$ and independent of the trajectory $(D_0, \dots, D_R)$, we obtain

$$\mathbb{E}_{\text{Ext}, J} [d_x(D_J) | \pi] \leq 2T \mathbb{E}_{\text{Ext}, J} \left[\sqrt{p_x(D_J)} \mid \pi \right] \quad (78)$$

$$\leq 2T \sqrt{\mathbb{E}_{\text{Ext}, J} [p_x(D_J) | \pi]} \quad (79)$$

$$= 2T \sqrt{\frac{1}{R} \sum_{j=0}^{R-1} \mathbb{E}_{\text{Ext}} [X_j | \pi]} \quad (80)$$

$$= 2T \sqrt{\frac{\kappa_\pi(x)}{R}}, \quad (81)$$

where the first inequality is Equation (73) and the second is Jensen's inequality for the concave square root.

Step 3. In this step, we show

$$\mathbb{E}_\pi \left[\max_{x \in \{0,1\}^n} \kappa_\pi(x) \right] \leq n + 2. \quad (82)$$

The heart of the proof is the following exponential moment bound, which holds uniformly for every $x \in \{0,1\}^n$:

$$\mathbb{E}_{\pi, \text{Ext}} [2^K] \leq e. \quad (83)$$

Indeed, using $\max_x a_x \leq \log_2 \sum_x 2^{a_x}$ and Jensen's inequality twice (for the concave $\log_2$ over π , and for the convex $t \mapsto 2^t$ over Ext), Equation (83) implies

$$\mathbb{E}_\pi \left[\max_{x \in \{0,1\}^n} \kappa_\pi(x) \right] \leq \log_2 \sum_{x \in \{0,1\}^n} \mathbb{E}_\pi [2^{\kappa_\pi(x)}] \leq \log_2 \sum_{x \in \{0,1\}^n} \mathbb{E}_{\pi, \text{Ext}} [2^K] \leq \log_2(2^n e) \leq n + 2. \quad (84)$$

It remains to prove Equation (83). Let H_j denote the complete classical transcript of the extraction $\mathcal{E}$ up to and including the measured outcome Q_j at stage j , but not the stage- j test result (i.e., checking $Q_j = (\ell, u, v)$ satisfies $V^\pi(\ell, u, v) = 1$). We claim that, for every j and every realization of H_j ,

$$\Pr_{\pi, \text{Ext}} [X_j = 1 \mid H_j] \leq \frac{1}{R}. \quad (85)$$

Since $X_0, \dots, X_{j-1}$ are determined by H_j and $\mathbb{E}[2^{X_j} \mid H_j] = 1 + \Pr[X_j = 1 \mid H_j]$, taking conditional expectations given $H_{R-1}, H_{R-2}, \dots$ in turn yields

$$\mathbb{E}_{\pi, \text{Ext}} [2^K] \leq \left(1 + \frac{1}{R}\right)^R \leq e, \quad (86)$$

which is Equation (83).

To prove Equation (85), fix a realization of H_j and write $Q_j = (\ell, u, v)$. We first claim the following: if $\ell \geq m$ and the oracle value $V^\pi(\ell, u, v)$ is not determined by the transcript, then $X_j = 1$ if and only if $\pi_\ell(u) = v$; in all other cases, $X_j = 0$ deterministically.

To see the claim, we consider each case in turn.

1. If $\ell \geq m$ and the value is undetermined, then in particular $(\ell, u, v) \notin D_j$, so the extractor tests (ℓ, u, v) and adds it to the patch exactly when $V^\pi(\ell, u, v) = 1$, i.e., when $\pi_\ell(u) = v$.
2. If $\ell < m$, no test is performed and $X_j = 0$.
3. When $\ell \geq m$ but the value is determined by the transcript. This happens in exactly two ways. First, (ℓ, u, v) itself was tested at an earlier stage: if the test was positive, then $(\ell, u, v) \in D_j$ and no new test is performed; if it was negative, then the new test again returns 0. Second, an earlier positive test revealed $\pi_\ell(u) = v'$ for some $v' \neq v$ or $\pi_\ell(u') = v$ for some $u' \neq u$; since π_ℓ is a permutation, this forces $V^\pi(\ell, u, v) = 0$, so the test returns 0. In every case $X_j = 0$, which proves the claim.

By the claim, it suffices to assume that $\ell \geq m$ and that the value is undetermined, and to bound the conditional probability of $\pi_\ell(u) = v$.

We claim that, conditioned on H_j , the permutation π_ℓ is uniformly distributed over the permutations of $\{0, 1\}^\ell$ consistent with the outcomes of the earlier classical tests in block ℓ . Indeed, the extraction interacts with V^π only through the short-block queries, which concern blocks $\ell' < m \leq \ell$, and through the classical tests: the runs of $\bar{\mathcal{A}}$ query only the patched oracles, which are functions of the transcript. Since the blocks of π are independent, no other part of the transcript carries information about π_ℓ , and the claim follows.

Among the earlier classical tests in block ℓ , let k_+ be the number of positive tests and k_- that of negative tests, so that $k_+ + k_- \leq j < R$. The positive tests pin down the value of π_ℓ on k_+ inputs; we call the remaining $2^\ell - k_+$ inputs and outputs *free*. Each negative test yields a *refuted pair* (a, b) , meaning that $\pi_\ell(a) \neq b$; there are k_- of them. We say that a permutation σ *hits* a pair (a, b) if $\sigma(a) = b$. Let Σ be the set of permutations of $\{0, 1\}^\ell$ that agree with the k_+ pinned values. In this terminology, the claim says that the conditional distribution of π_ℓ given H_j is that of a uniform $\sigma \leftarrow \Sigma$ conditioned on hitting no refuted pair. Hence

$$\Pr_{\pi, \text{Ext}} [X_j = 1 \mid H_j] = \Pr_{\sigma \leftarrow \Sigma} [\sigma(u) = v \mid \sigma \text{ hits no refuted pair}] \leq \frac{\Pr_{\sigma \leftarrow \Sigma} [\sigma(u) = v]}{\Pr_{\sigma \leftarrow \Sigma} [\sigma \text{ hits no refuted pair}]} \quad (87)$$

It therefore suffices to bound the numerator from above and the denominator from below.

For a uniform $\sigma \leftarrow \Sigma$, the values of σ on the free inputs form a uniformly random bijection onto the free outputs. Since the value of (ℓ, u, v) is undetermined, both u and v are free (if u were pinned, the transcript would reveal $\pi_\ell(u)$ and thereby determine $V^\pi(\ell, u, v)$, and similarly for v), and hence

$$\Pr_{\sigma \leftarrow \Sigma} [\sigma(u) = v] = \frac{1}{2^\ell - k_+}. \quad (88)$$

Next, each refuted pair (a, b) is hit with probability at most $1/(2^\ell - k_+)$: if both a and b are free this probability is exactly $1/(2^\ell - k_+)$; if a is pinned then $\sigma(a)$ equals its pinned value $\pi_\ell(a) \neq b$, and if b is pinned then b is taken by its pinned input under every $\sigma \in \Sigma$, so in either case the pair is never hit. By the union bound,

$$\Pr_{\sigma \leftarrow \Sigma} [\sigma \text{ hits no refuted pair}] \geq 1 - \frac{k_-}{2^\ell - k_+} > 0, \quad (89)$$

where the positivity follows from $2^\ell \geq 2^m \geq 2R > k_+ + k_-$. Combining Equations (87) to (89) yields

$$\Pr_{\pi, \text{Ext}} [X_j = 1 \mid H_j] \leq \frac{1}{2^\ell - k_+ - k_-} \leq \frac{1}{2^m - R} \leq \frac{1}{R}, \quad (90)$$

where the last two inequalities use $\ell \geq m$, $k_+ + k_- < R$, and $2^m \geq 2R$.

Step 4. Define the worst simulation error for a fixed permutation family π by

$$e_n(\pi) := \max_{x \in \{0,1\}^n} \mathbb{E}_{\text{Ext}, J} [d_x(D_J) \mid \pi]. \quad (91)$$

By Equation (75), taking the expectation over π , applying Jensen's inequality to the concave square root, and using Equation (82)

$$\mathbb{E}_\pi [e_n(\pi)] \leq 2T \sqrt{\frac{\mathbb{E}_\pi [\max_{x \in \{0,1\}^n} \kappa_\pi(x)]}{R}} \leq 2T \sqrt{\frac{n+2}{R}} \leq \frac{1}{2r(n)n^2}, \quad (92)$$

where the final inequality follows from Equation (66). By Markov's inequality,

$$\Pr_\pi \left[e_n(\pi) > \frac{1}{r(n)} \right] \leq r(n) \mathbb{E}_\pi [e_n(\pi)] \leq \frac{1}{2n^2}, \quad (93)$$

which is summable in n . Hence, by the first Borel–Cantelli lemma, for the fixed pair $(\mathcal{A}, r)$, with probability one over π there exists $n_0 = n_0(\pi, \mathcal{A}, r)$ such that $e_n(\pi) \leq 1/r(n)$ for every $n \geq n_0$; that is, Equation (65) holds for every $n \geq n_0$ and every $x \in \{0,1\}^n$. There are only countably many pairs of a uniform oracle machine and a positive polynomial with finite descriptions; taking the countable intersection of the corresponding probability-one events completes the proof. $\square$

6 Proof of Theorem 1.1

Proof. Apply Corollary 4.3 with the side oracle QBF, and apply Lemma 5.2 with the side oracle QBF. Each application gives a probability-one set of permutation families. Fix one permutation family π in the intersection of these probability-one sets, and define

$$\mathcal{O} := (V^\pi, \text{QBF}). \quad (94)$$

Items (1) and (2) follow from Lemma 4.6.

Proof of Item (3). Let $\{D_x\}_{x \in \{0,1\}^*}$ be a sampling problem such that $\{D_x\}_{x \in \{0,1\}^*} \in \text{SampBQP}^\mathcal{O}$. Then, by the definition of SampBQP , there exists a uniform QPT oracle algorithm $\mathcal{Q}$ such that for all x and all $\epsilon > 0$,

$$\text{SD} \left(\mathcal{Q}^\mathcal{O} \left(x, 1^{\lceil 1/\epsilon \rceil} \right), D_x \right) \leq \epsilon. \quad (95)$$

We construct a uniform PPT oracle algorithm $\mathcal{C}$ such that $\mathcal{C}^\mathcal{O}$ satisfies the analogous guarantee.

To apply Lemma 5.2, define a uniform QPT oracle algorithm $\mathcal{A}_\mathcal{Q}$ as follows. Its classical input z is an encoding of a pair $(x, 1^K)$, where the encoding is chosen so that $N := |z| \geq K$, and it also receives a bit $b \in \{0, 1\}$. It ignores b and runs the sampler $\mathcal{Q}$:

$$\mathcal{A}_\mathcal{Q}^{V^\pi, \text{QBF}} \left(\langle x, 1^K \rangle, b \right) := \mathcal{Q}^\mathcal{O} \left(x, 1^K \right). \quad (96)$$

For $z = \langle x, 1^K \rangle$ and a patch γ , define

$$P_z := \text{the output probability distribution of } \mathcal{Q}^{V^\pi, \text{QBF}} \left(x, 1^K \right), \quad (97)$$

$$P_{z, \gamma} := \text{the output probability distribution of } \mathcal{Q}^{\gamma, \text{QBF}} \left(x, 1^K \right). \quad (98)$$

Apply Lemma 5.2 to $\mathcal{A}_Q$ with the accuracy polynomial $r(N) := N$, and let ν_z denote the output probability distribution of the patch sampled from $\mathcal{E}_{\mathcal{A}_Q, r}^{V^\pi, \text{QBF}}(z)$. Since the outputs are classical strings, trace distance coincides with statistical distance, and since $\mathcal{A}_Q$ ignores its bit b , the b -averaged bound of the lemma reads: for all sufficiently large N and simultaneously for every $z \in \{0, 1\}^N$,

$$\mathbb{E}_{\gamma \leftarrow \nu_z} [\text{SD}(P_z, P_{z, \gamma})] \leq \frac{1}{N}. \quad (99)$$

We next classically simulate both random objects appearing in this construction. First, the procedure producing $\gamma \leftarrow \nu_z$ can be simulated by a uniform $\text{PPT}^{V^\pi, \text{QBF}}$ algorithm. This is because the procedure consists of classically sampling random bits, classically querying V^π , and running a QPT^{QBF} algorithm, but a QPT^{QBF} algorithm can be simulated with a PPT^{QBF} algorithm within an exponentially small precision. Hence there is a distribution $\hat{\nu}_z$ that can be sampled by $\text{PPT}^{V^\pi, \text{QBF}}$ such that

$$\text{SD}(\nu_z, \hat{\nu}_z) \leq \text{negl}(N). \quad (100)$$

Second, if we know a patch γ , the circuit $Q^{\gamma, \text{QBF}}(x, 1^K)$ can be simulated with a PPT^{QBF} algorithm. Hence there is a distribution $\hat{P}_{z, \gamma}$ that can be sampled by PPT^{QBF} (with knowing γ) such that

$$\text{SD}(P_{z, \gamma}, \hat{P}_{z, \gamma}) \leq \text{negl}(N). \quad (101)$$

Let N_0 be a threshold beyond which Equation (99) holds. Since the permutation family π and the algorithm Q are fixed in this proof, N_0 is a fixed constant and may be hardwired into $\mathcal{C}$.

The classical sampler $\mathcal{C}^\mathcal{O}$ operates as follows. On input $(x, 1^k)$, it sets

$$K := 12(k + 1) + N_0, \quad (102)$$

$$z := \langle x, 1^K \rangle, \quad N := |z|. \quad (103)$$

It samples a patch $\gamma \leftarrow \hat{\nu}_z$ using the classical patch simulator and then samples its output according to $\hat{P}_{z, \gamma}$. This defines a single uniform $\text{PPT}^\mathcal{O}$ algorithm because $K = O(k)$ up to the fixed additive constant N_0 , and all patch descriptions and all simulated circuits have size polynomial in N .

We now bound its error. By convexity of statistical distance and Equation (99),

$$\text{SD}(P_z, \mathbb{E}_{\gamma \leftarrow \nu_z} [P_{z, \gamma}]) \leq \mathbb{E}_{\gamma \leftarrow \nu_z} [\text{SD}(P_z, P_{z, \gamma})] \leq \frac{1}{N}. \quad (104)$$

By Equation (100),

$$\text{SD}(\mathbb{E}_{\gamma \leftarrow \nu_z} [P_{z, \gamma}], \mathbb{E}_{\gamma \leftarrow \hat{\nu}_z} [P_{z, \gamma}]) \leq \text{SD}(\nu_z, \hat{\nu}_z) \leq \text{negl}(N). \quad (105)$$

Finally, averaging Equation (101) over $\gamma \leftarrow \hat{\nu}_z$ gives

$$\text{SD}(\mathbb{E}_{\gamma \leftarrow \hat{\nu}_z} [P_{z, \gamma}], \mathcal{C}^\mathcal{O}(x, 1^k)) \leq \text{negl}(N). \quad (106)$$

Combining Equations (104)–(106), we obtain

$$\text{SD}(\mathcal{C}^\mathcal{O}(x, 1^k), P_z) \leq \frac{2}{N}. \quad (107)$$

By Equation (95), applied with accuracy $1/K$,

$$\text{SD}(P_z, D_x) \leq \frac{1}{K}. \quad (108)$$

Fix any $0 < \varepsilon \leq 1$ and set $k := \lfloor 1/\varepsilon \rfloor$. Moreover, $N \geq K$, and therefore the triangle inequality, Equations (107) and (108), and the definition of K give

$$\text{SD}\left(\mathcal{C}^{\mathcal{O}}\left(x, 1^{\lfloor 1/\varepsilon \rfloor}\right), D_x\right) \leq \frac{2}{N} + \frac{1}{K} \leq \frac{3}{K} \leq \frac{1}{4(k+1)} < \varepsilon. \quad (109)$$

Thus

$$\text{SampBQP}^{\mathcal{O}} \subseteq \text{SampBPP}^{\mathcal{O}}. \quad (110)$$

Proof of Item (4). Let $\text{Samp}^{\mathcal{O}}$ be any candidate classically-secure auxiliary-input distributional OWPuzz relative to $\mathcal{O}$. We construct a single uniform PPT oracle adversary $\mathcal{A}$ that breaks it.

Fix an auxiliary input x and write $n := |x|$. Define

$$\mu_x := \text{the output probability distribution of } \text{Samp}^{V^\pi, \text{QBF}}(x), \quad (111)$$

$$\mu_x^\gamma := \text{the output probability distribution of } \text{Samp}^{\gamma, \text{QBF}}(x) \quad (112)$$

for a patch γ .

Apply Lemma 5.2 to the algorithm $(x, b) \mapsto \text{Samp}^{V^\pi, \text{QBF}}(x)$, which ignores b , with the accuracy polynomial $r(n) := 4p(n)$, and let ν_x denote the output probability distribution of the patch sampled from $\mathcal{E}_{\text{Samp}, r}^{V^\pi, \text{QBF}}(x)$. Since the outputs are classical, the lemma gives, for every sufficiently large n and every $x \in \{0, 1\}^n$,

$$\mathbb{E}_{\gamma \leftarrow \nu_x} [\text{SD}(\mu_x, \mu_x^\gamma)] \leq \frac{1}{4p(n)}. \quad (113)$$

Moreover, exactly as in the proof of Item (3) (Equation (100)), there is a uniform PPT $^{\mathcal{O}}$ algorithm that samples a patch from a distribution $\hat{\nu}_x$ satisfying

$$\text{SD}(\nu_x, \hat{\nu}_x) \leq \text{negl}(n). \quad (114)$$

Because $\text{Samp}^{\gamma, \text{QBF}}(x)$ is a QPT algorithm querying QBF, there is a uniform PPT $^{\text{QBF}}$ algorithm Cond such that, for every n , every $x \in \{0, 1\}^n$, and every patch γ of length polynomial in n ,

$$\text{SD}\left((\text{puzz}, \text{Cond}(x, \gamma, \text{puzz}))_{(\text{puzz}, \text{ans}) \leftarrow \mu_x^\gamma}, \mu_x^\gamma\right) \leq 2^{-n}. \quad (115)$$

The adversary $\mathcal{A}_p^{\mathcal{O}}(x, \text{puzz})$ operates as follows: it samples $\gamma \leftarrow \hat{\nu}_x$, runs $\widehat{\text{ans}} \leftarrow \text{Cond}(x, \gamma, \text{puzz})$, and outputs $\widehat{\text{ans}}$. This is a single uniform PPT $^{\mathcal{O}}$ algorithm.

Fix a patch γ and let J_γ denote the distribution of $(\text{puzz}, \widehat{\text{ans}})$ when $(\text{puzz}, \text{ans}) \leftarrow \mu_x$ and $\widehat{\text{ans}} \leftarrow \text{Cond}(x, \gamma, \text{puzz})$. Let $\tilde{J}_\gamma$ denote the same distribution but with $(\text{puzz}, \text{ans}) \leftarrow \mu_x^\gamma$ instead. Then $\text{SD}(J_\gamma, \tilde{J}_\gamma) \leq \text{SD}(\mu_x, \mu_x^\gamma)$. Combining with Equation (115) and the triangle inequality,

$$\text{SD}(J_\gamma, \mu_x) \leq \text{SD}(J_\gamma, \tilde{J}_\gamma) + \text{SD}(\tilde{J}_\gamma, \mu_x^\gamma) + \text{SD}(\mu_x^\gamma, \mu_x) \leq 2 \text{SD}(\mu_x, \mu_x^\gamma) + 2^{-n}. \quad (116)$$

By convexity of statistical distance and Equation (116),

$$\text{SD}\left(\{(\text{puzz}, \mathcal{A}_p^\mathcal{O}(x, \text{puzz}))\}_{(\text{puzz}, \text{ans}) \leftarrow \text{Samp}^\mathcal{O}(x)}, \mu_x\right) \leq 2 \mathbb{E}_{\gamma \leftarrow \hat{\nu}_x} [\text{SD}(\mu_x, \mu_x^\gamma)] + 2^{-n} \quad (117)$$

$$\leq 2 \left(\mathbb{E}_{\gamma \leftarrow \nu_x} [\text{SD}(\mu_x, \mu_x^\gamma)] + \text{SD}(\nu_x, \hat{\nu}_x) \right) + 2^{-n} \quad (118)$$

$$\leq \frac{1}{2p(n)} + \text{negl}(n) < \frac{1}{p(n)} \quad (119)$$

for every sufficiently large n and every $x \in \{0, 1\}^n$. $\square$

7 Proof of Theorem 1.2

Apply Corollary 4.3 with the side oracle $\mathcal{H}$, and apply Lemma 5.2 with the side oracle $\mathcal{H}$. Each application gives a probability-one set of permutation families. Fix one permutation family π in the intersection of these probability-one sets, and define

$$\mathcal{O} := (V^\pi, \mathcal{H}). \quad (120)$$

Items (1) and (2) are shown in a similar way as the proof of Theorem 1.1.

We prove Item (3). Assume toward a contradiction that an auxiliary-input EFI pair (generator) G exists relative to $\mathcal{O}$, and let p be the positive polynomial guaranteed by Definition 2.7. Define

$$\rho_{b,x} := G^\mathcal{O}(x, b). \quad (121)$$

Below we construct a uniform QPT oracle distinguisher $\mathcal{D}$ that depends only on G and p , and then derive a contradiction from the infinite set $S_\mathcal{D}$ that Definition 2.7 guarantees for $\mathcal{D}$.

Apply Lemma 5.2 to the algorithm $(x, b) \mapsto G^{V^\pi, \mathcal{H}}(x, b)$ with the accuracy polynomial $r(n) := 16p(n)$. The lemma provides a random classical patch

$$\Gamma_x \leftarrow \mathcal{E}_{G,r}^{V^\pi, \mathcal{H}}(x), \quad (122)$$

sampled independently of the later challenge bit. For every realization γ of Γ_x , define the patched states

$$\sigma_{b,x}^\gamma := G^{\gamma, \mathcal{H}}(x, b). \quad (123)$$

Then for every sufficiently large n and every $x \in \{0, 1\}^n$,

$$\frac{1}{2} \sum_{b \in \{0,1\}} \mathbb{E}_{\gamma \leftarrow \Gamma_x} [\text{TD}(\rho_{b,x}, \sigma_{b,x}^\gamma)] \leq \frac{1}{16p(n)}. \quad (124)$$

Let $j(n)$ be a polynomial upper bound on the largest level of $\mathcal{H}$ queried by G on n -bit inputs. Such a bound exists because the level index is encoded in unary. For fixed (x, γ) , the patched generator $G^{\gamma, \mathcal{H}}(x, b)$ makes no query to V^π : the patch γ is an explicit finite set, and each patch query is implemented by an explicit reversible lookup circuit. The patched generator therefore uses only levels at most $j(n)$ of $\mathcal{H}$ and explicit gates, and so the two patched generators are valid circuit descriptions in a level- $(j(n) + 1)$ query to $\mathcal{H}$.

Construct a uniform QPT distinguisher $\mathcal{D}$ as follows. On input (x, τ) , it samples $\gamma \leftarrow \Gamma_x$ by running $\mathcal{E}_{G,r}^{V^\pi, \mathcal{H}}(x)$, and queries $\mathcal{H}^{(j(|x|)+1)}$ on the descriptions of the patched G , the classical data (x, γ) , the answer

bit initialized to 0, and the challenge state τ . After the query, $\mathcal{D}$ measures the answer bit, and outputs it. Let $M_{x,\gamma}$ be the projector onto the positive eigenspace of $\sigma_{1,x}^\gamma - \sigma_{0,x}^\gamma$, and set

$$e_{b,x,\gamma} := \text{TD} \left(\rho_{b,x}, \sigma_{b,x}^\gamma \right). \quad (125)$$

The distinguishing advantage satisfies

$$\Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{1,x}) \right] - \Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{0,x}) \right] = \mathbb{E}_\gamma \text{Tr} [M_{x,\gamma} (\rho_{1,x} - \rho_{0,x})] \quad (126)$$

$$\geq \mathbb{E}_\gamma \left[\text{TD} \left(\sigma_{0,x}^\gamma, \sigma_{1,x}^\gamma \right) - e_{0,x,\gamma} - e_{1,x,\gamma} \right] \quad (127)$$

$$\geq \text{TD} (\rho_{0,x}, \rho_{1,x}) - 2 \mathbb{E}_\gamma [e_{0,x,\gamma} + e_{1,x,\gamma}]. \quad (128)$$

Equation (124) implies

$$\mathbb{E}_\gamma [e_{0,x,\gamma} + e_{1,x,\gamma}] \leq \frac{1}{8p(n)}. \quad (129)$$

Combining Equations (128) and (129), for every sufficiently large n and every $x \in \{0, 1\}^n$,

$$\Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{1,x}) \right] - \Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{0,x}) \right] \geq \text{TD} (\rho_{0,x}, \rho_{1,x}) - \frac{1}{4p(n)}. \quad (130)$$

By the security of G applied to the distinguisher $\mathcal{D}$, there exist an infinite set $S_\mathcal{D} \subseteq \{0, 1\}^*$ and a negligible function negl such that Equations (24) and (25) hold for every $x \in S_\mathcal{D}$. Because there are only finitely many strings of each length, $S_\mathcal{D}$ contains strings of unbounded length. Hence, for every sufficiently long $x \in S_\mathcal{D}$, Equation (130) and the statistical farness Equation (24) give

$$\Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{1,x}) \right] - \Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{0,x}) \right] \geq \frac{1}{p(|x|)} - \frac{1}{4p(|x|)} = \frac{3}{4p(|x|)}, \quad (131)$$

while the computational indistinguishability Equation (25) gives

$$\Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{1,x}) \right] - \Pr \left[1 \leftarrow \mathcal{D}^\mathcal{O}(x, \rho_{0,x}) \right] \leq \text{negl}(|x|). \quad (132)$$

Since $3/(4p(|x|)) > \text{negl}(|x|)$ for all sufficiently large $|x|$, this is a contradiction.

Acknowledgements. TM is supported by JST CREST JPMJCR23I3, JST Moonshot R&D JPMJMS2061-5-1-1, JST FOREST, MEXT QLEAP, the Grant-in Aid for Transformative Research Areas (A) 21H05183, and the Grant-in-Aid for Scientific Research (A) No.22H00522.

AI Disclosure. We used Claude Fable 5 to assist with writing proofs and Introduction, and ChatGPT-5.6 Sol for proofreading. The authors verified the correctness and originality of all content including references.

References

- [AA11] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 333–342. ACM Press, June 2011. (Cited on page 4, 8.)

- [Aar14] Scott Aaronson. The equivalence of sampling and searching. *Theory of Computing Systems*, 55:281–298, 2014. (Cited on page 12.)
- [ABC⁺26] Akshima, Tyler Besselman, Kai-Min Chung, Siyao Guo, and Tzu-Yi Yang. Tight quantum time-space tradeoffs for permutation inversion. In Joan Daemen and Emmanuel Thomé, editors, *EUROCRYPT 2026, Part I*, volume 16541 of *LNCS*, pages 431–456. Springer, Cham, May 2026. (Cited on page 5, 13.)
- [ABK24] Scott Aaronson, Harry Buhrman, and William Kretschmer. A qubit, a coin, and an advice string walk into a relational problem. *ITCS*, 2024. (Cited on page 12.)
- [AC02] Mark Adcock and Richard Cleve. A quantum Goldreich-Levin theorem with cryptographic applications. In Helmut Alt and Afonso Ferreira, editors, *STACS 2002, 19th Annual Symposium on Theoretical Aspects of Computer Science*, volume 2285 of *Lecture Notes in Computer Science*, pages 323–334, Antibes - Juan les Pins, France, March 14–16, 2002. Springer, Heidelberg, Germany. (Cited on page 5, 14.)
- [AC17] Scott Aaronson and Lijie Chen. Complexity-theoretic foundations of quantum supremacy experiments. CCC’17: Proceedings of the 32nd Computational Complexity Conference, 2017. (Cited on page 8.)
- [AQY22] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 208–236. Springer, Cham, August 2022. (Cited on page 3.)
- [BBBV97] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh V. Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997. (Cited on page 7, 9, 19.)
- [BCQ23] Zvika Brakerski, Ran Canetti, and Luowen Qian. On the computational hardness needed for quantum cryptography. *ITCS 2023*, 2023. (Cited on page 3, 4, 12.)
- [BM09] Boaz Barak and Mohammad Mahmoody-Ghidary. Merkle puzzles are optimal - an $O(n^2)$ -query attack on any key exchange from a random oracle. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 374–390. Springer, Berlin, Heidelberg, August 2009. (Cited on page 9.)
- [BMS16] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd. Average-case complexity versus approximate simulation of commuting quantum computations. *Physical Review Letters*, 117:080501, 2016. (Cited on page 4, 8.)
- [IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *30th FOCS*, pages 230–235. IEEE Computer Society Press, October / November 1989. (Cited on page 3.)
- [Imp95] Russell Impagliazzo. A personal view of average-case complexity. In *Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995*, pages 134–147. IEEE Computer Society, 1995. (Cited on page 3.)
- [IR89] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *21st ACM STOC*, pages 44–61. ACM Press, May 1989. (Cited on page 9.)

- [JLS18] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 126–152. Springer, Cham, August 2018. (Cited on page 3.)
- [KQST23] William Kretschmer, Luowen Qian, Makrand Sinha, and Avishay Tal. Quantum cryptography in algorithmica. In Barna Saha and Rocco A. Servedio, editors, *55th ACM STOC*, pages 1589–1602. ACM Press, June 2023. (Cited on page 3.)
- [KQT25] William Kretschmer, Luowen Qian, and Avishay Tal. Quantum-computable one-way functions without one-way functions. In Michal Koucký and Nikhil Bansal, editors, *57th ACM STOC*, pages 189–200. ACM Press, June 2025. (Cited on page 3.)
- [Kre21] William Kretschmer. Quantum pseudorandomness and classical complexity. *TQC 2021*, 2021. (Cited on page 3.)
- [KT24] Dakshita Khurana and Kabir Tomer. Commitments from quantum one-wayness. In Bojan Mohar, Igor Shinkar, and Ryan O’Donnell, editors, *56th ACM STOC*, pages 968–978. ACM Press, June 2024. (Cited on page 3, 4, 5, 14.)
- [KT25] Dakshita Khurana and Kabir Tomer. Founding quantum cryptography on quantum advantage, or, towards cryptography from $\#p$ hardness. In Michal Koucký and Nikhil Bansal, editors, *57th ACM STOC*, pages 178–188. ACM Press, June 2025. (Cited on page 4.)
- [LMW24] Alex Lombardi, Fermi Ma, and John Wright. A one-query lower bound for unitary synthesis and breaking quantum cryptography. In Bojan Mohar, Igor Shinkar, and Ryan O’Donnell, editors, *56th ACM STOC*, pages 979–990. ACM Press, June 2024. (Cited on page 3, 8.)
- [MSY25] Tomoyuki Morimae, Yuki Shirakawa, and Takashi Yamakawa. Cryptographic characterization of quantum advantage. In Michal Koucký and Nikhil Bansal, editors, *57th ACM STOC*, pages 1863–1874. ACM Press, June 2025. (Cited on page 5.)
- [MY22] Tomoyuki Morimae and Takashi Yamakawa. Quantum commitments and signatures without one-way functions. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 269–295. Springer, Cham, August 2022. (Cited on page 3.)
- [MY24] Tomoyuki Morimae and Takashi Yamakawa. Quantum advantage from one-way functions. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part V*, volume 14924 of *LNCS*, pages 359–392. Springer, Cham, August 2024. (Cited on page 5.)
- [Wee06] Hoeteck Wee. Finding pessiland. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 429–442. Springer, Berlin, Heidelberg, March 2006. (Cited on page 5, 6, 8.)