

# Oracle Separation Between Quantum Commitments and Quantum One-wayness

John Bostanci<sup>1</sup>, Boyang Chen<sup>2</sup>, and Barak Nehoran<sup>3</sup>

<sup>1</sup>Columbia University

<sup>2</sup>Tsinghua University

<sup>3</sup>Princeton University

## Abstract

We show that there exists an oracle relative to which quantum commitments exist but no (efficiently verifiable) one-way state generators exist. Both have been widely considered candidates for replacing one-way functions as the minimal assumption for cryptography—the weakest cryptographic assumption implied by all of computational cryptography. Recent work has shown that commitments can be constructed from one-way state generators, but the other direction has remained open. Our results rule out any black-box construction, and thus settles this crucial open problem, suggesting that quantum commitments (as well as its equivalency class of EFI pairs, quantum oblivious transfer, and secure quantum multiparty computation) appear to be strictly weakest among all known cryptographic primitives.

**Note:** There is a bug in [Section 5](#), pointed out to us by Eli Goldin and Mark Zhandry. While [Lemma 5.5](#) is true, it only holds for fixed states  $\rho$ , and thus can not be applied to get an oracle separation between EFI pairs and OWSG in the Haar random swap model. The construction of OWPuzz and EFI can be lifted to the unitary model but the attack on OWSG cannot be lifted with the approach in this paper. Treating the swap oracle as a reflection around  $|0\rangle - |\psi\rangle$  for a Haar random  $|\psi\rangle$  allows one recover the result, see [\[16\]](#) for a more detailed discussion. We leave the buggy attack on OWSG in that section of the paper in the arXiv version of this paper for future reference.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Open Problems . . . . .	4
1.2	Related Work . . . . .	5
1.3	Concurrent Work . . . . .	5
<b>2</b>	<b>Technical Overview</b>	<b>6</b>
<b>3</b>	<b>Preliminaries</b>	<b>7</b>
3.1	Quantum Basics . . . . .	7
3.2	The Haar Measure . . . . .	7
3.3	Quantum Oracles and the Common Reference States . . . . .	8
3.4	Cryptographic Primitives . . . . .	8
3.5	Unitary Complexity Theory . . . . .	10
3.6	Quantum Learning Theory . . . . .	11
<b>4</b>	<b>Separation Between OWSG and EFI Pairs</b>	<b>11</b>
4.1	One-Way Puzzles in the Common Haar Random State Model . . . . .	13
<b>5</b>	<b>Separation in the Haar random swap oracle model</b>	<b>16</b>

# 1 Introduction

In classical cryptography, one-way functions (OWF) serve as a minimal assumption. That is to say, the existence of nearly any other classical cryptographic primitive implies the existence of one-way functions. Furthermore, many cryptographic primitives (termed “Minicrypt primitives”), such as pseudorandom generators (PRG), pseudorandom functions (PRF), secret-key encryption and authentication, digital signatures, efficient-far-indistinguishable distributions (EFID), and commitments, are equivalent to one-way functions.

Many of these Minicrypt primitives can be generalized to the setting of quantum states, producing fully quantum primitives such as pseudorandom unitaries (PRU) and pseudorandom states (PRS) [21], one-way state generators (OWSG) [29, 30], one-way puzzles (OWPuzz) [22], efficient-far-indistinguishable quantum state pairs (EFI) [8], and quantum bit commitments (QBC) [13]. A recent sequence of works has shown that although these primitives can be built from one-way functions [8, 21, 22, 29, 30], they may exist even if one-way functions do not [24–26]. The classical versions of these fully quantum primitives are known to be equivalent and jointly minimal for classical cryptography, and so a central question in quantum cryptography is whether the same is true for the quantum generalizations.

*What is the minimal computational assumption for quantum cryptography?*

In particular, one-way state generators (which generalize one-way functions) and quantum commitments (equivalent to EFI pairs by [8]) have received much attention, as the potential minimal assumptions for fully quantum cryptography, with most other such primitives implying one [5, 21, 29] or the other [3, 30, 32]. In the classical setting, one-way functions and classical commitments are equivalent and jointly minimal. This motivates the crucial question:

*Are commitments and one-wayness equivalent in the quantum setting?*

Recent work has given a partial answer by showing that quantum commitments can be constructed from OWSG [4, 22]. However, the other direction—showing whether OWSG can be constructed from quantum commitments—has remained open.<sup>1</sup> We resolve this by showing that OWSG cannot be constructed from quantum commitments in a black-box way. That is, we give a unitary quantum oracle relative to which quantum commitments exist, but every OWSG is insecure.

**Theorem 1.1 (informal).** *There is no black-box construction of (efficiently verifiable) one-way state generators from quantum bit commitments.*

As a direct consequence, we also rule out black-box constructions of a large collection of primitives that imply efficiently verifiable one-way puzzles and one-way state generators. In fact, our main theorem is stronger. Since our separating oracle allows us to build two stronger cryptographic primitives—(inefficiently verifiable) one-way puzzles, and a single-copy version of pseudorandom states—both of which are separately known to imply quantum commitments [22, 30], this gives us a stronger separation:

**Theorem 1.2 (informal).** *There is no black-box construction of (efficiently verifiable) one-way state generators from either single-copy pseudorandom states or (inefficiently verifiable) one-way puzzles.*

**The efficiency of verifying one-wayness.** In classical cryptography, one-wayness is inherently *verifiable*: that is, given a successful inversion of a one-way function, we can always run the function in the forward direction to check if the inversion was correct. On the other hand, the literature on *quantum* one-wayness distinguishes between efficiently verifiable [11, 29] and inefficiently verifiable (or “statistically verifiable”) [4, 22] versions. This is because there is no built-in way to verify the inversion of a quantum operation that traces

---

<sup>1</sup>Note that [4] shows that a variant of OWSG that allows verification to be inefficient *is*, in fact, equivalent to quantum commitments. However, it is not known how to construct the standard version of OWSG which requires verification to be efficient.

out some registers.<sup>2</sup> However, it has remained unclear whether these two versions of quantum one-wayness are—as is the case in the classical setting—in fact equivalent.<sup>3</sup>

*Can **efficiently verifiable** quantum one-wayness be constructed from **statistically verifiable** quantum one-wayness?*

As a corollary to our main theorem, we are able to answer this in the negative:

**Corollary 1.3 (informal).** *There is no black-box construction of either efficiently verifiable one-way state generators or efficiently verifiable one-way puzzles from either statistically verifiable one-way state generators or statistically verifiable one-way puzzles.*

In other words, in the quantum setting, one-wayness is *not* efficiently verifiable inherently. Efficiently verifiable one-wayness is a stronger assumption.

**Conceptual impact of our results.** Our results show that quantum commitments—together with their equivalence class of EFI pairs, quantum oblivious transfer, secure quantum multiparty computation, and statistically verifiable one-way state generators—are strictly weaker than nearly all other known computational cryptography. This motivates defining a new *world* in the spirit of Impagliazzo [20]. Impagliazzo defines five possible worlds, including Cryptomania (in which classical public-key cryptography exists), and Minicrypt (in which only one-way functions exist). (The three remaining worlds—Pessiland, Heuristica, and Algorithmica—do not allow classical cryptography.) The recent work on quantum cryptography has spoken of a Microcrypt, in which one-way functions do not exist, but pseudorandom unitaries and states exist (and consequently many other quantum cryptographic primitives).

We suggest the introduction of a new world to the Impagliazzo hierarchy, *Entanglementia*, a world in which only the bare minimum of (quantum) cryptography is possible, and the only secure computational cryptography that exists is the cryptography that is equivalent to quantum commitments. We propose the name *Entanglementia*<sup>4</sup> because it is a world in which the central cryptographic protocols—such as quantum commitments, oblivious transfer, and secure multiparty computation—seem to inherently require parties to maintain coherent entanglement between them. Specifically, verification in Entanglementia often requires a challenger to maintain a register that is coherently entangled with the adversary. Entanglementia primitives and assumptions that do not maintain entanglement—such as statistically verifiable OWSG and EFI—are inherently not efficiently verifiable.

## 1.1 Open Problems

We suggest the following open problems for future work:

1. Are there black-box constructions of pure-output OWSGs from efficiently verifiable mixed-output OWSGs? We show that there is no black-box construction of efficiently verifiable OWSGs from statistically verifiable OWSGs. This distinction between efficient/statistical verifiability is only meaningful for OWSGs that produce mixed states, since any OWSG that produces pure states can be efficiently verified using a SWAP test. This suggests that pure-output OWSGs are qualitatively different. Can a black-box reduction be ruled out?
2. This work suggests that EFI pairs, quantum commitments, and their equivalency class of Entanglementia primitives appear to be uniquely minimal among the known computational assumptions for (quantum) cryptography. Can EFI pairs be constructed from *all* of computational cryptography? Or are there computational assumptions that are even weaker than EFI but still useful for some cryptography?

<sup>2</sup>One-way state generators in their most general form can output mixed states, while one-way puzzles can sample a puzzle by measuring a quantum register. Both can be implemented by a unitary operation followed by the tracing out of some subset of registers.

<sup>3</sup>[11] observe that an oracle separation between efficiently verifiable and statistically verifiable one-way puzzles follows from [24]. However, the question for the more fundamentally quantum one-way state generators has remained open.

<sup>4</sup>The ending *-mentia* means “in the mind”.

3. Single-copy pseudorandom states with output longer than key (1PRS) are known to imply quantum commitments [30]. Furthermore, 1PRS appears to be a weak primitive: it can be built from OWF [30] or PRS [17], but *does not* imply OWF [25], PRS [10], or even OWSG [*this work*] in a black-box way. Similarly, (inefficiently verifiable) one-way puzzles (OWPuzz) are known to be implied by pure-output OWSG and imply quantum commitments [22]. However, the status of 1PRS and OWPuzz is unclear: Can 1PRS and OWPuzz be shown to be separated from commitments and therefore be stronger cryptographic primitives, or are they also contained in Entanglementia?

## 1.2 Related Work

Since the works of [24, 25], which showed that there is an oracle relative to which pseudorandom unitaries exist and one-way functions do not, there has been a large body of work identifying and separating candidates for a minimal assumption for quantum cryptography. Among these are pseudorandom states [21], EFI pairs [8], and one-way state generators [29]. [3] showed that pseudorandom states implied EFI pairs, and a number of other useful cryptographic primitives like secure multi-party computation and pseudo one-time encryption schemes. The work of [23] further showed that pure-output one-way state generators implied one-way puzzles which in turn implied EFI pairs and commitments. [4] further showed that mixed-output one-way state generators implied EFI by showing that if the efficient verification of one-way state generators was removed, they would become equivalent to EFI pairs, leaving the question of whether efficiently verifiable one-way state generators could be constructed from EFI pairs or one-way puzzles.

Oracle separations are widely developed for separating classical cryptographic primitives, but fewer techniques are known for separating cryptographic primitives in the fully quantum setting. [24] gave a quantum oracle relative to which pseudorandom unitaries exist but one-way functions do not. Relatedly, [25] shows a classical oracle relative to which single-copy pseudorandom states exist but one-way functions do not. [11] observes that the oracle of [24] can be used to separate one-way puzzles from efficiently verifiable one-way puzzles. [12] used a similar oracle to show that pseudorandom states can not be used to build digital signatures for classical messages with quantum public keys, and [15] uses a modified version of the oracle to separate quantum cryptographic primitives broken by a QCMA oracle from those broken by a PP oracle.

The common reference quantum state model, was introduced by [28, 31], where they show how to build EFI and quantum commitments. [2, 10] introduced the common Haar random state model, and [10] used this model to show that single-copy pseudorandom states do not imply multi-copy pseudorandom states in a black-box way.

## 1.3 Concurrent Work

Behera, Malavolta, Morimae, Mour, and Yamakawa [6] independently and concurrently demonstrate a result similar to ours. Similarly to our result, they show an oracle separation between quantum commitments and both OWSG and efficiently verifiable one-way puzzles. Our full set of results is, in some sense, incomparable. We additionally show that 1PRS is separated from OWSG, and they additionally show that primitives such as private-key quantum money are separated from QEFID pairs (*classical* EFI distribution pairs that are quantum-samplable). We note that they do not consider the common Haar random state model, instead defining a different quantum reference state, and therefore have different proof techniques.

## Acknowledgements

The authors thank Prabhanjan Ananth for helpful discussions about recent results on the common Haar random state model, Fermi Ma for suggesting a new interpretation of the main result of this paper, and Rahul Jain for giving the authors insights into the impact and implications of this work. The authors also thank Eli Goldin, Henry Yuen, and Mark Zhandry, for helpful discussions related to the Haar random swap oracle, and Amit Behera, Giulio Malavolta, Tomoyuki Morimae, Tamer Mour, and Takashi Yamakawa for their helpful discussions related to their concurrent work.

J.B. is supported by Henry Yuen’s AFORS (award FA9550-21-1-036) and NSF CAREER (award CCF2144219). B.C. acknowledges supported by National Key Research and Development Program of China (Grant No. 2023YFA1009403) and National Natural Science Foundation of China (Grant No. 12347104). This work was done in part while B.N. and J.B. were visiting the Simons Institute for the Theory of Computing, supported by NSF QLCI Grant No. 2016245.

## 2 Technical Overview

Our main technical contributions are a polynomial-space attack against one-way state generators relative to all quantum reference quantum state models, as well as a construction of one-way puzzles in the common Haar random state model. Together, they give our separation.

**Ruling out one-way state generators relative to common reference states.** To rule out one-way state generators relative to any common reference quantum state oracle, we notice that the quantum OR attack used in [10] can be extended to a so-called “threshold search” attack. A threshold search algorithm takes as input a set of  $m$  measurements  $M$  and  $O(\log^2 m \log n)$  copies of a quantum state, and outputs any measurement that has greater than  $1/3$  chance of accepting, promised that there exists one that is accepted with probability at least  $3/4$ . For one-way state generators, the measurement corresponding to  $k$  is to simply run verification with key  $k$  on  $O(\lambda)$  copies of the input state. By the correctness of the one-way state generator, the promise of threshold search is satisfied. Furthermore, because we are taking  $O(\lambda)$  copies of the input state, a measurement that accepts with probability  $1/3$  means that the input state passes verification with key  $k'$  with probability  $1 - O(1/\lambda)$ . Thus, this attack breaks the one-wayness of any one-way state generator.

To implement this attack in polynomial space, we observe that the algorithm from [33], combined with a space efficient pseudo-random generator from [14], provides a UnitaryPSPACE implementation of threshold search.

We further note that in some common reference state models, such as those of [28, 31], or the common Haar random state model of [2, 10], it has been shown that quantum commitments and EFI pairs exist even relative to adversaries that have unbounded computation—but a polynomial number of samples of the common reference state. Thus, for these reference states we arrive at a separation between EFI pairs and one-way state generators relative to state preparation oracles.<sup>5</sup>

**Constructing one-way puzzles in the common Haar random state model.** Our second result is to construct inefficiently verifiable one-way puzzles in the CHRS model, strengthening the results of [2, 10], which construct 1PRS and EFI pairs. Our construction goes as follows:

1. **Samp** <sup>$\{|\psi_\ell\rangle\}_\ell$</sup> ( $1^n$ ): **Samp** first samples a random  $n$ -bit string  $k$ . We make use of  $n$  independent Haar random states by taking the Haar random states from sizes  $n$  through  $2n$ .<sup>6</sup> Based on the  $(\ell - n)$ ’th bit of the key, **Samp** either applies a Pauli  $Z$  to the first qubit of  $|\psi_\ell\rangle$ , or does not (i.e.  $(Z_1^{k_{\ell-n}})|\psi_\ell\rangle$ ), and then takes the classical shadows of the resulting state. Let  $s$  be the collection of classical shadows generated for  $\ell$  from  $n$  to  $2n$ , then **Samp** outputs the pair  $(k, s)$ .
2. **Ver** <sup>$\{|\psi_\ell\rangle\}_\ell$</sup> ( $k, s$ ): **Ver** first performs tomography of  $|\psi_\ell\rangle$  for  $\ell$  from  $n$  through  $2n$ , then uses the classical shadows to estimate the value of the observable  $(Z_1^{k_{\ell-n}})|\psi_\ell\rangle\langle\psi_\ell|(Z_1^{k_{\ell-n}})$  on the puzzle, which is the overlap with the state that the sampler should have used to generate the classical shadows, if they output key  $k$ . If more than  $3n/4$  of the observables have value higher than  $1/2$ , the verifier accepts, otherwise it rejects.

<sup>5</sup>The result of [10] additionally constructs single-copy pseudo-random states (1PRS), so our results also imply a separation between them and one-way state generators.

<sup>6</sup>The Haar random state model provides a single Haar random state for every integer  $\ell$  (the equivalent of the input size for the oracle). We choose the set of sizes from  $n$  through  $2n$ , although any  $\omega(\log n)$  sufficiently large values would work in order to get negligible LOCC Haar indistinguishability.

For correctness, if the verifier receives a pair  $(k, s)$  that comes from the sampler, than the expected value of the observables is 1, so with high probability the shadow estimation outputs a value greater than  $1/2$ . To prove security, we first consider an adversary that does not sample the Haar random state at all. In this case, the states  $Z_1^{k_\ell - n} |\psi_\ell\rangle$  are also distributed according to the Haar measure, so the adversary outputs a fixed distribution over solutions to the puzzle, independent of the key  $k$ . Therefore, any adversary that does not request samples of the common Haar random state only passes verification with negligible probability. Finally, due to the LOCC Haar indistinguishability result of [2], an adversary that gets sample access to the common Haar random state outputs (almost) the same distribution over keys as an adversary that gets samples of an independently random family of states. However, this second adversary can be simulated without querying the common Haar random state family at all, and therefore must fail to pass verification.

### 3 Preliminaries

#### 3.1 Quantum Basics

For a bit string  $x \in \{0, 1\}^*$ , we denote by  $|x|$  its length (not its Hamming weight). When  $x$  describes an instance of a computational problem, we will often use  $\lambda = |x|$  to denote its size.

A function  $\delta : \mathbb{N} \rightarrow [0, 1]$  is an *inverse polynomial* if there exists a polynomial  $p$  such that  $\delta(n) \leq 1/p(n)$  for all sufficiently large  $n$ . A function  $\epsilon : \mathbb{N} \rightarrow [0, 1]$  is *negligible* if for every polynomial  $p(n)$ , for all sufficiently large  $n$  we have  $\epsilon(n) \leq 1/p(n)$ .

A *register*  $R$  is a named finite-dimensional complex Hilbert space. If  $A, B, C$  are registers, for example, then the concatenation  $ABC$  denotes the tensor product of the associated Hilbert spaces. We abbreviate the tensor product state  $|0\rangle^{\otimes n}$  as  $|0^n\rangle$ . For a linear transformation  $L$  and register  $R$ , we write  $L_R$  to indicate that  $L$  acts on  $R$ , and similarly we write  $\rho_R$  to indicate that a state  $\rho$  is in the register  $R$ . We write  $\text{Tr}[\cdot]$  to denote trace, and  $\text{Tr}_R[\cdot]$  to denote the partial trace over a register  $R$ .

We denote the set of linear transformations on  $R$  by  $L(R)$ , and linear transformations from  $R$  to another register  $S$  by  $L(R, S)$ . We denote the set of positive semidefinite operators on a register  $R$  by  $\text{Pos}(R)$ . The set of density matrices on  $R$  is denoted  $S(R)$ . For a pure state  $|\varphi\rangle$ , we write  $\varphi$  to denote the density matrix  $|\varphi\rangle\langle\varphi|$ . We denote the identity transformation by  $\text{id}$ . For an operator  $X \in L(R)$ , we define  $\|X\|_\infty$  to be its operator norm, and  $\|X\|_1 = \text{Tr}[|X|]$  to denote its trace norm, where  $|X| = \sqrt{X^\dagger X}$ . We write  $\text{td}(\rho, \sigma) = \frac{1}{2}\|\rho - \sigma\|_1$  to denote the trace distance between two density matrices  $\rho, \sigma$ , and  $F(\rho, \sigma) = \|\sqrt{\rho}\sqrt{\sigma}\|_1^2$  for the fidelity between  $\rho, \sigma$ .

#### 3.2 The Haar Measure

Here we state the definition of the Haar measure and Haar random states.

**Definition 3.1** (Haar measure and Haar random states). *The Haar measure is the unique left- and right-invariant probability measure on the unitary group  $U(d)$ . A Haar random state is sampled by applying a unitary sampled from the Haar measure to  $|0\rangle$  (although any initial state would yield the same distribution over states). We use the notation  $\text{Haar}(d)$  to refer to the distribution over  $d$ -dimensional states drawn from the Haar measure.*

One of the most useful properties of the Haar measure is its concentration properties.

**Lemma 3.2** (Concentration of Haar measure [27]). *Let  $N \in \mathbb{N}$ . Let  $\mu$  be the Haar measure on dimension  $N_i$ , and let  $\mu$  be the Haar measure on a  $N$ -dimensional space. Let  $f$  be  $L$ -Lipshitz function in the Frobenius norm, mapping  $N$ -dimensional unitaries to real numbers. Then the following holds for every  $t > 0$ :*

$$\Pr_{U \leftarrow \mu} \left[ f(U) \geq \mathbb{E}_{V \leftarrow \mu} [f(V)] + t \right] \leq \exp \left( -\frac{(\min_i \{N_i\} - 2)t^2}{24L^2} \right).$$

We use this for one important corollary, which is the following.



**Corollary 3.3** (Haar random states on trace 0 observables). *Let  $|\psi\rangle$  be a  $n$ -qubit Haar random state and  $O$  be a trace 0 observable. Then the following holds:*

$$\Pr_{|\psi\rangle \leftarrow \text{Haar}(2^n)} \left[ \langle \psi | O | \psi \rangle \geq 2^{-\sqrt{n/2}} \right] \leq \exp \left( -\frac{2^{n/2} - 2}{96} \right).$$

*Proof.* We note that we can equivalently phrase this probability as being over a Haar random unitary, and the state  $U|0\rangle$ . The function  $f(U) = \langle 0 | U^\dagger O U | 0 \rangle$  is a degree 2-polynomial in  $U$ , so  $f$  is 2-Lipshitz in the Frobenius norm. Applying [Lemma 3.2](#) to this function and  $t = 2^{-\sqrt{n/2}}$  yields the desired result.  $\square$

### 3.3 Quantum Oracles and the Common Reference States

**Definition 3.4** (Quantum oracle access). *Let  $f : \{0,1\}^n \mapsto \{0,1\}$  be a classical Boolean function, a quantum query algorithm  $\mathcal{A}^{(\cdot)}$  queries  $f$  via access to a unitary  $U_f$  that acts as*

$$U_f |x\rangle |b\rangle \mapsto |x\rangle |b \oplus f(x)\rangle.$$

Typically quantum oracles must be unitary transformations, however recently a new model of “isometry” oracles has appeared in the literature [\[2, 10\]](#). This model, which essentially allows access to a very specific quantum resource (a single quantum state and no way to un-compute it) has been shown to allow for oracle separations between EFI pairs and PRS. However, the question of boosting these separations to standard oracles remains an open question.

**Definition 3.5** (Common reference quantum state (CRQS) [\[28, 31\]](#)). *The common reference quantum state (CRQS) model is an isometry that can be accessed as a quantum oracle. Let  $\mathcal{V} = \{V_m\}_{m \in \mathbb{N}}$ , with  $V_m : \mathbb{C} \mapsto \mathbb{C}^{2^m}$  be a family of isometries such that*

$$V_m |\alpha\rangle \mapsto |\alpha\rangle |\phi_m\rangle,$$

*We call  $\{|\phi_m\rangle\}$  the family of common reference quantum states.<sup>7</sup>*

A special case of the common reference state model is when the reference state family is drawn uniformly at random from the Haar measure.

**Definition 3.6** (Common Haar random state model (CHRS) [\[2, 10\]](#)). *The common Haar random state (CHRS) model is a CRQS isometry where every  $|\psi_m\rangle$  is drawn from  $\text{Haar}(2^m)$ .*

Secure EFI pairs (and thus quantum commitments) as well as 1PRS are known to exist relative to a CHRS oracle [\[2, 10\]](#).

### 3.4 Cryptographic Primitives

Here we define a number of cryptographic primitives, as well as their instantiation in the common Haar random state model.

**Definition 3.7** (Efficient, far, indistinguishable pairs [\[8\]](#)). *A family of pairs of quantum states  $\{(\rho_{0,\lambda}, \rho_{1,\lambda})\}_\lambda$  is an EFI pair if*

- (Efficiently preparable) *There exists a family of polynomial-size, time efficient quantum circuits  $\{C_\lambda\}_\lambda$  such that  $\text{Tr}_B[C_\lambda |b0\rangle] = \rho_{b,\lambda}$ .*

---

<sup>7</sup>[\[28, 31\]](#) require that the common reference quantum states are efficiently preparable by a third-party setup algorithm. Since the states in our setting are prepared by an oracle, we do not make this requirement. Because of this, while the quantum auxiliary input model and CRQS models of [\[28, 31\]](#) are incomparable, both of their models are special cases of the CRQS model by our definition.



- (Statistically far) There exists a negligible function  $\mu$  such that

$$\text{td}(\rho_{0,\lambda}, \rho_{1,\lambda}) \geq 1 - \mu(\lambda).$$

- (Computationally indistinguishable) There exists a negligible function  $\nu$  such that for all polynomial-time quantum adversaries  $A$ ,

$$|\Pr[A(\rho_{0,\lambda}) = \top] - \Pr[A(\rho_{1,\lambda}) = \top]| \leq \nu(\lambda).$$

**Definition 3.8** (EFI pairs in the CRQS model). *For an EFI pair in a CRQS model, the circuits for preparing the EFI pair has access to the common reference quantum state oracle, and the computational indistinguishability holds for all adversaries who have query access to the common reference quantum state oracle.*

*Remark 3.9.* We do not formally define quantum bit commitments here, as they are equivalent to EFI pairs by [8], and this equivalence extends to the unitary oracle setting, which is our *ultimate* goal. Interestingly, whether this equivalence holds in under all CRQS models is not clear, since the formal equivalence between different flavors of commitments requires the uncomputing of quantum circuits, which cannot necessarily be done in CRQS models. More specifically it is not known if computationally binding and statistically hiding commitments are equivalent to statistically binding and computationally hiding commitments in general in CRQS models (where by “computational” security we mean secure against adversaries that receive only a polynomial number of copies of the CRQS). This is of course not an issue for the headline results of our paper, since we ultimately upgrade all our results to a unitary oracle model, in which the equivalences hold once more. Furthermore, EFI pairs in CRQS models are equivalent to computationally hiding commitments by the techniques of [28, 31] (which extend those of [8] to CRQS models). For simplicity, we will therefore mainly only refer to EFI pairs for the rest of this paper.

Next we define a one-way state generator, using the definition from [29].

**Definition 3.10** (One-way state generators [29, 30]). *A one-way state generator (OWSG) is a collection of QPT algorithms (KeyGen, StateGen, Ver) such that*

- KeyGen takes as input the security parameter  $1^\lambda$  and outputs a classical key  $k \in \{0, 1\}^\kappa$ .
- StateGen takes as input a classical key  $k$  and outputs a  $m$ -qubit quantum state  $\rho_k$ .
- Ver takes as input a classical key  $k$  and quantum state  $\rho$  and outputs either  $\perp$  or  $\top$ .

A OWSG satisfies correctness if for all  $\lambda$ ,

$$\Pr \left[ \text{Ver}(k, \rho_k) \text{ accepts} : \begin{array}{l} k \leftarrow \text{KeyGen}(1^\lambda) \\ \rho_k \leftarrow \text{StateGen}(k) \end{array} \right] \geq 1 - \text{negl}(\lambda).$$

A OWSG satisfies one-way security if for all polynomial-time quantum adversaries  $A$  and polynomials  $t$ ,

$$\Pr \left[ \text{Ver}(k, \rho_{k'}) \text{ accepts} : \begin{array}{l} k' \leftarrow \text{KeyGen}(1^\lambda) \\ \rho_{k'} \leftarrow \text{StateGen}(k') \\ k \leftarrow A(1^\lambda, \rho_{k'}^{\otimes t}) \end{array} \right] \leq \text{negl}(\lambda).$$

Note that we follow the convention set by the existing literature and define OWSGs to have efficient verification, but not necessarily with pure-state outputs.

**Definition 3.11** (One-way state generators in the CRQS model). *For a one-way state generator in the CRQS model, both KeyGen and StateGen have access to the common reference quantum state oracle, and one-way security holds relative to all polynomial-time quantum adversaries that have access to the common reference quantum state oracle.*

**Definition 3.12** (One-way puzzle [22]). A one-way puzzle (OWPuzz) is a pair of quantum algorithms (Samp, Ver) such that

1. Samp takes as input a security parameter  $1^\lambda$  and outputs a pair of classical strings  $(k, s)$ , where  $k \in \{0, 1\}^\lambda$ . Samp must be efficient.
2. Ver takes as input a pair  $(k, s)$  and outputs either  $\perp$  or  $\top$ .

A OWPuzz satisfies correctness if for all  $\lambda$ ,

$$\Pr [\text{Ver}(k, s) \text{ accepts} : (k, s) \leftarrow \text{Samp}(1^\lambda)] \geq 1 - \text{negl}(\lambda).$$

A OWPuzz satisfies security if for all polynomial-time quantum adversaries  $A$

$$\Pr [\text{Ver}(A(s), s) \text{ accepts} : (k, s) \leftarrow \text{Samp}(1^\lambda)] \leq \text{negl}(\lambda).$$

We require that Samp is efficient (QPT), but Ver may be efficient (efficiently verifiable one-way puzzle) or inefficient (statistically verifiable one-way puzzle). We use the convention from prior work of using OWPuzz to mean the inefficiently verifiable version, but we specify which version we mean when we believe it may not be clear from context.

**Definition 3.13** (One-way puzzles with sample-efficient verifier in the CRQS model). For a one-way puzzle with sample-efficient verifier in the CRQS model, both Samp and Ver can make polynomial many calls to the common reference quantum state oracle, and security holds relative to all polynomial-time quantum adversaries that have access to the common reference quantum state oracle.

**Definition 3.14** (One-way puzzles with sample-inefficient verifier in the CRQS model). A one-way puzzle with sample-inefficient verifier is a pair of sampling and verification algorithms (Samp, Ver) with the same syntax as Definition 3.12, except that  $\text{Ver}(k, s, \bigotimes_{i=1}^r |\psi_i\rangle^{\otimes r}) \rightarrow \top/\perp$  is a time-unbounded algorithm that on input of any pair classical strings  $(k, s)$  halts and outputs  $\top/\perp$ , where  $r = r(n)$  can be arbitrarily function of  $n$ .

Note that since Ver is allowed to be unbounded in a one-way puzzle, there is a distinction between one-way puzzles and sample-efficient one-way puzzles in the CRQS model. In order to truly rule out one-way puzzles in a CRQS model, one should rule out one-way puzzles in the CRQS model that have unbounded query access to the reference state.

**Definition 3.15** (Single-copy pseudo-random states [30]). A single-copy pseudo-random states generator 1PRS is a QPT algorithm Gen that takes as input a key  $k \in \{0, 1\}^\lambda$  of length  $\lambda$  and outputs a pure state  $|\psi_k\rangle$  on  $m(\lambda) > \lambda$  qubits.

A 1PRS satisfies the pseudo-randomness property if for all polynomial-time quantum adversaries  $A$  and  $\lambda \in \mathbb{N}$ ,

$$\left| \Pr_{k \leftarrow \{0, 1\}^\lambda} [A(|\psi_k\rangle) \text{ accepts}] - \Pr_{|\psi\rangle \leftarrow \text{Haar}(2^{m(\lambda)})} [A(|\psi\rangle) \text{ accepts}] \right| \leq \text{negl}(\lambda).$$

**Definition 3.16** (Single copy pseudo-random states in the CRQS model). For a single-copy pseudo-random state in the CRQS model, Gen has access to the common reference quantum state oracle, and pseudo-randomness holds relative to all polynomial-time quantum adversaries that have access to the common reference quantum state.

### 3.5 Unitary Complexity Theory

**Definition 3.17** (unitaryPSPACE [7]). A unitary synthesis problem  $(U_x)_x$  is in UnitaryPSPACE if for all polynomials  $p(n)$ , there exists a uniform polynomial-space algorithm  $C$  that implements  $(U_x)_x$  with worst-case error  $1/p(n)$ .

An important fact about unitaryPSPACE is that it has a complete problem (under Turing reductions), the succinct Uhlmann transformation problem [7]. By oracle access to unitaryPSPACE, we therefore mean oracle access to the succinct Uhlmann transformation unitary synthesis problem, or any other unitaryPSPACE-complete problem.

### 3.6 Quantum Learning Theory

In this section we review results and definitions from quantum learning theory that will be relevant to our result.

**Definition 3.18** (Threshold search [9]). *Let  $\{M_i\}_{i \in [m]}$  be a collection of 2-outcome measurements. Let  $\rho$  be an unknown quantum state with the promise that there exists an index  $i$  such that*

$$\text{Tr}[M_i \rho] \geq 3/4.$$

*The threshold search problem is to output a measurement  $M_j$  such that  $\text{Tr}[M_j \rho] \geq 1/3$ .*

**Theorem 3.19** (Random threshold search [33]). *There is an algorithm that uses  $O(\log^2(m))$  space and samples of  $\rho$ , has expected time  $O(m)$ , and makes intermediate measurements, that solves the threshold search problem with constant probability.*

The algorithm from [33] is to fix a threshold  $\theta \in [0.4, 0.6]$ , and to repeatedly measure a threshold measurement for a randomly sampled  $M_i$  on  $\log^2(m)$  copies of the state, testing whether more than a  $\theta$  percent of the measurements accepted.

The following theorem allows us to simulate this algorithm in `unitaryPSPACE`.

**Theorem 3.20** (unitaryPSPACE-simulation [14]). *Every quantum algorithm that runs in time  $T$  with space  $S \geq \log(T)$  with unitary operations and intermediate unital<sup>8</sup> measurements can be simulated by a quantum algorithm of time  $T \cdot S^2 \text{poly log}(S)$  and space  $O(S \cdot \log T)$  with only unitary operations and no intermediate measurements.*

This allows us to simulate the threshold search algorithm in `UnitaryPSPACE`.

*Remark 3.21.* The result of [14] is proven by providing an unconditional pseudo-random generator with small seed that can be implemented in `PSPACE`. Since the algorithm of [33] just samples  $O(m)$  random measurements and applies controlled versions of them on the input state (controlled on a single output register that indicates whether any of the previous measurements accepted), it could be applied directly to threshold search to get a simpler simulation of this algorithm.

This also means that a polynomial-time query algorithm can easily generate a succinct circuit representing the algorithm, and we really need *only* an oracle to a `unitaryPSPACE`-complete problem, *not* the ability to perform any `PSPACE` computation ourselves. This is in contrast to previous work on this subject, which provided an oracle called a “QSPACE” oracle, which essentially made all parties polynomial space computations with intermediate measurements.

## 4 Separation Between OWSG and EFI Pairs

In this section, we show that there exists an oracle in any common reference state model such that no one-way state generator exists. Note that, as stated before, [2, 10] already showed that in the common Haar random state model, quantum commitments and EFI pairs exist (additionally, they show that 1PRS exist), and that the security cannot be broken by an adversary of any complexity that only has access to polynomial copies of the common Haar state. Together, this will imply an oracle in the common reference state model relative to which EFI pairs exist but one-way state generators do not.

**Theorem 4.1.** *Let  $\mathcal{V}$  be a common reference quantum state oracle for any family of reference states. Relative to  $(\mathcal{V}, \text{unitaryPSPACE})$ , efficiently verifiable one-way state generators do not exist.*

<sup>8</sup>Unital measurements are those that output a collapsed quantum state but no classical outcome. Thus they send the fully mixed state to the fully mixed state. In particular, they do not allow discarding a quantum state nor resetting it to a fixed state. This limited kind of measurement, however, is sufficient for our setting in which we only need to sample randomness.

*Proof.* Let  $(\text{KeyGen}, \text{StateGen}, \text{Ver})$  be a OWSG. In the common Haar random state model, we can assume that  $\text{Ver}$  works by calling some quantum circuit (one for each key  $k$ ) on input  $|\psi\rangle_{\text{A}} \otimes (|\phi_1\rangle^{\otimes s} \dots |\phi_s\rangle^{\otimes s})_{\text{B}}$  for some  $s = \text{poly}(\lambda)$ , and then measuring a bit in the computational basis. Let  $U_k$  be said circuit and consider the following operator that acts on the input for  $U_k$  (registers AB), and a copy of a OWSG state in register C. We define the measurement  $\Pi_k$  as follows

$$\Pi_k = \left( ((U_k^\dagger)_{\text{CD}}) (|1\rangle\langle 1|_{\text{C}} \otimes \text{id}_{\text{B}}) ((U_k)_{\text{AB}}) \right)^{\otimes 10\lambda}.$$

**Claim 4.2.** *On quantum input  $(|\psi_k\rangle \otimes |\phi_1\rangle^{\otimes s} \dots |\phi_s\rangle^{\otimes s})^{\otimes 10\lambda}$ ,  $\Pi_k$  accepts with probability  $1 - \text{negl}(\lambda)$ .*

*Proof.* By the correctness of the OWSG, running  $\text{Ver}$  on a copy of  $|\psi_k\rangle$  and key  $k$  accepts with probability  $1 - \text{negl}(\lambda)$ . Since the probability that  $\Pi_k$  accepts is the probability that  $10\lambda$  many verifiers (run in parallel) accept, its accept probability is given by

$$(1 - \text{negl}(\lambda))^{\otimes 10\lambda} \geq 1 - 10\lambda \cdot \text{negl}(\lambda) = 1 - \text{negl}(\lambda). \quad (1)$$

Here the second line is Bernoulli's inequality. This completes the proof of the completeness of the algorithm.  $\square$

**Claim 4.3.** *If  $\Pi_k$  accepts with probability  $\geq 1/3$  for some key  $k$ , then the probability that verification accepts is at least  $1 - \frac{1}{5\lambda}$ .*

*Proof.* Let  $p$  be the probability that  $\text{Ver}$  accepts when given key  $k$  and state  $|\psi\rangle$ . Then it is clear to see that the probability  $\Pi_k$  accepts on the state  $|\psi\rangle^{\otimes 10\lambda}$  is

$$p^{10\lambda} \geq \frac{1}{3}.$$

Solving for  $p$ , we see that

$$\begin{aligned} p &\geq e^{-\frac{\ln(3)}{10\lambda}} \\ &\geq 1 - \frac{\ln(3)}{10\lambda} \\ &\geq 1 - \frac{1}{5\lambda}. \end{aligned}$$

Here we use the inequality  $1 - x \leq e^{-x}$ , and then we use the fact that  $\ln(3) \leq 2$ . This completes the proof that the algorithm always provides a key that violates one-way state generator security.  $\square$

The algorithm for breaking a one-way state generator is to run threshold search on  $O(\lambda^2)$  many copies of the input state, and return the key corresponding to the measurement that threshold search outputs. Note that we need  $O(\lambda)$  for every  $\Pi_k$  and threshold search requires  $O(\log(m)) = O(\lambda)$  copies of the input state, which is itself  $O(\lambda)$  copies of the one-way state generator state, to run.

From the first claim, the promise of threshold search is met, so threshold search outputs a key such that  $\text{Tr}[\Pi_k \rho] \geq 1/3$  with constant probability. From the second claim, we know that the key will be accepted by the verifier with probability at least  $1 - \frac{1}{5\lambda}$ , which contradicts one-way state generator security. Finally, as noted in [Remark 3.21](#), the threshold search algorithm can be implemented with oracle access to a unitaryPSPACE-complete problem. This completes the proof of [Theorem 4.1](#).  $\square$

We can apply our attack to the common Haar random state model to get an oracle separation between 1PRS and one-way state generators.

**Theorem 4.4.** *In the common Haar random state model, 1PRS exists and one-way state generators do not.*

*Proof.* [2, 10] prove that in the common Haar reference state model, 1PRS exists, and consequently EFI pairs exist as well, relative to all polynomial-sample adversaries (with unbounded computation otherwise). Since the common Haar reference state model is a CRQS model, Theorem 4.1 implies that one-way state generators do not exist.  $\square$

We also note that since our attack against OWSGs works in *all* common reference quantum state models, a weaker separation, between EFI pairs and one-way state generators, can be similarly attained with a deterministic oracle if one instead takes the quantum auxiliary input model from [28, 31].<sup>9</sup>

## 4.1 One-Way Puzzles in the Common Haar Random State Model

In this section, we provide a construction of inefficiently verifiable one-way puzzles in the common Haar random state model. We also note that our adversary (from Theorem 4.1) breaks all verification-efficient one-way puzzles in the common Haar random state model.

**Corollary 4.5.** *Let  $\mathcal{V}$  be a common reference quantum state oracle for any family of reference states. Relative to  $(\mathcal{V}, \text{UnitaryALL})$ , sample-efficient one-way puzzles do not exist.*

*Proof.* The adversary for one-way puzzles is similar to the adversary from Theorem 4.1, except that instead of requiring  $10\lambda$  copies of  $|\psi_k\rangle$ , it simply copies  $s$  into  $10\lambda$  registers and runs threshold search on  $\{\Pi_k\}$ . The same proof shows that the adversary will retrieve a key that is accepted by the verifier with non-negligible probability.

Formally, to get an adversary that calls an oracle, we can define the “inefficient one-way puzzle verification” problem, where the instance is a description of the one-way puzzle, the input is a classical pair of strings  $[k, s]$ , and copies of the common reference quantum state, and the output is the result of the verification. Since verification exists, this is a problem in **unitaryALL**. We further note that our adversary uses the same amount of space as the verifier for the one-way puzzle does, but since verification for a one-way puzzle is not required to be space efficient, our verifier might not be. If the verifier happens to be polynomial space, this adversary will also be polynomial space.  $\square$

As noted in the discussion, sample-efficient one-way puzzles were already ruled out by the LOCC indistinguishability results of [2], but we believe our proof is simpler and thus might be of independent interest to the reader.

On the contrary, inefficiently-verifiable OWPuzz exist in the CHRS model, as formally states in the following theorem.

**Theorem 4.6.** *Let  $\mathcal{V}_{\text{Haar}}$  be the common reference quantum state oracle for Haar random states. Relative to  $(\mathcal{V}_{\text{Haar}}, \text{UnitaryALL})$ , inefficiently verifiable one-way puzzles exist.*

We now present our construction of Theorem 4.6. The construction of OWPuzz relies on the classical shadow tomography [19], so here we first describe how to sample classical shadows of a state and the theoretical guarantee of the classical shadows algorithm. Assume we have  $N$  copies of a state  $\rho$  and we want to estimate the observables  $O_1, \dots, O_M$  with the copies of  $\rho$ . One can do this by just doing independent random Clifford measurements over all the copies of  $\rho$ . Namely, we can sample random Clifford unitaries  $C_1, \dots, C_N$ , and then measure  $C_i \rho C_i^\dagger$  and record their measurement results  $b_1, \dots, b_N$ . We call the collection of  $(b_i, C_i)$  the classical shadows of the state  $\rho$ , denoted as  $\text{ShadowGen}(\rho, N)$ , which can be sampled without prior knowledge of the observables. The following lemma show that the classical shadows can be used to estimate the probabilities  $\text{Tr}[\Pi_i \rho]$ .

**Lemma 4.7** (Classical shadow tomography, adapted from [19]). *For any  $n$ -qubit observables  $O_1, \dots, O_M$ , and accuracy parameter  $\epsilon, \delta \in [0, 1]$ . Let  $N \geq \frac{204}{\epsilon^2} \log(2M/\delta) \max_{1 \leq i \leq M} \text{Tr}[O_i^2]$ . Then for any  $n$ -qubit state  $\rho$ , let  $\text{ShadowGen}(\rho, N)$  be the classical shadow tomography of  $N$  copies of  $\rho$ . Then there is a time-unbounded*

<sup>9</sup>The quantum auxiliary input model, as defined in [28, 31] is a special type of CRQS model (by our definition), in which the common reference quantum state for each value of the security parameter (or input size) is a predetermined *fixed* state.

classical algorithm that can give an estimation  $\hat{o}_i$  on all the observables  $\text{Tr}[O_i \rho]$  given  $\text{ShadowGen}(\rho, N)$  such that

$$|\hat{o}_i - \text{Tr}[O_i \rho]| \leq \epsilon \quad \forall i \in [1, M],$$

with probability at least  $1 - \delta$ .

Now we can describe the construction of one-way puzzles in the CHRS model:

- The sampler **Samp** takes 10000 copies of each  $l$ -qubit CHRS state for  $n \leq l \leq 2n$ . For each  $l \in [n, 2n]$ , the sampler chooses a random bit  $k_l \in \{0, 1\}$  and then generates the classical shadow of  $(Z_1^{k_l} \otimes \text{id}) |\psi_l\rangle$  using the 10000 copies of the state. The sampler will output  $(k_n \dots k_{2n})$  as the key and  $\text{ShadowGen}((Z_1^{k_l} \otimes \text{id}) |\psi_l\rangle\langle\psi_l| (Z_1^{k_l} \otimes \text{id}))$  as the puzzle.
- The verifier **Ver** runs the algorithm from Lemma 4.7 for estimating observables for all  $l \in [n, 2n]$ , for the observable  $(Z_1^{k_l} \otimes \text{id}) |\psi_l\rangle\langle\psi_l| (Z_1^{k_l} \otimes \text{id})$  and state  $\rho = (Z_1^{k_l} \otimes \text{id}) |\psi_l\rangle\langle\psi_l| (Z_1^{k_l} \otimes \text{id})$ . The verifier accepts if for at least  $3n/4$  different  $l$ , the estimation of the corresponding observable is greater than  $1/2$ . The estimation of  $(Z_1^{k_l} \otimes \text{id}) |\psi_l\rangle\langle\psi_l| (Z_1^{k_l} \otimes \text{id})$  needs the classical description of  $|\psi_l\rangle$ , which can be done if the verifier has access to exponentially many copies of  $|\psi_l\rangle$ .

**Theorem 4.8** (One-way puzzle correctness). *On input  $(k, s) \leftarrow \text{Samp}_l^{\{|\psi_l\rangle\}}$ , **Ver** accepts with probability  $1 - \text{negl}(n)$ .*

*Proof.* According to Lemma 4.7 with  $\epsilon = 1/3$  and  $\delta = 1/10$ , the probability that the estimate of  $Z_1^{k_l} \otimes \text{id}$  deviates by at least  $1/2$  is less than  $1/10$  when the number of copies of the classical shadow is  $N \geq 204.9 \cdot \log(2 \cdot 100)$ , for which 10000 copy is enough. Since  $(Z_1^{k_l} \otimes \text{id}) |\psi_l\rangle$  measures 1 on  $(Z_1^{k_l} \otimes \text{id}) |\psi_l\rangle\langle\psi_l| (Z_1^{k_l} \otimes \text{id})$ , the probability that the estimation is lower than  $1/2$  is at most  $1/10$ . Thus by the Chernoff bound over all  $n$  keys, the probability that a puzzle is not accepted by the verifier (i.e. more than  $1/4$  of the puzzles have an estimate lower than  $1/2$ ) is exponentially small, which shows **OWPuzz** correctness.  $\square$

At a high level, the security proof will begin by first showing that an adversary  $\mathcal{A}$  that receives no copies of the common Haar random state succeeds with probability negligible in  $n$ . Then, using the LOCC Haar indistinguishability lemma from [2], we will show that an adversary that receives  $\text{poly}(n)$  copies of the Haar random state can not distinguish between the case when they have the same common Haar random state as the sampler, and when they have independently sampled Haar random states. As the second case can be simulated without sampling any copies of the Haar random state by an exponential time adversary that samples Haar random states and runs  $\mathcal{A}$  on them, the probability that the adversary passes **Ver** must be negligible in  $n$ . Formally, we have the following.

**Lemma 4.9.** *Let  $\mathcal{A}$  be an adversary that gets no samples of the common Haar random state  $\{|\psi_l\rangle\}_l$ , then  $\mathcal{A}(s)$ ,  $s \leftarrow \text{Samp}^{\{|\psi_l\rangle\}_l}$ , passes verification with probability that is negligible in  $n$ .*

*Proof.* In order to prove the theorem, we note that by the definition of the Haar measure as the left and right invariant measure on states, for all choices of  $k_n, \dots, k_{2n}$ ,

$$\begin{aligned} \mathbb{E}_{\{|\psi_l\rangle\}_l} & \left[ |\psi_n\rangle\langle\psi_n|^{\otimes 10000} \otimes \dots \otimes |\psi_{2n}\rangle\langle\psi_{2n}|^{\otimes 10000} \right] \\ &= \mathbb{E}_{\{|\psi_l\rangle\}_l} \left[ (Z_1^{k_n})^{\otimes 10000} |\psi_n\rangle\langle\psi_n|^{\otimes 10000} (Z_1^{k_n})^{\otimes 10000} \otimes \right. \\ & \quad \left. \dots \otimes (Z_1^{k_{2n}})^{\otimes 10000} |\psi_{2n}\rangle\langle\psi_{2n}|^{\otimes 10000} (Z_1^{k_{2n}})^{\otimes 10000} \right]. \end{aligned}$$

Therefore, the two distributions are identical.

$$\mathcal{A}(\text{Samp}^{\{|\psi_l\rangle\}_l}) = \mathbb{E}_{\{k_l\} \leftarrow \{0,1\}^*} \left[ \mathcal{A}(\text{Samp}^{\{X_1^{k_l} |\psi_l\rangle\}_l}) \right].$$



Then the probability that the adversary  $\mathcal{A}$  passes  $l$ 'th test of the verification is upper bounded by the probability that  $\mathcal{A}$  outputs  $k_l$ , plus the probability that the verifier accepts, given that the  $l$ 'th key output by the adversary is not  $k_l$ . The first probability is at most  $1/2$ , and from [Lemma 4.7](#) with  $\delta = 1/10$ , the second is at most  $1/10$ . Since each  $k_l$  is chosen uniformly at random and the distribution over the keys output by  $\mathcal{A}$  does not depend on the choice of  $k_l$ , the probability that the adversary succeeds is at most  $(3/5)^n = \text{negl}(n)$ .  $\square$

The final step is to apply the LOCC Haar indistinguishability result of [\[2\]](#), stated formally below.

**Lemma 4.10** (LOCC Haar indistinguishability, adapted from [\[2\]](#)). *For positive integers  $s, t, n_1, \dots, n_s$ , define*

$$\begin{aligned}\rho_{AB} &:= \bigotimes_{i=1}^s \mathbb{E}_{|\psi_i\rangle \leftarrow \text{Haar}(2^{n_i})} \left[ (|\psi_i\rangle\langle\psi_i|^{\otimes t})_{A_i} \otimes (|\psi_i\rangle\langle\psi_i|^{\otimes t})_{B_i} \right] \\ \sigma_{AB} &:= \bigotimes_{i=1}^s \mathbb{E}_{|\psi_i\rangle \leftarrow \text{Haar}(2^{n_i})} \left[ (|\psi_i\rangle\langle\psi_i|^{\otimes t})_{A_i} \right] \otimes \bigotimes_{i=1}^s \mathbb{E}_{|\phi_i\rangle \leftarrow \text{Haar}(2^{n_i})} \left[ (|\phi_i\rangle\langle\phi_i|^{\otimes t})_{B_i} \right],\end{aligned}$$

where  $A = (A_1, \dots, A_s)$ ,  $B = (B_1, \dots, B_s)$ . Then  $\rho_{AB}$  and  $\sigma_{AB}$  are  $O(\sum_{i=1}^s t^2/2^{n_i})$ -LOCC indistinguishable.

At a high level, [Lemma 4.10](#) says that any adversary for the one-way puzzle can not distinguish between the case when they have been given copies of  $|\psi_l\rangle$ , the states that **Samp** used, or  $|\psi'_l\rangle$ , independently sampled Haar random states. However, an adversary  $\mathcal{A}$  that receives copies of independently sampled  $|\psi'_l\rangle$  can be simulated by an (exponential time) adversary without oracle access to any states. Thus, the probability that an adversary in the CHRS model breaks the one-way puzzle construction is negligible as well. Formally, we have the following lemma.

**Theorem 4.11** (One-way puzzle security). *For all adversaries  $\mathcal{A}^{\{|\psi_l\rangle\}}$  with access to polynomially many copies of the CHRS states, if the sampler samples a puzzle  $(k = (k_1, \dots, k_t), s)$  and inputs the puzzle  $s$  to  $\mathcal{A}^{\{|\psi_l\rangle\}}$ , then  $\mathcal{A}$  passes verification with negligible probability.*

*Proof.* Consider an adversary  $\mathcal{A}^{\{|\psi_l\rangle\}_t}$  with samples of the common Haar random state, and  $\mathcal{A}^{\{|\psi'_l\rangle\}_t}$  that gets samples of an independently sampled Haar random state. We first claim that  $\mathcal{A}^{\{|\psi'_l\rangle\}_t}$  passes verification with probability at most  $(3/5)^n$ . Consider the following adversary  $\mathcal{B}$  that makes no calls to the common Haar random state, but runs in exponential time.  $\mathcal{B}$  maintains a database of classical description of states  $\{(l, |\phi_l\rangle)\}$ . Whenever  $\mathcal{A}$  requests copies of one of the states  $|\psi_l\rangle$ , if  $l$  appears in the database of  $\mathcal{B}$ , then  $\mathcal{B}$  generates a copy of  $|\phi_l\rangle$ , otherwise  $\mathcal{B}$  samples  $|\phi_l\rangle$  independently at random from the Haar measure on  $l$  qubits, adds  $(l, |\phi_l\rangle)$  to the database, and outputs a copy of  $|\phi_l\rangle$ . Because  $|\phi_l\rangle$  are sampled independently at random from  $\{|\psi_l\rangle\}_t$ , the common Haar random state,  $\mathcal{B}$  is accepted with the same probability as  $\mathcal{A}^{\{|\psi'_l\rangle\}_t}$ . However  $\mathcal{B}$  makes no queries to the common Haar random state and by [Lemma 4.9](#), passes one-way puzzle verification with probability at most  $(3/5)^n$ . Therefore  $\mathcal{A}^{\{|\psi_l\rangle\}_t}$  passes one-way puzzle verification with probability at most  $(3/5)^n$ .

Now assume for the sake of contradiction that  $\mathcal{A}^{\{|\psi_l\rangle\}_t}$  passes verification with probability greater than  $(3/5)^n + O(t^2/2^n)$ . Then consider the LOCC procedure where Alice first calls **Samp** and sends the puzzle to Bob. Bob then runs  $\mathcal{A}$  on  $s$  and the copies of their state, and returns the output. Alice then runs **Ver** and outputs 1 if the verifier accepts. If  $\mathcal{A}^{\{|\psi_l\rangle\}_t}$  is accepted with probability greater than  $(3/5)^n + O(t^2/2^n) = \text{negl}(n)$ , then the LOCC protocol described above distinguishes between the case when Alice and Bob have the same state or different state with advantage greater than  $O(t^2/2^n)$ , contradicting [Lemma 4.10](#).  $\square$

Putting [Theorem 4.6](#) together with [Corollary 4.5](#), we have the following.

**Corollary 4.12.** *In the common Haar random state model, inefficiently verifiable one-way puzzles exist but efficiently verifiable one-way puzzles do not exist.*



## 5 Separation in the Haar random swap oracle model

**Note:** We highlight again that there is a bug in this section, pointed out to us by Eli Goldin and Mark Zhandry. While [Lemma 5.5](#) is true, it only holds for fixed states  $\rho$ , and thus can not be applied to get an oracle separation between EFI pairs and OWSG in the Haar random swap model. The construction of OWPuzz and EFI can be lifted to the unitary model but the attack on OWSG cannot be lifted with the approach in this paper. Treating the swap oracle as a reflection around  $|0\rangle - |\psi\rangle$  for a Haar random  $|\psi\rangle$  allows one recover the result, see [\[16\]](#) for a more detailed discussion.

In the previous section, we showed that in CHRS model, one-way state generators do not exist. In this section, we show how to lift certain constructions in the CHRS to a unitary oracle that swaps between  $|0\rangle$  and a Haar random state. Given a state  $|\psi\rangle$ , define the *swap oracle* to be the following:

**Definition 5.1** (Swap oracle). *Let  $|\psi\rangle$  be an  $n$ -qubit state, then the swap oracle  $\mathcal{O}_\psi$  is defined as follows:*

$$\mathcal{O}_\psi = |0^n\rangle\langle\psi| + |\psi\rangle\langle 0^n| + \text{id}_\perp.$$

where we assume WLOG that  $|\psi\rangle$  is orthogonal to  $|0^n\rangle$ , since if not, we can always append a single  $|1\rangle$  to it in order to make it orthogonal. Here  $\text{id}_\perp$  is the identity on the subspace orthogonal to  $\text{span}\{|0^n\rangle, |\psi\rangle\}$ . For a family of states  $\Psi$ , the swap oracle  $\mathcal{O}_\Psi$  is a family of oracles that each swap around the corresponding state in the state family.

We can instantiate the swap model with any common reference quantum state model, but for this paper we will define the Haar random swap oracle as follows.

**Definition 5.2** (Haar random swap oracle). *Let  $\{|\phi_m\rangle\}_{m \in \mathbb{N}}$  be a collection of states sampled from the Haar measure on  $2^m - 1$  dimensions, where  $|\phi_m\rangle$  is a state on  $m$  qubits, conditioned in being orthogonal to  $|0\rangle$ .<sup>10</sup> The Haar random swap oracle is the collection of unitaries:*

$$\mathcal{O}_{|\text{Haar}\rangle} = \mathcal{O}_{\{|\phi_m\rangle\}}.$$

Next, we show that one can use copies of the Haar random state to simulate calls to the Haar random swap oracle, at an inverse polynomial trace distance error. We note that a very similar swap oracle over *two* Haar random state was considered in the work of [\[34\]](#), where he proved a very similar state simulation technique. We adapt his proof strategy for our case.

---

<sup>10</sup>We note that we add the condition that  $|\psi_m\rangle$  is orthogonal to  $|0\rangle$  so that the swap oracle is a unitary. However, for all polynomials,  $\text{poly}(m)$ -copies of a Haar random state on  $2^m$ -dimensions has negligible trace distance to  $\text{poly}(m)$ -copies of a Haar random state on  $2^m - 1$  dimensions, so making this change is indistinguishable to any adversary that makes  $\text{poly}(m)$  calls to the swap oracle, which is what we consider below. Thus, it will often be convenient—and it will add only negligible error to our analysis—to consider the state prepared by the oracle as if it were Haar random on the full  $2^m$ -dimensional space.

*Algorithm 1.* Algorithm for simulating swap oracle  $\mathcal{O}_\psi$ .

**Input:** Unknown quantum state  $\rho$  in register R and  $2q(\lambda) + 1$  copies of  $|\psi\rangle$ , orthogonal to  $|0^n\rangle$ .

1. Coherently measure R using the POVM  $\{|0^n\rangle\langle 0^n|, \text{id} - |0^n\rangle\langle 0^n|\}$ , saving the result in register  $A_1$ .
  - (a) If the measurement result in  $A_1$  has outcome  $|0^n\rangle\langle 0^n|$ , swap out register R with a fresh copy of  $|\psi\rangle$ .
  - (b) If the measurement result in  $A_1$  has outcome  $\text{id} - |0^n\rangle\langle 0^n|$ , perform a symmetric subspace projector  $\Pi_{\text{sym}}^{q(\lambda)+1}$  on register R and  $q(\lambda)$  other registers containing fresh copies of  $|\psi\rangle$ , saving the result in register  $A_2$ .
    - i. If the measurement result in  $A_2$  has outcome  $\Pi_{\text{sym}}$ , swap out register R with an ancilla register containing the state  $|0^n\rangle$ .
2. Coherently measure register R using the POVM  $\{|0^n\rangle\langle 0^n|, \text{id} - |0^n\rangle\langle 0^n|\}$ , writing the result to register  $A_2$ .
3. Coherently measure register R and  $|\psi\rangle^{\otimes q(\lambda)}$  using a symmetric subspace projector  $\Pi_{\text{sym}}^{q(\lambda)+1}$ , writing the result to register  $A_1$ .
4. Return the R register.

Before proving that the algorithm works, we will need to use the following facts, one about the post-measurement state of the symmetric subspace projector and another about Haar random states.

**Lemma 5.3.** *Let  $|\phi\rangle$  be a state perpendicular to  $|\psi\rangle$ , then the post-measurement state after applying the symmetric subspace projector (i.e. the sum over all permutations of  $l+1$  registers) and accepting on  $|\phi\rangle \otimes |\psi\rangle^{\otimes l}$  is*

$$\frac{1}{l+1} \sum_{i=0}^l |\psi\rangle^{\otimes i} |\phi\rangle |\psi\rangle^{\otimes l-i}$$

Similarly, the post-measurement state after applying the symmetric subspace projector and rejecting is given by

$$\frac{l}{l+1} |\phi\rangle |\psi\rangle^{\otimes l} - \frac{1}{l+1} \sum_{i=1}^l |\phi\rangle^{\otimes i} |\phi\rangle |\psi\rangle^{\otimes l-i}.$$

*Proof.* From [18], the symmetric subspace projector is given by

$$\Pi_{l+1}^{\text{sym}} = \frac{1}{(l+1)!} \sum_{\pi \in S_{l+1}} P_\pi.$$

Where  $P_\pi$  acts on  $n$  registers by permuting the registers. Note that we can split  $S_{l+1}$  into cosets of  $S_l$ , where the representative of the  $i$ 'th coset is the swap  $(1, i)$ . Then we have the following

$$\begin{aligned} \frac{1}{(l+1)!} \sum_{\pi \in S_{l+1}} |\phi\rangle \otimes |\psi\rangle^{\otimes l} &= \frac{1}{(l+1)!} \sum_{i \in [l+1]} P_{(1,i)} \sum_{\pi' \in S_l} \text{id} \otimes P_{\pi'} |\phi\rangle \otimes |\psi\rangle^{\otimes l} \\ &= \frac{1}{l+1} \sum_{i \in [l+1]} P_{(1,i)} |\phi\rangle \otimes |\psi\rangle^{\otimes l} \\ &= \frac{1}{l+1} \sum_{i=0}^l |\psi\rangle^{\otimes i} |\phi\rangle |\psi\rangle^{\otimes l-i}, \end{aligned}$$

as desired. To compute the state after the anti-symmetric subspace projector is applied, we simply take  $\text{id} - \Pi_{l+1}^{\text{sym}}$ , i.e. subtract the above state from the original state. Taking the difference yields the desired state.  $\square$

We adapt the following lemma from [34].

**Lemma 5.4** (Paraphrased from [34]). *Let  $|\psi\rangle$  be an  $n$ -qubit state drawn from a phase invariant distribution<sup>11</sup> and  $A^{\mathcal{O}_\psi}$  be a quantum oracle algorithm that makes  $p(\lambda) = \text{poly}(\lambda)$  many queries to the swap oracle  $\mathcal{O}_\psi$ . Let  $\mathcal{O}_\psi^{\text{res}}$  be a simulation that maintains a reservoir register  $\text{res}$  containing up to  $p(\lambda)$  copies of  $|\psi\rangle$ , and performs the following unitary for each query:*

$$\mathcal{O}_\psi^{\text{res}} = \sum_{k \in [p(\lambda)]} |0^n\rangle_{\text{A}} |\psi^k\rangle_{\text{res}} \langle \psi|_{\text{A}} \langle \psi^{k-1}|_{\text{res}} + |\psi\rangle_{\text{A}} |\psi^{k-1}\rangle_{\text{res}} \langle 0^n|_{\text{A}} \langle \psi^k|_{\text{res}} + \text{id}_{\perp}.$$

where  $\text{A}$  is the query register of the algorithm and  $\text{res}$  is the reservoir state, with  $|\psi^k\rangle$  representing the state of the reservoir containing  $k$  copies of  $|\psi\rangle$ .

Then we have that for all input states  $\rho$ ,

$$\mathbb{E}_{\psi \leftarrow \text{Haar}} [A^{\mathcal{O}_\psi}(\rho)] = \mathbb{E}_{\psi \leftarrow \text{Haar}} [A^{\mathcal{O}_\psi^{\text{res}}}(\rho)].$$

*Proof.* This follows directly by an application of the proofs of Lemmas 5.5 and 5.9 of [34] to  $\mathcal{O}_\psi$ .<sup>12</sup>  $\square$

We note that  $\mathcal{O}_\psi^{\text{res}}$  uses a perfect projective measurement onto the state  $|\psi\rangle$ . It remains to show how to implement this using copies of the state  $|\psi\rangle$ . Like [34], we use a technique from [21], implementing a projection on the symmetric subspace of polynomially many copies of  $|\psi\rangle$ . We show how to do this for completeness. We combine them in the following lemma where we extend Lemma 5.4 to an algorithm that takes as input copies of the state  $|\psi\rangle$ , sampled from the Haar measure.

**Lemma 5.5** (Swap oracle from sample access). *Let  $|\psi\rangle$  be an  $n$ -qubit Haar random quantum state and  $A^{\mathcal{O}_\psi}$  be a polynomial-space quantum oracle algorithm that makes  $p(\lambda) = \text{poly}(\lambda)$  many queries to the swap oracle  $\mathcal{O}_\psi$ . Then for every  $\epsilon > 0$ , there exists a polynomial-space quantum algorithm  $B(|\psi\rangle^{\otimes p(\lambda)(\frac{12}{\epsilon})}, \cdot)$  that such for all  $\rho$ ,*

$$\text{td} \left( \mathbb{E}_{\psi \leftarrow \text{Haar}(2^n)} \left[ B(|\psi\rangle^{\otimes p(\lambda)(\frac{12}{\epsilon})}, \rho) \right], \mathbb{E}_{\psi \leftarrow \text{Haar}(2^n)} [A^{\mathcal{O}_\psi}(\rho)] \right) \leq \epsilon.$$

*Proof.*  $B$  proceeds by simulating  $A$ , where for each of the oracle queries made by  $A$ , it uses  $(2t(\lambda) + 1)$  many copies of  $|\psi\rangle$  to run Algorithm 1. Note that since  $\epsilon$  is a constant and  $p$  is a polynomial,  $B$  runs in polynomial space. In order to analyze the error bound with the ideal algorithm, we step through the algorithm for a pure state  $|\phi\rangle\langle\phi|$ , and the result will extend by linearity.<sup>13</sup> First note that we can always find phases such that

$$|\phi\rangle = \alpha_0 |0^n\rangle + \alpha_\psi |\psi\rangle + \alpha_\perp |\phi_\perp\rangle.$$

Then after performing the first measurement, we have the following state mixed state:

$$|0^n\rangle\langle 0^n| |\phi\rangle \otimes |1\rangle_{\text{A}_1} + (\text{id} - |0^n\rangle\langle 0^n|) |\phi\rangle \otimes |0\rangle_{\text{A}_1} = \alpha_0 |0^n\rangle \otimes |1\rangle_{\text{A}_1} + (\alpha_\psi |\psi\rangle + \alpha_\perp |\phi_\perp\rangle) \otimes |0\rangle_{\text{A}_1}.$$

Conditioned on  $\text{A}_1$  being 1, we swap in one of our copies of  $|\psi\rangle$ , and in the 0 branch we perform a multi-SWAP test with  $q(\lambda)$  copies of the state  $|\psi\rangle\langle\psi|$  to project on the the symmetric subspace and its

<sup>11</sup>A distribution on quantum states over a subspace is *phase invariant* if it is invariant to applying a uniformly random phase to each basis state of the subspace [34]. In particular, note that the Haar measure is phase invariant.

<sup>12</sup>The lemmas in [34] are stated for a slightly different oracle. The oracle of [34] has an index register and uses a reservoir of many copies of each state from an indexed collection of different Haar random states. Furthermore, the oracle—at least when queried on index 0—swaps in two different Haar states. Formally the lemmas do not apply to our case. However, the proofs of the lemmas do not make use of these differences with our setting and therefore apply directly to our oracle as a special case.

<sup>13</sup>For the sake of brevity, we drop the expectation over the Haar measure and use kets, but all equations should be taken as being averaged over the Haar measure in  $|\psi\rangle$ .

complement. To save space, we analyze each branch of the superposition separately, and we will re-combine them later. From [Lemma 5.4](#), the state in the branch where  $A_1$  is 1 is indistinguishable (over the Haar measure) to the following, after tracing out extra copies of  $|\psi\rangle$ .

$$\alpha_0 |\psi\rangle |10\rangle_{A_1 A_2}.$$

For the term with coefficient  $\alpha_\psi$ , the SWAP test will pass with probability 1 and leave the following state

$$\alpha_\psi |\psi\rangle^{t(\lambda)+1} |01\rangle_{A_1 A_2}.$$

For the term  $\alpha_\perp |\phi_\perp\rangle$ , we apply our observation about the post-measurement state of the symmetric subspace projector to get

$$\begin{aligned} & \frac{\alpha_\perp}{t(\lambda)+1} \sum_{i=0}^{t(\lambda)} |\psi\rangle^{\otimes i} |\phi_\perp\rangle |\psi\rangle^{t(\lambda)-i} |01\rangle_{A_1 A_2} \\ & + \frac{\alpha_\perp}{t(\lambda)+1} \left( t(\lambda) |\phi_\perp\rangle |\psi\rangle^{\otimes t(\lambda)} - \sum_{i=1}^{t(\lambda)} |\psi\rangle^{\otimes i} |\phi_\perp\rangle |\psi\rangle^{t(\lambda)-i} \right) |00\rangle_{A_1 A_2}. \end{aligned}$$

In the next step, for all of the branches that have  $|01\rangle_{A_1 A_2}$ , we swap out the first register with a fresh ancilla containing  $|0^n\rangle$ . Again, from [Lemma 5.4](#), the state is indistinguishable over the Haar measure from

$$\begin{aligned} & \alpha_0 |\psi\rangle |\psi\rangle^{\otimes t(\lambda)} |10\rangle_A + \alpha_\psi |0^n\rangle |\psi\rangle^{\otimes t(\lambda)} |01\rangle_A \\ & + \frac{\alpha_\perp}{t(\lambda)+1} \left( |0^n\rangle |\psi\rangle^{\otimes t(\lambda)} + \sum_{i=1}^{t(\lambda)} |0\rangle |\psi\rangle^{i-1} |\phi_\perp\rangle |\psi\rangle^{t(\lambda)-i} \right) |01\rangle_A \\ & + \frac{\alpha_\perp}{t(\lambda)+1} \left( t(\lambda) |\phi_\perp\rangle |\psi\rangle^{\otimes t(\lambda)} - \sum_{i=1}^{t(\lambda)} |\psi\rangle^{\otimes i} |\phi_\perp\rangle |\psi\rangle^{t(\lambda)-i} \right) |00\rangle_A. \end{aligned}$$

Re-arranging terms, we have the following state on all the registers after performing the first half of [algorithm 1](#).

$$\begin{aligned} & \alpha_0 |\psi\rangle |\psi\rangle^{\otimes t(\lambda)} |10\rangle_A + \alpha_\psi |0^n\rangle |\psi\rangle^{\otimes t(\lambda)} |01\rangle_A + \frac{\alpha_\perp t(\lambda)}{t(\lambda)+1} |\phi_\perp\rangle |\psi\rangle^{\otimes t(\lambda)} |00\rangle_A \\ & + \frac{\alpha_\perp}{t(\lambda)+1} \left( |0^n\rangle |\psi\rangle^{\otimes t(\lambda)} + \sum_{i=1}^{t(\lambda)} |0^n\rangle |\psi\rangle^{i-1} |\phi_\perp\rangle |\psi\rangle^{t(\lambda)-i} \right) |01\rangle_A \\ & - \frac{\alpha_\perp}{t(\lambda)+1} \left( \sum_{i=1}^{t(\lambda)} |\psi\rangle^{\otimes i} |\phi_\perp\rangle |\psi\rangle^{t(\lambda)-i} \right) |00\rangle_A. \end{aligned}$$

Note that the trace of the absolute value final line is equal to  $|\alpha_\perp|^2 (2t(\lambda)+1)/(t(\lambda)+1)^2$ . Thus, we can instead consider the following state, remembering that we have incurred a trace distance cost of  $|\alpha_\perp|^2 (2t(\lambda)+1)/(t(\lambda)+1)^2 \leq 2|\alpha_\perp|^2/(t(\lambda)+1)$ .

$$\alpha_0 |\psi\rangle |\psi\rangle^{\otimes t(\lambda)} |10\rangle_A + \alpha_\psi |0\rangle |\psi\rangle^{\otimes t(\lambda)} |01\rangle_A + \frac{\alpha_\perp t(\lambda)}{t(\lambda)+1} |\phi_\perp\rangle |\psi\rangle^{\otimes t(\lambda)} |00\rangle_A.$$

For this state we can cleanly trace out the  $t(\lambda)$  registers which contain  $|\psi\rangle^{\otimes t(\lambda)}$ , which means the state on the rest was close in trace distance to

$$|\phi_{\text{half}}\rangle = \alpha_0 |\psi\rangle |10\rangle_A + \alpha_\psi |0\rangle |01\rangle_A + \frac{\alpha_\perp t(\lambda)}{t(\lambda)+1} |\phi_\perp\rangle |00\rangle_A.$$

The final half of the algorithm is the same as the first, except with the  $A_1$  and  $A_2$  registers swapped, so the state after the algorithm is done will be close to the following state.

$$|\phi_{\text{final}}\rangle = \alpha_0 |\psi\rangle_A + \alpha_\psi |0\rangle + \frac{\alpha_\perp t(\lambda)^2}{(t(\lambda) + 1)^2} |\phi_\perp\rangle.$$

By the same argument as before, the trace distance between the second half of the algorithm acting on  $|\phi_{\text{half}}\rangle$  and  $|\phi_{\text{final}}\rangle$  is at most

$$\frac{2|\alpha_\perp t(\lambda)/(t(\lambda) + 1)|^2 t(\lambda)}{(t(\lambda) + 1)^2} \leq \frac{2|\alpha_\perp|^2}{t(\lambda) + 1}.$$

Here we used the fact that the new amplitude of  $|\phi_\perp\rangle$  is  $\alpha_\perp t(\lambda)/(t(\lambda) + 1)$  and plugged it into the trace distance calculation from before. By the triangle inequality, the trace distance between the actual state of the algorithm and  $|\phi_{\text{final}}\rangle$  is at most

$$\text{td}(|\phi_{\text{final}}\rangle\langle\phi_{\text{final}}|, \rho_{\text{alg}}) \leq \frac{2|\alpha_\perp|^2}{t(\lambda) + 1} + \frac{2|\alpha_\perp|^2}{t(\lambda) + 1} \leq \frac{4|\alpha_\perp|^2}{t(\lambda) + 1} \leq \frac{4}{t(\lambda) + 1}.$$

Finally, we can write down the state after the ideal swap as

$$|\phi_{\text{ideal}}\rangle = \alpha_0 |\psi\rangle + \alpha_\psi |0\rangle + \alpha_\perp |\phi_\perp\rangle.$$

We can compute directly the trace distance between the two states to get the following bound

$$\text{td}(|\phi_{\text{final}}\rangle\langle\phi_{\text{final}}|, |\phi_{\text{ideal}}\rangle\langle\phi_{\text{ideal}}|) \leq \frac{2t(\lambda) + 1}{(t + 1)^2} \leq \frac{2}{t(\lambda) + 1}.$$

Applying the triangle inequality for trace distance, we get that the trace distance between the state of the algorithm on the first register and the ideal state is upper bounded by

$$\text{td}(\rho_{\text{alg}}, |\phi_{\text{ideal}}\rangle\langle\phi_{\text{ideal}}|) \leq \frac{6}{t(\lambda) + 1}.$$

Setting  $t(\lambda) = \frac{6p(\lambda)}{\epsilon} - 1$ , we get a trace distance error of at most  $\epsilon/p(\lambda)$ . For an algorithm that makes  $p(\lambda)$  calls to the oracle, we apply the triangle inequality to every call to get a total error bound of  $\epsilon$  over the course of the entire algorithm  $A^{\mathcal{O}_\psi}$ . This means that the algorithm requires  $\frac{12p(\lambda)}{\epsilon}$  copies of the state  $|\psi\rangle$ , as desired. Finally, we note that since this bound holds for all pure states, we can write all mixed states as a mixture of pure states and apply this to each of them, so this applies to all entangled inputs too.  $\square$

*Remark 5.6.* [Lemmas 5.4](#) and [5.5](#) allow us to simulate the effect of querying the Haar random swap oracle using only polynomially many copies of the Haar random state itself, up to inverse polynomial error. Note that in order to apply it directly to an interactive computation (for instance, an interaction that occurs as part of a cryptographic security game), we must have a single global simulator that simulates the oracle for all the parties to the computation. For instance, consider a scenario in which two parties prepare a state close to  $|\psi\rangle + |1\rangle$  by querying the Haar random swap oracle on the state  $|0\rangle + |1\rangle$  and then proceed to query a second time expecting  $|0\rangle + |1\rangle$  on the output. If there were only a single global simulator, the second query would approximately undo the first, and the state is guaranteed by [Lemmas 5.4](#) and [5.5](#) to return to  $|0\rangle + |1\rangle$  up to inverse polynomial error. If, on the other hand, the two queries are made to two *different* simulators (say a different simulator for each party), the second query does not undo the first. This is because this process returns a copy of  $|\psi\rangle$  that was pulled from the reservoir of the first simulator to the reservoir of the second simulator, and thus produces on the output a mixed state that is entangled with both simulators and potentially quite far from  $|0\rangle + |1\rangle$ .<sup>14</sup> This is, however, not an issue for our case. As we will see below, in order to upgrade the existence of the primitives we are interested in to the Haar random swap oracle setting, we only need to consider a simulator for the adversary, and not the challenger of the security game. Similarly, in order to rule out efficiently verifiable OWPuzz and OWSG in the Haar random swap oracle setting, we only need to simulate the verification algorithm.

<sup>14</sup>We thank Mark Zhandry and Eli Goldin for pointing out this scenario.

With these lemmas, we can show that since EFI pairs exist in the common Haar random state model, they also exist in the corresponding swap model. We apply the result to the common Haar random state model to get the following corollaries.

**Corollary 5.7.** *EFI pairs exist relative to  $(\mathcal{O}_{|\text{Haar}\rangle}, \text{UnitaryPSPACE})$ .*

*Proof.* The construction of EFI pairs runs the construction in the common Haar random state model, using the swap oracle to generate copies of the desired states. To show security of the model, we show that any adversary in the Haar random swap model would imply an adversary against the EFI pair in the common Haar random state model.

Assume for the sake of contradiction that there is an adversary  $A$  that breaks the security of the EFI pair relative to  $(\mathcal{O}_{|\text{Haar}\rangle}, \text{UnitaryPSPACE})$ . Consider an algorithm  $\text{Sim}$  that, given bit  $b$ , first runs the construction of the EFI pair using the swap oracle with bit  $b$ , and then runs  $A$  on the output. By assumption, the adversary  $A$  breaks the security of the EFI pair, so there exists a polynomial  $q$  such that

$$|\Pr[\text{Sim}(0) = \top] - \Pr[\text{Sim}(1) = \top]| \geq \frac{1}{q(\lambda)}.$$

Note that this algorithm starts from the all 0's state, so we can apply Lemma 5.5 to simulate it. When we apply Lemma 5.5 with  $\epsilon = 1/(2q(\lambda))$ , we will get a simulator  $\widetilde{\text{Sim}}$  in the common Haar random state model. We assume without loss of generality that the construction of EFI pairs begins by generating all of the copies of the Haar random states by calling the swap oracle on  $|0\rangle$ , so we see that Algorithm 1 will simulate the EFI pair perfectly (as in, it will swap the initial state  $|0\rangle$  with copies of  $|\psi\rangle$  and then terminate, and the construction never calls the swap oracle again). Thus, the probability that  $\widetilde{\text{Sim}}(b)$  accepts is exactly the probability that an adversary  $\widetilde{A}$  (the simulation of  $A$ ) accepts when given  $\rho_b$ , the corresponding state in the EFI pair.

Thus, the simulated adversary breaks the indistinguishability of the EFI pair with advantage  $1/(2q(\lambda))$ , which contradicts the security of the EFI pair in the common Haar random model. Thus, we conclude that the construction of EFI pairs in the Haar random swap model is secure.  $\square$

We can apply the same idea to 1PRS and OWPuzz (and more generally, any primitive with a non-interactive security game) and conclude that they exist in the common Haar random swap model. We claim that one-way state generators *do not* exist relative to the Haar random swap oracle model.

**As we pointed out at the start of the section, the proof of the claim is wrong. See [16] for more details.**

**Claim 5.8.** *Relative to  $(\mathcal{O}_{|\text{Haar}\rangle}, \text{UnitaryPSPACE})$ , efficiently verifiable one-way state generators do not exist.*

*Proof.* We consider a similar quantum threshold attack as in Theorem 4.1. Let  $(\text{KeyGen}, \text{StateGen}, \text{Ver})$  be a candidate one-way state generator. Applying Lemma 5.5 with  $\epsilon = 1/5$ , we can construct a simulated verifier  $\widetilde{\text{Ver}}$  which takes  $\text{poly}(\lambda)$  copies of the Haar random state from the swap oracle, and satisfies the following properties:

- On average over the Haar random states, when  $\widetilde{\text{Ver}}$  is given  $k$  and a copy of  $\rho_k$  (where  $\rho_k$  is the real one-way state generator state),  $\widetilde{\text{Ver}}$  accepts with probability at least  $3/4$ .
- If  $\widetilde{\text{Ver}}$  accepts with probability  $p$ , then  $\text{Ver}$  accepts with probability at least  $p - 1/5$ .

Let  $U_k$  be the unitary that implements  $\widetilde{\text{Ver}}$  on key  $k$ , which takes in a copy of a one-way state generator state in register A, and  $\text{poly}(\lambda)$  copies of the Haar random state from the swap oracle in register B, and let  $|\geq m\rangle\langle\geq m|$  be the projector onto bit strings with Hamming weight greater than  $m$ , and consider the following measurement.

$$\Pi_k = \left( \left( U_k^\dagger \right)_{\text{CD}}^{\otimes 10\lambda} (|\geq 5\lambda\rangle\langle\geq 5\lambda|_{\text{C}} \otimes \text{id}_{\text{D}}) (U_k)_{\text{AB}}^{\otimes 10\lambda} \right).$$

Here C is the output register of  $U_k$ , and D is the purifying register for the output.

**Claim 5.9.** *On input  $(\rho_k \otimes |\phi_1\rangle\langle\phi_1| \otimes \dots \otimes |\phi_1\rangle\langle\phi_1|)^{\otimes 10\lambda}$ ,  $\Pi_k$  accepts with probability at least*

$$1 - \exp(-\lambda/8) .$$

*Proof.* From the first observation, every run of  $U_k$  has probability at least  $3/4$  of accepting, and since the input state is in tensor product on the original state, every run is independent of each other. Applying Hoeffdings inequality, we get the following:

$$\Pr[\Pi_k \text{ rejects}] \leq \exp\left(-2\frac{(\lambda/4)^2}{n}\right) = \exp(-\lambda/8) .$$

This completes the proof of [Claim 5.9](#). □

We also prove the other direction in a similar way to [Claim 4.3](#).

**Claim 5.10.** *If  $\Pi_k$  accepts with probability  $\geq 1/3$  for some key  $k$ , then the probability that the original verification accepts is at least  $3/4$ .*

*Proof.* From the same proof as [Claim 4.3](#), the probability that the simulated verifier accepts is at least  $1 - \frac{1}{5\lambda}$ . By [Lemma 5.5](#), the probability that the original verifier would accept is at least  $1 - \frac{1}{5\lambda} - \frac{1}{5} \geq \frac{3}{4}$  for suitably large  $\lambda$ . □

The algorithm for breaking the one-way state generator is to first query the Haar random swap oracle  $\text{poly}(\lambda)$  many times to produce that many copies of the corresponding Haar random state, and then use the **UnitaryPSPACE** oracle to run threshold search on those  $\text{poly}(\lambda)$  copies of the Haar random state and  $O(\lambda^2)$  many copies of the OWSG state. Note that, as before, the verification might query the Haar random swap oracle for many different integers  $m$ , so our attack must make  $\text{poly}(\lambda)$  copies of the Haar random state for all  $m$  from 1 to  $\text{poly}(\lambda)$  in order to run the simulated verification. From the previous claims, the procedure is guaranteed to output a key that passes verification with probability  $3/4$ , with constant probability. □

By the same observation as in the common Haar random state case, we note that the exact same attack rules out constructions of efficiently verifiable one-way puzzles in the Haar random swap model.

**Theorem 5.11.** *Relative to  $(\mathcal{O}_{|\text{Haar}\rangle}, \text{UnitaryPSPACE})$ , EFI pairs, 1PRS, and OWPuzz exist, but one-way state generators do not.*

Thus, we have achieved a unitary oracle that separates one-way state generators from EFI pairs, 1PRS, and OWPuzz. Note that the oracle we provide is a randomized unitary oracle—that is, it is a unitary oracle that is chosen from a probability distribution. However, we note that this is not a disadvantage, as if we desire a fixed deterministic unitary oracle, this can easily be achieved as well by applying the techniques of [1, proof of Theorem 1.1] to  $\mathcal{O}_{|\text{Haar}\rangle}$ .

## References

- [1] Scott Aaronson and Greg Kuperberg. Quantum versus classical proofs and advice. In *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC'07)*, pages 115–128. IEEE, 2007.
- [2] Prabhanjan Ananth, Aditya Gulati, and Yao-Ting Lin. Cryptography in the common haar state model: Feasibility results and separations. *arXiv preprint arXiv:2407.07908*, 2024.
- [3] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. In *Annual International Cryptology Conference*, pages 208–236. Springer, 2022.
- [4] Rishabh Batra and Rahul Jain. Commitments are equivalent to one-way state generators, 2024.



- [5] Amit Behera, Zvika Brakerski, Or Sattath, and Omri Shmueli. Pseudorandomness with proof of destruction and applications. *Cryptology ePrint Archive*, Paper 2023/543, 2023.
- [6] Amit Behera, Giulio Malavolta, Tomoyuki Morimae, Tamer Mour, and Takashi Yamakawa. A new world in the depths of microcrypt: Separating owsgs and quantum money from qefid, 2024.
- [7] John Bostanci, Yuval Efron, Tony Metger, Alexander Poremba, Luowen Qian, and Henry Yuen. Unitary complexity and the uhlmann transformation problem, 2023.
- [8] Zvika Brakerski, Ran Canetti, and Luowen Qian. On the Computational Hardness Needed for Quantum Cryptography. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [9] Costin Bădescu and Ryan O’Donnell. Improved quantum data analysis. *TheoretiCS*, Volume 3, March 2024.
- [10] Boyang Chen, Andrea Coladangelo, and Or Sattath. The power of a single haar random state: constructing and separating quantum pseudorandomness, 2024.
- [11] Kai-Min Chung, Eli Goldin, and Matthew Gray. On central primitives for quantum cryptography with classical communication. *Cryptology ePrint Archive*, Paper 2024/356, 2024. <https://eprint.iacr.org/2024/356>.
- [12] Andrea Coladangelo and Saachi Mutreja. On black-box separations of quantum digital signatures from pseudorandom states. *arXiv preprint arXiv:2402.08194*, 2024.
- [13] Paul Dumais, Dominic Mayers, and Louis Salvail. Perfectly concealing quantum bit commitment from any quantum one-way permutation. In *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*, volume 1807 of *Lecture Notes in Computer Science*, pages 300–315. Springer, 2000.
- [14] Uma Girish and Ran Raz. Eliminating intermediate measurements using pseudorandom generators. *arXiv preprint arXiv:2106.11877*, 2021.
- [15] Eli Goldin, Tomoyuki Morimae, Saachi Mutreja, and Takashi Yamakawa. Countcrypt: Quantum cryptography between QCMA and PP, 2024.
- [16] Eli Goldin and Mark Zhandry. Translating between the common haar random state model and the unitary model. *arXiv preprint arXiv:2503.11634*, 2025.
- [17] Sam Gunn, Nathan Ju, Fermi Ma, and Mark Zhandry. Commitments to quantum states. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, page 1579–1588, New York, NY, USA, 2023. Association for Computing Machinery.
- [18] Aram W Harrow. The church of the symmetric subspace. *arXiv preprint arXiv:1308.6595*, 2013.
- [19] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Predicting many properties of a quantum system from very few measurements. *Nature Physics*, 16(10):1050–1057, June 2020.
- [20] Russell Impagliazzo. A personal view of average-case complexity. In *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*, pages 134–147. IEEE, 1995.
- [21] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. *Pseudorandom Quantum States*, page 126–152. Springer International Publishing, 2018.
- [22] Dakshita Khurana and Kabir Tomer. Commitments from quantum one-wayness. *Cryptology ePrint Archive*, Paper 2023/1620, 2023. <https://eprint.iacr.org/2023/1620>.

- [23] Dakshita Khurana and Kabir Tomer. Commitments from quantum one-wayness. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 968–978, 2024.
- [24] William Kretschmer. Quantum pseudorandomness and classical complexity. In *16th Conference on the Theory of Quantum Computation, Communication and Cryptography*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.
- [25] William Kretschmer, Luowen Qian, Makrand Sinha, and Avishay Tal. Quantum cryptography in algorithmica, 2023.
- [26] Alex Lombardi, Fermi Ma, and John Wright. A one-query lower bound for unitary synthesis and breaking quantum cryptography, 2023.
- [27] Elizabeth S. Meckes. *The Random Matrix Theory of the Classical Compact Groups*. Cambridge Tracts in Mathematics. Cambridge University Press, 2019.
- [28] Tomoyuki Morimae, Barak Nehoran, and Takashi Yamakawa. Unconditionally secure commitments with quantum auxiliary inputs. In *Annual International Cryptology Conference*, pages 59–92. Springer, 2024.
- [29] Tomoyuki Morimae and Takashi Yamakawa. One-Wayness in Quantum Cryptography, 10 2022.
- [30] Tomoyuki Morimae and Takashi Yamakawa. *Quantum Commitments and Signatures Without One-Way Functions*, page 269–295. Springer Nature Switzerland, 2022.
- [31] Luowen Qian. Unconditionally secure quantum commitments with preprocessing. In *Annual International Cryptology Conference*, pages 38–58. Springer, 2024.
- [32] Luowen Qian, Justin Raizes, and Mark Zhandry. Hard quantum extrapolations in quantum cryptography, 2024.
- [33] Adam Bene Watts and John Bostanci. Quantum event learning and gentle random measurements, 2024.
- [34] Mark Zhandry. The space-time cost of purifying quantum computations. In Venkatesan Guruswami, editor, *15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*, volume 287 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 102:1–102:22, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.