A Note on Copy-Protection from Random Oracles

Prabhanjan Ananth[*] UCSB

Fatih Kaleoglu[†] UCSB

Abstract

Quantum copy-protection, introduced by Aaronson (CCC'09), uses the no-cloning principle of quantum mechanics to protect software from being illegally distributed. Constructing copy-protection has been an important problem in quantum cryptography.

Since copy-protection is shown to be impossible to achieve in the plain model, we investigate the question of constructing copy-protection for arbitrary classes of unlearnable functions in the random oracle model. We present an impossibility result that rules out a class of copy-protection schemes in the random oracle model assuming the existence of quantum fully homomorphic encryption and quantum hardness of learning with errors. En route, we prove the impossibility of approximately correct copy-protection in the plain model.

1 Introduction

Quantum copy-protection, introduced by Aaronson [Aar09], is a foundational concept in quantum cryptography. It stipulates that the no-cloning principle of quantum mechanics [Die82, WZ82] can be employed to protect against illegal distribution of software. In more detail, an efficient adversary, on input a copy-protected software (represented as a quantum state), cannot create two copies of software, possibly entangled with each other, such that both copies compute the same functionality as the original software.

The primitive of copy-protection can be classified under the broad area of unclonable cryptography, which deals with using the no-cloning principle to design cryptographic primitives with security properties that are classically unachievable. Many interesting primitives in this category, such as quantum money [Wie83, AC12, Zha19b, RS19], one-shot signatures [AGKZ20], single-decryptor encryption [GZ20, CLLZ21], unclonable encryption [Got02, BL20], and encryption with certifiable deletion [BI20], can be seen as copy-protecting specific functionalities.

The focus of our work is on understanding the feasibility of constructing copy-protection for *all* classes of unlearnable functions. Ananth and La Placa [ALP21] show that there are functions that cannot be copy-protected in the plain model. Thus, one has to rely on alternate models to construct copy-protection.

In this work, we restrict our attention to the random oracle model. Interestingly, random oracles have been helpful for achieving copy-protection for specific classes of functions. Coladangelo, Majenz and Poremba [CMP20] showed the existence of copy-protection for multi-bit output functions in the random oracle model. Ananth, Kaleoglu, Liu, Li and Zhandry [AKL$^+$22] presented a

[*] prabhanjan@cs.ucsb.edu

[†] kaleoglu@ucsb.edu

new construction of copy-protection for single-bit output point functions also in the random oracle model. However, the existence of copy-protection for *all* classes of unlearnable functions in the random oracle model is still yet to be explored.

Before we delve into this direction further, we first need to model the type of access the algorithms in the copy-protection scheme and the adversarial entities will have, with the random oracle. There are two types of accesses we can consider. The first one is classical access, where the interface is entirely classical: the algorithms submit a binary string x to the oracle and get back $f(x)$, where f is a classical random function implemented by the oracle. We will refer to this setting as *classical-accessible random oracle model* (CAROM). The second type is quantum access: the algorithms submit a query of the form $\sum_{x\in\{0,1\}^n} \alpha_x|x\rangle|y_x\rangle$ and get back $\sum_{x\in\{0,1\}^n} \alpha_x|x\rangle|f(x)\oplus y_x\rangle$. This setting was defined by Boneh, Dagdelen, Fischlin, Lehmann, Schaffner and Zhandry [BDF$^+$11] as the *quantum random oracle model* (QROM).

As argued in [BDF$^+$11], QROM is the preferred model over CAROM under most circumstances. For instance, if the adversary has access to any classical code that computes the function f, then it can always run this code coherently, thus achieving quantum access. Nonetheless, as per the impossibility result of [ALP21], one cannot simply instantiate a generic copy-protection scheme in QROM by replacing the random function f with a concrete function, e.g. a heuristically secure hash function such as SHA-512 or an obfuscated pseudorandom function. In contrast, classical-access is appropriate for alternative methods of instantiation, such as using trusted hardware or a trusted party, where classical interface can be enforced. Therefore, we argue that achieving copy-protection in CAROM is still meaingful.

1.1 Our Result

We make progress towards understanding the feasibility of copy-protection in the random oracle model. It turns out that proving impossibility of copy-protection in the QROM seems quite challenging and thus, we focus on the CAROM setting. Note that QROM and CAROM are incomparable models, since QROM gives more power to both the honest algorithms and the adversary.

We show that copy-protection of arbitrary unlearnable functions is impossible in CAROM. This implies that if copy-protection is possible with a random oracle, then honest algorithms must query the oracle in superposition.

Theorem 1. *Assuming unbounded fully homomorphic encryption for quantum computations [Mah18, Bra18] and quantum hardness of learning with errors (QLWE), there exists a class of unlearnable functions $\mathcal{F}$ such that quantum copy-protection for $\mathcal{F}$ is impossible in CAROM.*

The above result suggests that if one were to base copy-protection in the classical-accessible oracle models then the oracle needs to have some structure. We note that in both the works of [CMP20] and [AKL$^+$22], the copy-protection algorithm only makes classical queries to the oracle whereas the evaluation algorithm makes quantum queries.

At the heart of the above theorem is a new impossibility result for copy-protection in the plain model. Specifically, we show that even copy-protection with approximate correctness can be ruled out in the plain model (Theorem 17), while prior works [ALP21, ABDS21] only ruled out copy-protection with statistical correctness.

We combine this impossibility result with a generic transformation from a copy-protection scheme in CAROM to an approximate copy-protection scheme in the plain model (Lemma 21) to rule out copy-protection of arbitrary unlearnable functions in CAROM (Theorem 22). Our transformation crucially relies on recording the oracle queries of the honest algorithms and efficient simulation of a random oracle consistent with a prerecorded database. Recording quantum queries to a random oracle can be achieved using Zhandry's compressed oracle technique [Zha19a], but to our knowledge it is not known how to efficiently simulate a quantum random oracle consistent with a prerecorded database. Therefore, any construction in QROM where honest parties make superposition queries to the oracle would circumvent our impossibility result.

1.2 Related Work

Aaronson [Aar09] was the first to study copy-protection in the oracle models. Their construction relied upon oracles implementing quantum functionalities. Recently, Aaronson, Liu, Liu, Zhandry and Zhang [ALL+21] constructed copy-protection in a quantum-accessible oracle model. As mentioned earlier, two works [CMP20, AKL+22] present constructions of copy-protection for point functions in the quantum random oracle model.

In a recent exciting work, Coladangelo, Liu, Liu and Zhandry [CLLZ21] proposed the first construction of copy-protection for a restricted class of functions, namely pseudorandom functions, in the plain model, based on post-quantum indistinguishability obfuscation and post-quantum one-way functions. Another work by Ananth and Kaleoglu [AK21] present a construction of approximately correct copy-protection for a class of point functions from post-quantum one-way functions. Some recent works [ALP21, ALL+21, KNY21, BJL+21] present constructions of weaker notions of copy-protection.

2 Overview of Techniques

In this section, we explain the main ideas behind our result. Inspired by the ideas developed in the program obfuscation literature [BV16, CKP15], we design a two step approach to proving this impossibility result.

- *Ruling out approximate copy-protection in the plain model*: In the first step, we rule out a notion of copy-protection where correctness is only guaranteed for a large constant fraction of inputs.

- *From CP using oracles to approximate CP in the plain model*: In the second step, we show that copy-protection in the classical-accessible random oracle model implies approximate copy-protection in the plain model. In other words, given any copy-protection in the classical-accessible random oracle model, we can generically get rid of the random oracle model at the cost of weakening the correctness guarantee.

Ruling out Approximate Copy-Protection. We show that copy-protection, where correctness is guaranteed for a fraction of inputs, say $(1-\varepsilon)$, is impossible to achieve in the plain model. This strengthens the previous result [ALP21] which ruled out copy-protection with correctness negligibly close to 1.

Our goal is to transform this scheme, call it CP, such that in the transformed scheme, call it CP', on every input, the evaluation of the copy-protected state is correct with probability $(1-\varepsilon)$ (i.e., per-input $(1-\varepsilon)$-correctness); assume for now, that C is a boolean circuit with 1-bit output. The scheme CP' is designed as follows: to copy-protect C, compute a copy-protection of a circuit

G, with respect to the scheme CP, such that G takes as input an encryption of x, homomorphically computes C on encryption of x and outputs the result. The output is copy-protection of G along with the public key-secret key pair of the encryption scheme. During the evaluation process of CP', first encrypt the input x, run the CP copy-protection of G to obtain encryption of $C(x)$ and finally, decrypt the answer to obtain $C(x)$ in the clear.

This idea was notably developed by Bitansky and Vaikuntanathan [BV16] in the context of indistinguishability obfuscation. To make this idea work, we would need fully homomorphic encryption schemes for quantum computations (QFHE). Moreover, we require the QFHE scheme to satisfy circuit privacy; that is, the homomorphically evaluated ciphertext does not leak information about the circuit being used during evaluation. This property is necessary because the evaluator who obtains a classical description of the function can trivially break the copy-protection security. Fortunately, a recent work by Chardouvelis, Döttling, and Malavolta [CDM20] demonstrates the existence of a QFHE scheme satisfying the circuit privacy property we need[1].

Failure of Majority Argument. Suppose we manage to reduce the feasibility of copy-protection with correctness over a $(1-\varepsilon)$-fraction of inputs to per-input $(1-\varepsilon)$-correctness. The next step would be to rule out copy-protection with per-input correctness. A natural attempt would be to consider multiple copies of copy-protection and take a majority vote; this would improve the correctness to be close to 1. While this argument would work for program obfuscation, this unfortunately fails in the context of copy-protection. The reason is simple: once you have many copies of copy-protection, an adversary can now distribute each copy to a different individual, thus breaking the security of copy-protection.

Thus, we need to figure out an alternate method to rule out quantum copy-protection with per-input correctness guarantees. As done in prior works, we rely upon non-black box techniques to rule out copy-protection. We need to be especially careful when invoking non-black box arguments in the per-input $(1-\varepsilon)$-correctness setting, as every evaluation of the copy-protected state significantly degrades the correctness guarantee of the original state. If ε is the correctness error then after k evaluations, the trace distance between the original state and the new state is $k\sqrt{\varepsilon}$ (by quantum union bound). If $k \geq \frac{1}{\sqrt{\varepsilon}}$ then the new state is useless. Luckily, the quantum non-black box technique of [ALP21] involves the attacker only making two evaluations of the copy-protected state.

In the technical sections, we combine the self-reducibility technique (presented above) with the non-black box technique of [ALP21] to rule out copy-protection for $(1-\varepsilon)$-fraction of inputs.

From CP using oracles to approximate CP. The next step would be to rule out copy-protection in the classical-accessible random oracle model. We rely upon "de-oracle-izing" techniques developed in the context of obfuscation [CKP15]; the idea is to remove the use of random oracle in the construction at the cost of weakening the correctness guarantee. While the overall template is inspired from [CKP15], the actual construction is different.

In more detail, given a copy-protection scheme $(\mathsf{CP}, \mathsf{Eval})$ in the classical-accessible random oracle model, where CP represents the copy-protection algorithm and Eval represents the evaluation algorithm, define a copy-protection scheme $(\widetilde{\mathsf{CP}}, \widetilde{\mathsf{Eval}})$ in the plain model as follows:

[1] We note that the definition considered in [CDM20] is weak: for example, they do not even handle adversaries who can entangle the quantum messages with their private state. We provide a stronger definition in this paper and remark that the construction provided in [CDM20] already satisfies this stronger definition.

1. Using a random oracle $\mathcal{O}$ simulated *on-the-fly*, $\widetilde{\mathsf{CP}}(f)$ runs $\mathsf{CP}^{\mathcal{O}}(f)$ to obtain a copy-protected program ρ_f. It then runs a random (polynomial) number of test executions $\mathsf{Eval}^{\mathcal{O}}(\rho_f, x_i)$ for randomly chosen inputs x_i. It records all the queries made by Eval during the test executions in a database D. Finally, it samples a set of random oracle answers R and outputs (ρ, D, R).

2. $\widetilde{\mathsf{Eval}}((\rho_f, D, R), x)$ simulates a random oracle $\mathcal{O}'$ using the database D and using R for queries not recorded in D. It runs $\mathsf{Eval}^{\mathcal{O}'}(\rho_f, x)$ and outputs the answer.

The key difference between our construction and the construction of [CKP15] is that we choose the number of test executions at random. In the classical world, running more test executions can never hurt the simulation. In the quantum world, on the other hand, every test execution can significantly alter the state ρ_f by performing measurements. This could be true even if ρ_f is reusable, in the sense that its correctness guarantee is preserved after polynomially many evaluations[2]. For instance, the state ρ_f could maintain a counter which affects the oracle queries made by Eval. To ensure that the oracle queries $\widetilde{\mathsf{Eval}}$ needs to answer are captured in the database D with probability close to 1, even if ρ_f keeps changing, we choose the number of test queries at random and sufficiently large.

Perhaps surprisingly, our technique also improves the classical impossibility of obfuscation in the random oracle model! Specifically, it reduces the number of test executions needed by a factor of N when compared to [CKP15], where N is a query-bound for Eval.

2.1 Acknowledgements

We thank Luowen Qian for valuable discussions.

2.2 Organization

Preliminaries are described in Section 3. The impossibility of approximate copy-protection in the plain model is presented in Section 4.1. We prove the impossibility of copy-protection in CAROM in Section 4.2.

3 Preliminaries

3.1 Notation

We denote by $x \xleftarrow{\$} X$ the sampling of an element x from the uniform distribution over X. We denote by λ the security parameter. We use the terms *function* and *circuit* interchangeably. We denote by $\mathsf{negl}(\cdot)$ a generic negligible function. $[N] := \{1, 2, \ldots, N\}$. If an algorithm $\mathcal{A}$ has oracle access to $\mathcal{O}$, we may write $\mathcal{A}^{\mathcal{O}}$ to emphasize this fact. We call a function $\epsilon : \mathbb{Z}^+ \rightarrow \mathbb{R}^+ \cup \{0\}$ *noticable* if there exists a polynomial p such that $\epsilon(n) > \frac{1}{p(n)}$ for sufficiently large n. We say *overwhelming* probability to mean probability $1 - \mathsf{negl}(\lambda)$. We sometimes shorten a string of zeros to 0, when the length will be understood from the context. We write $\mathcal{A}(x; r)$ when a classical algorithm $\mathcal{A}$ is run on input x using specified randomness r.

[2] These *forced* measurements are a unique feature of the classical-accessible oracle setting, since quantum queries need not be measured at the time of the query by the deferred measurement principle.

3.2 Quantum Computing Basics

For a finite set (register) X we denote by $\mathcal{H}_X$ the Hilbert Space generated by the basis $\{|x\rangle\}_{x\in X}$. We denote by $\mathcal{D}(\mathcal{H})$ the set of valid mixed quantum states over Hilbert space $\mathcal{H}$, i.e. linear, positive-semidefinite operators over $\mathcal{H}$ with unit trace, also known as density operators. We define a quantum polynomial time (QPT) algorithm as a family of generalized quantum circuits $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$ such that each $\mathcal{A}_\lambda$ contains at most $p(\lambda)$ input/output qubits (including auxiliary qubits) and at most $p(\lambda)$ gates from a universal gate set (such as $\{CNOT, H, T\}$), for some polynomial $p(\cdot)$.

3.2.1 Trace Distance

A common way to measure dissimilarity between two quantum states $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ is the (normalized) trace distance, defined as

$$T(\rho,\sigma) := \frac{1}{2}|\rho-\sigma|_{tr} = \frac{1}{2}\mathsf{Tr}\left(\sqrt{(\rho-\sigma)^2}\right).$$

It is a fact that no quantum algorithm can increase the trace distance between two quantum states, i.e. $T(\mathcal{A}(\rho),\mathcal{A}(\sigma)) \leq T(\rho,\sigma)$ for any $\mathcal{A},\rho,\sigma$. When two quantum states $\rho(\lambda),\sigma(\lambda)$ depend on a security parameter and satisfy $T(\rho,\sigma) \leq \mathsf{negl}(\lambda)$, we write $\rho \approx_s \sigma$, meaning ρ and σ are *statistically close.*

3.2.2 Measurement

A quantum measurement on a Hilbert Space $\mathcal{H}$ with a finite set of outcomes $[m]$ can be represented by a set of operators $(M_i)_{i\in[m]}$ satisfying $\sum_{i\in[m]} M_i^\dagger M_i = I$. Given a quantum state $\rho \in \mathcal{D}(\mathcal{H})$, the probability of outcome i is given by $\mathsf{Tr}(M_i\rho M_i^\dagger)$, whereas the post-measurement state after measuring outcome i is given by $\frac{M_i\rho M_i^\dagger}{\mathsf{Tr}(M_i\rho M_i^\dagger)}$. If the measurement outcome is not revealed, then the post-measurement state equals the mixture $\sum_{i\in[m]} M_i\rho M_i^\dagger$.

3.2.3 Almost As Good As New Lemma

The following lemma from [Aar04] is widely used in literature[3].

Lemma 2 (Almost as Good as New Lemma)**.** *Let ρ be a quantum state and (M_0, M_1) be a 2-outcome measurement such that $\mathsf{Tr}(M_0\rho M_0^\dagger) \geq 1-\varepsilon$. Then, after this measurement is performed on ρ, it is possible to recover a state ρ' such that $T(\rho,\rho') \leq \sqrt{\varepsilon}$. In addition, the recovery procedure is independent of ρ, and is efficient (runs in polynomial time) given that (M_0, M_1) is efficient.*

Corollary 3. *Let $\mathcal{A}$ be a quantum algorithm that given as input a quantum state ρ and a classical string x, outputs a classical string y. Suppose that*

$$\mathsf{Pr}\left[f(x) \leftarrow \mathcal{A}(\rho,x)\right] \geq 1-\varepsilon(x), \tag{1}$$

where ρ is a quantum input state, x is a classical input, and $f(x)$ is a classical deterministic function. Then, there exists a quantum algorithm $\mathcal{A}'$ such that (1) $\mathcal{A}'(\rho,x)$ outputs a classical string y identically distributed to output of $\mathcal{A}(\rho,x)$, and (2) in addition $\mathcal{A}'$ outputs a residual state ρ' satisfying $T(\rho,\rho') \leq \sqrt{\varepsilon(x)}$. Moreover, $\mathcal{A}'$ is efficient given that $\mathcal{A}$ is efficient.

[3] In quantum information theory, it is known as *the gentle measurement lemma* [Win99].

Sketch. By deferred measurement principle, we can modify $\mathcal{A}$ so that it applies a unitary transformation followed by measuring and outputting a value y. Fix $x \in X$, then the statement follows after applying Lemma 2 with respect to the 2-outcome measurement which checks whether $y = f(x)$ or not. □

3.3 Quantum Fully Homomorphic Encryption

A fully homomorphic encryption scheme allows for publicly evaluating an encryption of x using a function f to obtain an encryption of $f(x)$. Traditionally f has been modeled as classical circuits but in this work, we consider the setting when f is modeled as quantum circuits and when the messages are quantum states. This notion is referred to as quantum fully homomorphic encryption (QFHE). We state our definition borrowed directly from[4] [BJ15].

Definition 4. *Let $\mathcal{M}$ be the Hilbert space associated with the message space (plaintexts) and $\mathcal{C}$ be the Hilbert space associated with the ciphertexts. A quantum fully homomorphic encryption (QFHE) scheme is a tuple of QPT algorithms* $\mathsf{QFHE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$*:*

- $\mathsf{QFHE.Gen}(1^\lambda)$*: Takes as input a security parameter λ in unary; outputs a classical public-secret key pair,* $(\mathsf{PK}, \mathsf{SK})$.
- $\mathsf{QFHE.Enc}(\mathsf{PK}, \cdot) : \mathcal{D}(\mathcal{M}) \to \mathcal{D}(\mathcal{C})$*: Takes as input a public key* PK *and a quantum message ρ; outputs a quantum ciphertext σ.*
- $\mathsf{QFHE.Dec}(\mathsf{SK}, \cdot) : \mathcal{D}(\mathcal{C}) \to \mathcal{D}(\mathcal{M})$*: Takes as input a secret key* SK *and a quantum ciphertext σ; outputs a quantum message ρ.*
- $\mathsf{QFHE.Eval}(\mathsf{PK}, \mathcal{E}, \cdot) : \mathcal{D}(\mathcal{C}^{\otimes n}) \to \mathcal{D}(\mathcal{C}^{\otimes m})$*: Takes as input a public key* PK*, description of a quantum circuit $\mathcal{E} : \mathcal{D}(\mathcal{M}^{\otimes n}) \to \mathcal{D}(\mathcal{M}^{\otimes m})$, and a tuple of quantum ciphertexts $\sigma \in (\mathcal{D}(\mathcal{C}^{\otimes n}))$; outputs a tuple of quantum ciphertexts $\sigma' \in \mathcal{D}(\mathcal{C}^{\otimes m})$.*

Semantic security and compactness are defined analogously to the classical setting, and we defer to [BJ15] for a definition. For the impossibility result, we require a QFHE scheme where ciphertexts of classical plaintexts are also classical. Given any classical message $x \in \{0,1\}^k$, we want $\mathsf{QFHE.Enc}(\mathsf{PK}, |x\rangle\langle x|)$ to be a computational basis state $|z\rangle\langle z|$ for some $z \in \{0,1\}^l$ (here, l is the length of ciphertexts for k-qubit messages). In this case, we write $\mathsf{QFHE.Enc}_{\mathsf{PK}}(x)$. We also want the same to be true for evaluated ciphertexts, i.e. if $\mathcal{E}(|x\rangle\langle x|) = |y\rangle\langle y|$ for some $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$, then

$$\mathsf{QFHE.Enc}_{\mathsf{PK}}(y) \leftarrow \mathsf{QFHE.Eval}(\mathsf{PK}, \mathcal{E}, \mathsf{QFHE.Enc}_{\mathsf{PK}}(x))$$

is a classical ciphertext of y.

3.3.1 Circuit Privacy

An additional property we need a QFHE scheme to satisfy is called *malicious circuit privacy.* Informally, it states that the evaluation algorithm does not leak any information about the circuit being evaluated, i.e. given a homomorphic evaluation of a circuit $\mathcal{E}$, an adversary cannot learn

[4] With the slight modification that the evaluation key is included as part of the public key.

any information she would not have learned given only black-box access to $\mathcal{E}$. We give the formal definition below, adapted from[5] [CDM20]:

Definition 5 (Malicious Circuit Privacy). *A QFHE scheme* $\mathsf{QFHE} = (\mathsf{QFHE.Gen}, \mathsf{QFHE.Enc}, \mathsf{QFHE.Dec}, \mathsf{QFHE.Eval})$ *satisfies* ***malicious circuit privacy*** *if there exist unbounded quantum algorithms* $\mathsf{QFHE.Ext}$ *and* $\mathsf{QFHE.Sim}$ *such that for any (possibly invalid) public key* PK^**, any quantum circuit* $\mathcal{E} : \mathcal{D}(\mathcal{M}^{\otimes n}) \to \mathcal{D}(\mathcal{M}^{\otimes m})$*, any ancillary register* A *with* $\mathcal{H}_A = \mathcal{M}^{\otimes \mathrm{poly}(\lambda)}$*, and any* $\rho \in \mathcal{D}(\mathcal{C}^{\otimes n} \otimes \mathcal{H}_A)$ *we have*

$$(\mathsf{QFHE.Eval}(\mathsf{PK}^*, \mathcal{E}, \cdot) \otimes \mathbf{id}_A)\, \rho \approx_s \mathsf{QFHE.Sim}(1^\lambda, (\mathcal{E} \otimes \mathbf{id}_{AB})(\widetilde{\rho})),$$

where $\widetilde{\rho} \in \mathcal{D}(\mathcal{C}^{\otimes n} \otimes \mathcal{H}_A \otimes \mathcal{H}_B)$ *is obtained by applying* $\mathsf{QFHE.Ext}(1^\lambda, \mathsf{PK}^*, \cdot)$ *to the* $\mathcal{C}^{\otimes n}$ *(ciphertext) register of* ρ*, and* B *is an ancillary register.*

The definition can be alternatively stated as follows: for an unbounded adversary, one query $\mathsf{QFHE.Eval}$ evaluation of the circuit $\mathcal{E}$ is no more powerful than one query to $\mathcal{E}$ itself. Note also that A is a hidden register held by an adversary that could be entangled with the ciphertext, so it is not accessible by the algorithms $\mathsf{QFHE.Eval}$ and $\mathsf{QFHE.Ext}$.

Instantiation. Malicious Circuit Privacy can be achieved using a regular QFHE scheme [CDM20]. We state their result as a theorem:

Theorem 6. *Assuming QLWE and circular security*[6]*, there exists a QFHE scheme with malicious circuit privacy, which reduces to a classical FHE scheme for classical inputs.*

3.4 Compute-and-Compare Circuits and Quantum-Secure Lockable Obfuscation

Compute-and-compare Circuits. The subclass of circuits that we are interested in is called compute-and-compare circuits, denoted by $\mathcal{C}_{\mathsf{cnc}}$. A compute-and-compare circuit is of the following form: $\mathbf{C}[C, \alpha, \beta]$, where α is called a lock and C has output length $|\alpha|$, is defined as follows:

$$\mathbf{C}[C, \alpha](x) = \begin{cases} \beta, & \text{if } C(x){=}\alpha, \\ 0, & \text{otherwise} \end{cases}$$

Definition 7 (Quantum-Secure Lockable Obfuscation). *An obfuscation scheme* $(\mathsf{Obf}, \mathsf{ObfEval})$ *for a class of compute-and-compare circuits* $\mathcal{C}_{\mathsf{cnc}}$ *is said to be a* ***quantum-secure lockable obfuscation scheme*** *if the following properties are satisfied:*

- *It satisfies the functionality of obfuscation.*
- **Security***: For every polynomial-sized circuit* C*, string* $\beta \in \{0,1\}^{\mathrm{poly}(\lambda)}$*,for every QPT adversary* $\mathcal{A}$ *there exists a QPT simulator* Sim *such that the following holds: sample* $\alpha \xleftarrow{\$} \{0,1\}^{\mathrm{poly}(\lambda)}$*,*

  $$\left\{\mathsf{Obf}\left(1^\lambda, \mathbf{C}\right)\right\} \approx_{Q,\varepsilon} \left\{\mathsf{Sim}\left(1^\lambda, 1^{|C|}\right)\right\},$$

  where $\mathbf{C}$ *is a circuit parameterized by* C, α, β *with* $\varepsilon \leq \frac{1}{2^{|\alpha|}}$*.*

[5] The difference is that we sensibly strengthen the definition to account for adversaries with a possibly entangled auxiliary register B. Even though this was not formally considered by [CDM20], a quick analysis of their argument reveals that their construction indeed satisfies this stronger definition of circuit privacy.

[6] This assumption is needed to evaluate circuits with unbounded depth in both classical and quantum FHE.

Instantiation. The works of [WZ17, GKW17, GKVW19] construct a lockable obfuscation scheme based on polynomial-security of learning with errors. Since learning with errors is conjectured to be hard against QPT algorithms, the obfuscation schemes of [WZ17, GKW17, GKVW19] are also secure against QPT algorithms. Thus, we have the following theorem.

Theorem 8 ([GKW17, WZ17, GKVW19])**.** *Assuming quantum hardness of learning with errors, there exists a quantum-secure lockable obfuscation scheme.*

3.5 Classical-Accessible Oracles

We call an oracle $\mathcal{O} : \mathcal{X} \to \mathcal{Y}$ *classical-accessible* if it only accepts a classical query $x \in X$, to which it responds with the classical value $\mathcal{O}(x)$. More formally, a query to $\mathcal{O}$ made by a quantum algorithm $\mathcal{A}$ can be described as the following quantum operation: measure the $\mathcal{X}$ (query) register of $\mathcal{A}$ in the computational basis to obtain $x \in \mathcal{X}$, then XOR the value $\mathcal{O}(x)$ to the $\mathcal{Y}$ (answer) register. In other words, if $\mathcal{A}$ queries the oracle in state $\sum_{x,y,z} \alpha_{x,y,z}|x\rangle|y\rangle|z\rangle$, where $\mathcal{Z}$ is an ancillary register corresponding to the internal state of $\mathcal{A}$, then the state after the query equals $\sum_{y,z} \beta_{y,z}|x\rangle|y \oplus \mathcal{O}(x)\rangle|z\rangle$ with probability $p_x = \sum_{y,z} |\alpha_{x,y,z}|^2$, where $\beta_{y,z} = \alpha_{x,y,z}/\sqrt{p_x}$.

3.5.1 Classical-Accessible Random Oracle Model

In classical-accessible random oracle model (CAROM), the function in question is assumed to be a uniformly random function $\mathcal{O} : \mathcal{X} \to \mathcal{Y}$ and modeled as a classical-accessible oracle.

A classical-accessible random oracle $\mathcal{O} : X \to Y$ can be efficiently simulated *on-the-fly* as follows:

- Create an empty database $D \subset X \times Y$.
- On query $x \in X$, check if D contains x. If yes, answer consistently; otherwise sample $y \xleftarrow{\$} Y$, add (x, y) to D, and answer with y.

A classical-accessible oracle simulated *on-the-fly* is perfectly indistinguishable from a classical-accessible random oracle.

3.6 Copy-Protection

Below we present the definition of a copy-protection scheme, adapted from [BJL+21] and originally due to [Aar09].

Definition 9 (Copy-Protection Scheme)**.** *Let $\mathcal{F} = \mathcal{F}(\lambda)$ be a class of efficiently computable functions of the form $f : X \to Y$. A copy protection scheme for $\mathcal{F}$ is a pair of QPT algorithms* $(\mathsf{CP}, \mathsf{Eval})$ *such that for some output space $\mathcal{D}(\mathcal{H}_Z)$:*

- ***Copy Protected State Generation:*** $\mathsf{CP}(1^\lambda, d_f)$ *takes as input the security parameter 1^λ and a classical description d_f of a function $f \in \mathcal{F}$ (that efficiently computes f). It outputs a mixed state $\rho_f \in \mathcal{D}(\mathcal{H}_Z)$.*
- ***Evaluation:*** $\mathsf{Eval}(1^\lambda, \rho, x)$ *takes as input the security parameter 1^λ, a mixed state $\rho \in \mathcal{D}(\mathcal{H}_Z)$, and an input value $x \in X$. It outputs a bipartite state $\rho' \otimes |y\rangle\langle y| \in \mathcal{D}(\mathcal{H}_Z) \otimes \mathcal{D}(\mathcal{H}_Y)$.*

We will sometimes abuse the notation and write $\mathsf{Eval}(1^\lambda, \rho, x)$ to denote either the classical output $y \in Y$ or the residual state ρ' alone when the other is insignificant. Note that when we work in CAROM, the algorithms $(\mathsf{CP}, \mathsf{Eval})$ will have access to the random oracle and be written as $(\mathsf{CP}^{\mathcal{O}}, \mathsf{Eval}^{\mathcal{O}})$.

There are three properties we require of a copy-protection scheme: correctness, reusability, and security.

Correctness: Informally speaking, if an honestly generated copy-protected state ρ_f for a function $f \in \mathcal{F}$ is honestly evaluated using Eval on any input $x \in X$, the output should be $f(x)$. We present the formal definition below:

Definition 10 (Correctness)**.** *A copy-protection scheme* $(\mathsf{CP}, \mathsf{Eval})$ *for* $\mathcal{F}$ *is* δ***-correct*** *if the following holds for every* $x \in X$*,* $f \in \mathcal{F}$*:*

$$\Pr\left[f(x) \leftarrow \mathsf{Eval}(1^\lambda, \rho_f, x) \ : \ \rho_f \leftarrow \mathsf{CP}(1^\lambda, d_f)\right] \geq \delta.$$

Correctness property can be relaxed by averaging over inputs x sampled from a distribution $\mathcal{D}$.

Definition 11 (Mean-Correctness)**.** *Let* $\mathcal{D}_X$ *be a distribution over* X*. A copy-protection scheme* $(\mathsf{CP}, \mathsf{Eval})$ *for* $\mathcal{F}$ *is* $(\mathcal{D}_X, \delta)$***-mean-correct*** *if the following holds for every* $f \in \mathcal{F}$*:*

$$\Pr\left[f(x) \leftarrow \mathsf{Eval}(1^\lambda, \rho_f, x) \ : \ {\scriptstyle \substack{\rho_f \leftarrow \mathsf{CP}(1^\lambda, d_f) \\ x \leftarrow \mathcal{D}_X}}\right] \geq \delta.$$

Reusability: The correctness notions we define above are for a single evaluation. In practice, we would like our copy-protected program to evaluate polynomially many inputs without losing its functionality. Accordingly, we define reusability below as a stronger version of these correctness notions.

Definition 12 (Reusability)**.** *Let* $(\mathsf{CP}, \mathsf{Eval})$ *be a* δ*-correct copy-protection scheme for* $\mathcal{F}$*. Then,* $(\mathsf{CP}, \mathsf{Eval})$ *is called* ***reusable*** *if the following holds for every* $m = \mathrm{poly}(\lambda)$*, every* $(x_1, \ldots, x_m) \in X^m$ *and every* $f \in \mathcal{F}$*:*

$$\Pr\left[f(x_m) \leftarrow \mathsf{Eval}(1^\lambda, \rho_f^m, x_m) \ : \ {\scriptstyle \substack{\rho_f^1 \leftarrow \mathsf{CP}(1^\lambda, d_f) \\ \rho_f^{i+1} \leftarrow \mathsf{Eval}(1^\lambda, \rho_f^i, x_i),\ 1 \leq i \leq m-1}}\right] \geq \delta - \mathsf{negl}(\lambda).$$

Similarly, let $(\mathsf{CP}, \mathsf{Eval})$ *be a* $(\mathcal{D}_X, \delta)$*-mean-correct copy-protection scheme for* $\mathcal{F}$*. Then,* $(\mathsf{CP}, \mathsf{Eval})$ *is called* ***reusable*** *if the following holds for any* $m = \mathrm{poly}(\lambda)$ *and any* $f \in \mathcal{F}$*:*

$$\Pr\left[f(x_m) \leftarrow \mathsf{Eval}(1^\lambda, \rho_f^m, x_m) \ : \ {\scriptstyle \substack{\rho_f^1 \leftarrow \mathsf{CP}(1^\lambda, d_f) \\ x_i \leftarrow \mathcal{D}_X,\ 1 \leq i \leq m \\ \rho_f^{i+1} \leftarrow \mathsf{Eval}(1^\lambda, \rho_f^i, x_i),\ 1 \leq i \leq m-1}}\right] \geq \delta - \mathsf{negl}(\lambda).$$

Remark 13. *In the plain model or the quantum-accessible oracle models, reusability is implied by correctness by Corollary 3. In particular,* $\delta = 1 - \mathsf{negl}(\lambda)$ *yields* $\gamma = \mathsf{negl}(\lambda)$ *and* $\delta = 1 - 1/\mathrm{poly}(\lambda)$ *yields* $\gamma = 1/\mathrm{poly}(\lambda)$*. However, in the classical-accessible setting such an implication is unclear due to the fact that the oracle queries force an intermediate measurement that cannot be pushed to the end.*

Security: Security in the context of copy-protection means that given a copy-protected program ρ_f of a function $f \in \mathcal{F}$, no QPT adversary can produce two programs that can both be used to compute f. This is captured in the following definition adapted from the "malicious-malicious security" definition in [BJL+21]:

Definition 14 (Piracy Experiment)**.** *A* ***piracy experiment*** *is a game defined by a copy-protection scheme* $(\mathsf{CP}, \mathsf{Eval})$*, a distribution* $\mathcal{D}_{\mathcal{F}}$ *over* $\mathcal{F}$*, and a class of distributions* $\mathfrak{D}_X = \{\mathfrak{D}_X(f)\}_{f\in\mathcal{F}}$ *over* X*. It is the following game between a challenger and an adversary, which is a triplet of algorithms* $(\mathcal{A}, \mathcal{B}, \mathcal{C})$*:*

- ***Setup Phase:*** *The challenger samples a function* $f \leftarrow \mathcal{D}_{\mathcal{F}}$ *and sends* $\rho_f \leftarrow \mathsf{CP}(1^\lambda, d_f)$ *to* $\mathcal{A}$*.*
- ***Splitting Phase:*** $\mathcal{A}$ *applies a CPTP map to split* ρ_f *into a bipartite state* ρ_{BC}*; she sends the* B *register to* $\mathcal{B}$ *and the* C *register to* $\mathcal{C}$*. No communication is allowed between* $\mathcal{B}$ *and* $\mathcal{C}$ *after this phase.*
- ***Challenge Phase:*** *The challenger samples* $(x_B, x_C) \leftarrow \mathfrak{D}_X(f) \times \mathfrak{D}_X(f)$ *and sends* x_B, x_C *to* $\mathcal{B}, \mathcal{C}$*, respectively.*
- ***Output Phase:*** $\mathcal{B}$ *and* $\mathcal{C}$ *output*[7] $y_B \in Y$ *and* $y_C \in Y$*, respectively, and send to the challenger. The challenger outputs 1 if* $y_B = f(x_B)$ *and* $y_C = f(x_C)$*, indicating that the adversary has succeeded, and 0 otherwise.*

The bit output by the challenger is denoted by $\mathsf{PirExp}^{\mathsf{CP},\mathsf{Eval}}_{\mathcal{D}_{\mathcal{F}},\mathfrak{D}_X}(1^\lambda, (\mathcal{A}, \mathcal{B}, \mathcal{C}))$*.*

Definition 15 (Copy-Protection Security)**.** *Let* $(\mathsf{CP}, \mathsf{Eval})$ *be a copy-protection scheme for a class* $\mathcal{F}$ *of functions* $f : X \to Y$*. Let* $\mathcal{D}_{\mathcal{F}}$ *be a distribution over* $\mathcal{F}$ *and* $\mathfrak{D}_X = \{\mathfrak{D}_X(f)\}_{f\in\mathcal{F}}$ *a class of distributions over* X*. Then,* $(\mathsf{CP}, \mathsf{Eval})$ *is called* $(\mathcal{D}_{\mathcal{F}}, \mathfrak{D}_X, \delta)$*-**secure** if any QPT adversary* $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ *satisfies*

$$\mathsf{Pr}\left[b = 1 \ : \ b \leftarrow \mathsf{PirExp}^{\mathsf{CP},\mathsf{Eval}}_{\mathcal{D}_{\mathcal{F}},\mathfrak{D}_X}\left(1^\lambda, (\mathcal{A}, \mathcal{B}, \mathcal{C})\right)\right] \leq \delta.$$

Note that this definition is referred to as malicious-malicious security because the adversary is free to choose the registers B, C as well as the evaluation algorithms used by $\mathcal{B}$ and $\mathcal{C}$.

3.7 Quantum-Unlearnability

Below we state the definition of a quantum unlearnable circuit class from [ALP21], which states that a QPT adversary cannot output a quantum implementation of $C \in \mathcal{C}$ given oracle access:

Definition 16. *A circuit class* $\mathcal{C}$ *is* ν*-**quantum unlearnable** with respect to distribution* $\mathcal{D}_{\mathcal{C}}$ *over* $\mathcal{C}$ *if for any quantum adversary* $\mathcal{A}^C$ *making at most* $\mathrm{poly}(\lambda)$ *queries, we have*

$$\mathsf{Pr}\left[\forall x, \mathsf{Pr}[U^*(\rho^*, x) = C(x)] \geq \nu \ : \ \substack{C \leftarrow \mathcal{D}_{\mathcal{C}} \\ (U^*, \rho^*) \leftarrow \mathcal{A}^{C(\cdot)}(1^\lambda)}\right] \leq \mathsf{negl}(\lambda).$$

If there exists a distribution $\mathcal{D}_{\mathcal{C}}$ *satisfying this property, then* $\mathcal{C}$ *is simply called* ν***-quantum unlearnable****.*

[7] Since $\mathcal{B}$ and $\mathcal{C}$ cannot communicate, the order in which they use their share of the copy-protected program is insignificant.

4 Impossibility in Classical-Accessible Random Oracle Model

We show the infeasibility of copy-protection in the classical-accessible *random* oracle model (CAROM). We prove this in two steps:

- First, we show the impossibility of approximately correct copy-protection (even without reusability) in the plain model.
- Next, we show that any copy-protection in CAROM can be transformed into approximately-correct copy-protection in the plain model. Invoking the above result, it follows that copy-protection in CAROM is impossible.

4.1 Impossibility of Approximate Copy-Protection

Following the approach of [ALP21], we will construct a class of circuits, which cannot be learned via oracle access, yet can be learned given the quantum circuit. The key tool enabling the latter is the power of homomorphic evaluation. The class of circuits we construct is related to but different from that constructed by [ALP21].

Theorem 17. *Assuming the existence of QFHE with malicious circuit privacy and QLWE, there exists an unlearnable class $\mathcal{G}$ of circuits of the form $G : X \to Y$ and an input distribution $\mathcal{D}_X$ over X such that there exists no copy-protection scheme* $(\mathsf{CP}, \mathsf{Eval})$ *in the plain model for $\mathcal{G}$ with $(\mathcal{D}_X, 1-\varepsilon(\lambda))$-mean-correctness and $(\mathcal{D}_{\mathcal{G}}, \mathfrak{D}_X, \delta(\lambda))$-security for any (ε, δ) satisfying $\delta \leq 1 - 3\sqrt{\varepsilon}$ and for any $\mathcal{D}_{\mathcal{G}}, \mathfrak{D}_X$.*

Proof. Let Obf be a quantum-secure lockable obfuscation scheme (see Section 3.4) and QFHE be a quantum fully homomorphic encryption scheme satisfying malicious circuit privacy (see Section 3.3). We first recall the circuit class $\mathcal{C}$ used in the impossibility result of [ALP21], presented in Figure 1. Every circuit in $\mathcal{C}$ is of the form $C_{a,b,r,\mathsf{PK},\mathcal{O}}$, where $a, b, r, \mathsf{PK}, \mathcal{O}$ are parameters described below.

- $a, b, r \in \{0,1\}^\lambda$.
- $(\mathsf{PK}, \mathsf{SK})$ is in the support of $\mathsf{QFHE.Gen}(1^\lambda)$.
- $\mathcal{O}$ is in the support of $\mathsf{Obf}(\mathsf{CC}[\mathsf{QFHE.Dec}(\mathsf{SK}, \cdot), b, (\mathsf{SK}||r)])$.

Let $\widetilde{\mathsf{QFHE}} = (\widetilde{\mathsf{QFHE}}.\mathsf{Setup}, \widetilde{\mathsf{QFHE}}.\mathsf{Enc}, \widetilde{\mathsf{QFHE}}.\mathsf{Dec}, \widetilde{\mathsf{QFHE}}.\mathsf{Eval})$ be a QFHE scheme with malicious circuit privacy (Definition 5).

Using $\mathcal{C}$, we define[8] another class of circuits $\mathcal{G}$ in Figure 2. We show that $\mathcal{G}$ cannot be copy-protected. Every circuit in $\mathcal{G}$ is of the form $G_{C_{a,b,r,\mathsf{PK},\mathcal{O}}}$, which is a fixed circuit parameterized by a circuit $C_{a,b,r,\mathsf{PK},\mathcal{O}} \in \mathcal{C}$.

Unlearnability. We will first show that $\mathcal{G}$ is unlearnable with respect to a distribution $\mathcal{D}_{\mathcal{C}}(\lambda)$, which we define as follows: $\mathcal{D}_{\mathcal{C}}(\lambda)$ outputs a circuit from $\mathcal{C}_\lambda$ by sampling $a, b, r \xleftarrow{\$} \{0,1\}^\lambda$, then computing $(\mathsf{PK}, \mathsf{SK}) \leftarrow \mathsf{QFHE.Gen}(1^\lambda)$, and finally computing an obfuscation $\mathcal{O} \leftarrow \mathsf{Obf}(\mathsf{CC}[\mathsf{QFHE.Dec}(\mathsf{SK}, \cdot),$ $(b, (\mathsf{SK}|r))])$, where CC is a compute-and-compare circuit. Since every $G_C \in \mathcal{G}$ is uniquely determined by the description of $C \in \mathcal{C}$, $\mathcal{D}_{\mathcal{C}}(\lambda)$ induces a distribution $\mathcal{D}_{\mathcal{G}}(\lambda)$ on $\mathcal{G}$ given by $G_C : C \leftarrow \mathcal{D}_{\mathcal{C}}$.

We cite the following result:

[8] In both $\mathcal{C}$ and $\mathcal{G}$, we assume padding with zeros appropriately to make input/output lengths compatible.

```
$C_{a,b,r,\mathsf{PK},\mathcal{O}}(x)$:

  1. If $x = 0$, output $\mathsf{QFHE.Enc}(\mathsf{PK}, a; r)\,|\mathcal{O}|\mathsf{PK}$, where $\mathcal{O}$ is generated as follows:
     $\mathcal{O} \leftarrow \mathsf{Obf}(\mathsf{CC}[\mathsf{QFHE.Dec}(\mathsf{SK}, \cdot), b, (\mathsf{SK}|r)])$, where $\mathsf{CC}$ is a compute-and-compare circuit.
  2. Else if $x = a$, output $b$.
  3. Otherwise, output 0
```

Figure 1: Description of a circuit $C_{a,b,r,\mathsf{PK},\mathcal{O}}$ in $\mathcal{C}$.

```
$G_{C_{a,b,r,\mathsf{PK},\mathcal{O}}}(\widetilde{\mathsf{PK}}, X)$:

  1. Parse $X$ as ciphertext $CT$.
  2. Compute and output $CT^* \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Eval}_{\widetilde{\mathsf{PK}}}(C_{a,b,r,\mathsf{PK},\mathcal{O}}, CT)$.
```

Figure 2: Description of a circuit $G \in \mathcal{G}$, used for proving impossibility of approximate CP

Lemma 18 (Proposition 46 in [ALP21])**.** *For any non-negligible function $\nu = \nu(\lambda)$, $\mathcal{C}$ is ν-quantum unlearnable with respect to $\mathcal{D}_{\mathcal{C}}(\lambda)$.*

Using this as a black-box, we can now prove unlearnability of $\mathcal{G}$:

Lemma 19. *$\mathcal{G}$ is unlearnable with respect to the distribution $\mathcal{D}_{\mathcal{G}}(\lambda)$.*

Proof. Suppose that there exists an unbounded quantum adversary $\mathcal{A}^{G_C(\cdot)}$ which makes at most poly(λ) queries and satisfies

$$\mathsf{Pr}\left[\forall x, \mathsf{Pr}[U^*(\rho^*, x) = G(x)] \geq \nu(\lambda) \ : \ \substack{C \leftarrow \mathcal{D}_{\mathcal{C}} \\ (U^*, \rho^*) \leftarrow \mathcal{A}^{G_C(\cdot)}(1^\lambda)}\right] \geq \mu(\lambda)$$

for some non-negligible functions ν, μ. We will use $\mathcal{A}^{G_C(\cdot)}$ to construct an adversary which violates the unlearnability of C.
We will proceed in two steps:

- By malicious circuit privacy of $\widetilde{\mathsf{QFHE}}$, there exists an unbounded quantum adversary $(\mathcal{A}')^{C(\cdot)}$ which makes at most poly(λ) queries to $C(\cdot)$ and satisfies

$$\begin{aligned}&\mathsf{Pr}\left[\forall x, \mathsf{Pr}[U^*(\rho^*, x) = G(x)] \geq \nu(\lambda) - \mathsf{negl}(\lambda) \ : \ \substack{C \leftarrow \mathcal{D}_{\mathcal{C}} \\ (U^*, \rho^*) \leftarrow (\mathcal{A}')^{C(\cdot)}(1^\lambda)}\right] \\ &\geq \mu(\lambda) - \mathsf{negl}(\lambda). \qquad (2)\end{aligned}$$

$\mathcal{A}'$ simply uses $(\widetilde{\mathsf{QFHE}}.\mathsf{Sim}, \widetilde{\mathsf{QFHE}}.\mathsf{Ext})$ and oracle access to $C(\cdot)$ to simulate the queries of $\mathcal{A}$.

- Using $(\mathcal{A}')^{C(\cdot)}$, we will construct $(\widetilde{\mathcal{A}'})^{C(\cdot)}$ which makes at most $\text{poly}(\lambda)$ queries to $C(\cdot)$ and satisfies

$$\begin{aligned}&\mathsf{Pr}\left[\forall x, \mathsf{Pr}[U^*(\rho^*, x) = C(x)] \geq \nu(\lambda) - \mathsf{negl}(\lambda) \ : \ {}^{C \leftarrow \mathcal{D}_{\mathcal{C}}}_{(U^*,\rho^*) \leftarrow (\widetilde{\mathcal{A}'})^{C(\cdot)}(1^\lambda)}\right] \\ &\geq \mu(\lambda) - \mathsf{negl}(\lambda).\end{aligned} \tag{3}$$

$(\widetilde{\mathcal{A}'})^{C(\cdot)}$ does the following:

  - Run $(\mathcal{A}')^{C(\cdot)}(1^\lambda)$ to obtain (U, ρ)
  - Output (U^*, ρ), where $U^*(\rho, x)$ does the following:
    1. Compute $(\mathsf{PK}, \mathsf{SK}) \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Setup}(1^\lambda)$ and $\mathsf{CT} \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Enc}(\mathsf{PK}, x)$
    2. Compute $\mathsf{CT}^* \leftarrow U(\rho, (\mathsf{PK}, \mathsf{CT}))$.
    3. Output $y \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Dec}(\mathsf{SK}, \mathsf{CT}^*)$.

Conditioned on (U, ρ) satisfying the event in eq. (2), (U, ρ^*) will satisfy the event in eq. (3) by correctness of $\widetilde{\mathsf{QFHE}}$ evaluation, thereby violating μ-quantum unlearnability of $\mathcal{C}$.

□

Lemma 20. *There exists $\mathcal{D}_X$ such that there is no copy-protection scheme $(\mathsf{CP}, \mathsf{Eval})$ in the plain model for $\mathcal{G}$ with $(\mathcal{D}_X, 1 - \varepsilon(\lambda))$-mean-correctness and $(\mathcal{D}_{\mathcal{G}}, \mathfrak{D}_X, \delta(\lambda))$-security for any $\mathfrak{D}_X$ and any ε, δ satisfying $\delta \leq 1 - 3\sqrt{\varepsilon}$.*

Proof. Define

$$\mathcal{D}_X := \left\{\widetilde{\mathsf{QFHE}}.\mathsf{Enc}_{\widetilde{\mathsf{PK}}}(x) : {}^{(\widetilde{\mathsf{PK}},\widetilde{\mathsf{SK}}) \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Setup}(1^\lambda)}_{x \xleftarrow{\$} \{0,1\}^{\text{poly}(\lambda)}}\right\}$$

Suppose there is an approximate copy-protection scheme $(\mathsf{CP}, \mathsf{Eval})$ for $\mathcal{G}$ with $1 - \varepsilon(\lambda)$-mean-correctness. Let $\rho \leftarrow \mathsf{CP}(1^\lambda, G_{C_{a,b,r,\mathsf{PK},\mathcal{O}}})$. We construct a QPT adversary $\mathcal{A}$ in Figure 3 that given ρ, can recover an approximate classical description of G with probability greater than $\delta(\lambda)$, hence violating security[9].

Analysis of $\mathcal{A}(\rho)$: By approximate correctness of $(\mathsf{CP}, \mathsf{Eval})$, we have

$$\begin{aligned}&\mathsf{Pr}\left[G_C(x) \leftarrow \mathsf{Eval}(1^\lambda, \rho, x) \ : \ {}^{\rho \leftarrow \mathsf{CP}(1^\lambda, G_C)}_{x \leftarrow \mathcal{D}_X}\right] \\ &= \mathsf{Pr}\left[G_C(x) \leftarrow \mathsf{Eval}(1^\lambda, \rho, \widetilde{\mathsf{QFHE}}.\mathsf{Enc}_{\widetilde{\mathsf{PK}}}(x)) \ : \ \begin{array}{c}\rho \leftarrow \mathsf{CP}(1^\lambda, G_C) \\ (\widetilde{\mathsf{PK}},\widetilde{\mathsf{SK}}) \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Setup}(1^\lambda) \\ x \xleftarrow{\$} \{0,1\}^{\text{poly}(\lambda)}\end{array}\right] \\ &\geq 1 - \varepsilon(\lambda).\end{aligned} \tag{4}$$

[9] The attack is simply to recover the classical description and send it to both parties at the splitting phase. Note that this attack succeeds irrespective of the test distributions $\mathfrak{D}_X$.

$\mathcal{A}(\rho)$:

1. Compute $(\widetilde{\mathsf{PK}}, \widetilde{\mathsf{SK}}) \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Setup}(1^\lambda)$.
2. Compute $CT \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Enc}_{\widetilde{\mathsf{PK}}}(0)$.
3. Run the copy-protection evaluation, $\rho' \otimes CT^* \leftarrow \mathsf{Eval}(1^\lambda, \rho, CT)$.
4. Run the decryption, $(\mathsf{CT}_a|\mathcal{O}|\mathsf{PK}) \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Dec}_{\widetilde{\mathsf{SK}}}(CT^*)$. Since CT_a is classical, maintain a copy of CT_a.
5. Run the QFHE homomorphic evaluation,

   $$\mathsf{CT}_b^* \leftarrow \mathsf{QFHE}.\mathsf{Eval}_{\mathsf{PK}}(\Phi, \mathsf{CT}_a),$$

   where Φ is a quantum circuit that on input σ, does the following:

   (a) Compute $\sigma_1 \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Enc}(\widetilde{\mathsf{PK}}, \sigma)$.

   (b) Compute $\sigma_2 \leftarrow \mathsf{QFHE}.\mathsf{Eval}(\mathsf{PK}, \mathsf{Eval}(1^\lambda, \cdot, \cdot), \mathsf{QFHE}.\mathsf{Enc}(\widetilde{\mathsf{PK}}, \rho'), \sigma_1)$.

   (c) Compute $\sigma_3 \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Dec}(\widetilde{\mathsf{SK}}, \sigma_2)$ and output σ_3.
6. Compute the unitary $\mathcal{O}$ on CT_b^* and measure basis to obtain $(sk'|r')$.
7. Compute $a \leftarrow \mathsf{QFHE}.\mathsf{Dec}(sk', \mathsf{CT}_a)$ and $b \leftarrow \mathsf{QFHE}.\mathsf{Dec}(sk', \mathsf{CT}_b^*)$.
8. Output $G_{C_{a,b,r',\mathsf{PK},\mathcal{O}}}$.

Figure 3: Description of $\mathcal{A}$

Define

$$\zeta_x := \Pr\left[G_C(x) \leftarrow \mathsf{Eval}(1^\lambda, \rho, \widetilde{\mathsf{QFHE}}.\mathsf{Enc}_{\widetilde{\mathsf{PK}}}(x)) \;:\; \substack{\rho \leftarrow \mathsf{CP}(1^\lambda, G_C) \\ (\widetilde{\mathsf{PK}}, \widetilde{\mathsf{SK}}) \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Setup}(1^\lambda)}\right], \tag{5}$$

so that eq. (4) can be written as

$$\mathop{\mathbb{E}}_{x \xleftarrow{\$} \{0,1\}^{\mathrm{poly}(\lambda)}} [\zeta_x] \geq 1 - \varepsilon(\lambda). \tag{6}$$

We observe that $|\zeta_x - \zeta_{x'}| \leq \mathsf{negl}(\lambda)$ for any $x, x' \in \{0,1\}^{\mathrm{poly}(\lambda)}$. To see this, suppose the difference is not negligible. Define an adversary $\mathcal{A}'$ who can break semantic security of $\widetilde{\mathsf{QFHE}}$. Given a ciphertext $CT \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Enc}(\widetilde{\mathsf{PK}}, x)$, $\mathcal{A}'$ does the following:

- Sample $\rho \leftarrow \mathsf{CP}(1^\lambda, G_C)$
- Output 1 if $G_C(x) \leftarrow \mathsf{Eval}(1^\lambda, \rho, CT)$, and 0 otherwise.

$\mathcal{A}'$ outputs 1 with probability ζ_x, hence she can distinguish encryptions of x and x', contradiction. Hence, $|\zeta_x - \zeta_0| \leq \mathsf{negl}(\lambda)$ for all x. Let E_1 be the event that $CT^* = \widetilde{\mathsf{QFHE}}.\mathsf{Enc}(\widetilde{\mathsf{PK}}, \mathsf{QFHE}.\mathsf{Enc}_{\mathsf{PK}}(a)$ $|\mathcal{O}|\mathsf{PK})$, then by eq. (6) we have

$$\Pr[E_1] = \zeta_0 \geq \underset{x \xleftarrow{\$} \{0,1\}^{\mathrm{poly}(\lambda)}}{\mathbb{E}} [\zeta_x - \mathsf{negl}(\lambda)] \geq 1 - \varepsilon(\lambda) - \mathsf{negl}(\lambda). \tag{7}$$

At this stage, by modifying Eval if necessary, we can assume $T(\rho', \rho) \leq \sqrt{\varepsilon(\lambda)} + \mathsf{negl}(\lambda)$ by Corollary 3. Let E_2 be the event $\mathcal{A}$ succeeds in step 4; in particular, E_2 implies $\mathsf{CT}_a = \mathsf{QFHE}.\mathsf{Enc}(\mathsf{PK}, a)$. By $\widetilde{\mathsf{QFHE}}$ correctness we have $\Pr[E_2 \mid E_1] \geq 1 - \mathsf{negl}(\lambda)$.

Conditioned on E_2, step 5b is nothing but a QFHE homomorphic evaluation of $\mathsf{Eval}(1^\lambda, \rho', \sigma_1)$, where $\sigma_1 \leftarrow \widetilde{\mathsf{QFHE}}.\mathsf{Enc}_{\widetilde{\mathsf{PK}}}(\mathsf{CT}_a)$. Equation (7) holds for any $\zeta_{x'}$ including ζ_{CT_a}, hence $\mathsf{Eval}(1^\lambda, \rho, \sigma_1)$ will succeed with probability $1 - \varepsilon(x) - \mathsf{negl}(\lambda)$, i.e. it will output a QFHE encryption of b. Since ρ' is $\sqrt{\varepsilon}$-close to ρ in trace distance, it follows that $\mathsf{Eval}(1^\lambda, \rho', \mathsf{CT}_a)$ will succeed with probability $1 - \varepsilon(\lambda) - \sqrt{\varepsilon(\lambda)} - \mathsf{negl}(\lambda)$. This inequality together with correctness of $\widetilde{\mathsf{QFHE}}$ and correctness of QFHE homomorphic evaluation imply that $\Pr[E_3 \mid E_2] \geq 1 - \varepsilon(\lambda) - \sqrt{\varepsilon(\lambda)} - \mathsf{negl}(\lambda) \geq 1 - 2\sqrt{\varepsilon(\lambda)}$, where E_3 is the event that $\mathsf{CT}^*_b \in \mathsf{QFHE}.\mathsf{Enc}_{\mathsf{PK}}(b)$. Conditioned on E_1, E_2, E_3, the adversary can recover the true classical description with probability $1 - \mathsf{negl}(\lambda)$ by correctness of QFHE and Obf. Thus, $\mathcal{A}'$ succeeds in recovering the classical description of G_C wiith probability at least

$$\begin{aligned}
&\Pr[E_3] - \mathsf{negl}(\lambda) \geq \Pr[E_3 \mid E_2] \cdot \Pr[E_2 \mid E_1] \cdot \Pr[E_1] - \mathsf{negl}(\lambda) \geq \\
&\geq (1 - \varepsilon(\lambda) - \mathsf{negl}(\lambda))\,(1 - \mathsf{negl}(\lambda))(1 - 2\sqrt{\varepsilon(\lambda)}) - \mathsf{negl}(\lambda) \\
&\geq 1 - 3\sqrt{\varepsilon(\lambda)},
\end{aligned}$$

which suffices for the proof.

□

□

4.2 Impossibility of Copy-Protection in CAROM

Assuming impossibility of approximate copy-protection in the plain model, we will show that it is impossible to have a copy-protection scheme in CAROM which satisfies $(1 - \mathsf{negl}(\lambda))$-correctness and $\left(1 - \frac{1}{\mathrm{poly}(\lambda)}\right)$-security. The proof will follow closely the proof in [CKP15], which shows impossibility of classical virtual-black-box obfuscation in the random oracle model assuming its impossibility in the plain model.

Lemma 21. *Let $\delta = \delta(\lambda)$ be a function and $\mathcal{F}$ be a class of functions of the form $f : X \to Y$. Let $\mathcal{D}_\mathcal{F}, \mathcal{D}_X$ be distributions over $\mathcal{F}, X$, respectively, and $\mathfrak{D}_X = \{\widetilde{\mathcal{D}_X}^f\}_{f \in \mathcal{F}}$ be a class of distributions over X. Assume that $\mathcal{D}_X$ can be efficiently sampled. Suppose there exists a reusable copy-protection scheme $(\mathsf{CP}^\mathcal{O}, \mathsf{Eval}^\mathcal{O})$ in CAROM for $\mathcal{F}$ with $(\mathcal{D}_X, 1 - \mathsf{negl}(\lambda))$-mean-correctness and $(\mathcal{D}_\mathcal{F}, \mathfrak{D}_X, \delta)$-security. Then, for any noticeable function ε there exists a copy-protection scheme $(\widetilde{\mathsf{Eval}}, \widetilde{\mathsf{CP}})$ in the plain model with $(\mathcal{D}_X, 1 - \varepsilon(\lambda))$-mean-correctness and $(\mathcal{D}_\mathcal{F}, \mathfrak{D}_X, \delta)$-security.*

Combining Lemma 21 with Theorem 17, we have the following.

Theorem 22 (Main Theorem)**.** *There exists an unlearnable class of circuits $\mathcal{G} = \mathcal{G}(\lambda)$ and an input distribution $\mathcal{D}_X$ such that there does not exist a reusable copy-protection scheme in CAROM with $(\mathcal{D}_X, 1 - \mathsf{negl}(\lambda))$-mean-correctness and $(\mathcal{D}_\mathcal{F}, \mathfrak{D}_X, 1 - \varepsilon(\lambda))$-security for any noticeable function ε and any distributions $\mathcal{D}_\mathcal{F}, \mathfrak{D}_X$.*

Proof of Lemma 21. Let[10] $(\mathsf{CP}^\mathcal{O}, \mathsf{Eval}^\mathcal{O})$ be given as in the lemma statement. Since $\mathsf{CP}^\mathcal{O}$ and $\mathsf{Eval}^\mathcal{O}$ both run in polynomial time, the numbers of queries they make to $\mathcal{O}$ are bounded by some $M = \text{poly}(\lambda)$ and $N = \text{poly}(\lambda)$, respectively. We will construct a valid copy-protection scheme $(\widetilde{\mathsf{CP}}, \widetilde{\mathsf{Eval}})$ for $\mathcal{F}$ in the plain model. Let $T := \lceil \frac{2M}{\varepsilon} \rceil$.

Copy Protection: $\widetilde{\mathsf{CP}}(1^\lambda, d_f)$ takes as input the security parameter 1^λ and the classical description d_f of a function $f \in \mathcal{F}$, and it does the following:

1. Simulate *on-the-fly* a classical random oracle $\mathcal{O}$ to be used in any of the steps below.
2. Run $\rho_f \leftarrow \mathsf{CP}^\mathcal{O}(1^\lambda, d_f)$.
3. Set $\rho_f^1 := \rho_f$. Pick $S \xleftarrow{\$} \{0, 1, \ldots, T-1\}$. For $i = 1$ to S:
   (a) Sample $x_i \leftarrow \mathcal{D}_X$.
   (b) Compute $\rho_f^{i+1} \otimes |y_i\rangle\langle y_i| \leftarrow \mathsf{Eval}^\mathcal{O}(1^\lambda, \rho_f^i, x_i)$ and record the queries made by Eval with their answers in a database D^i_{Eval}.
4. Sample random oracle answers $r_1, r_2, \ldots, r_N \xleftarrow{\$} Y$. These will be used by $\widetilde{\mathsf{Eval}}$ to answer queries not recorded in any of the databases D^i_{Eval}.
5. Output the state $\widetilde{\rho_f} := \rho_f^{S+1} \otimes |D_{\mathsf{Eval}}\rangle\langle D_{\mathsf{Eval}}| \bigotimes_{j=1}^N |r_j\rangle\langle r_j|$, where $D_{\mathsf{Eval}} := \bigcup_{i=1}^S D^i_{\mathsf{Eval}}$.

Evaluation: $\widetilde{\mathsf{Eval}}(1^\lambda, \widetilde{\rho_f}, x)$ takes as input the security parameter 1^λ, a copy-protected state $\widetilde{\rho}$, and an input $x \in X$, and it does the following:

1. Parse the state $\widetilde{\rho_f}$ as $\rho_f \otimes |D\rangle\langle D| \bigotimes_{j=1}^N |r_j\rangle\langle r_j|$ by measuring the registers corresponding to the database D and the oracle answers $\{r_j\}_{j=1}^N$.
2. Run $\rho'_f \otimes |y\rangle\langle y| \leftarrow \mathsf{Eval}^\mathcal{O}(1^\lambda, \rho_f, x)$, answering the oracle queries of Eval as follows: to answer the jth query made by Eval, answer consistently if the query is in D, and answer using r_j if the query is not in D (without loss of generality Eval does not make repeated queries). Output the state $\left(\rho'_f \otimes |D\rangle\langle D| \bigotimes_{j=1}^N |r_j\rangle\langle r_j|\right) \otimes |y\rangle\langle y|$.

Approximate Correctness: Fix $f \in \mathcal{F}$. By $(\mathcal{D}_X, 1 - \mathsf{negl}(\lambda))$-mean-correctness and reusability of $(\mathsf{CP}^\mathcal{O}, \mathsf{Eval}^\mathcal{O})$, we have

$$\Pr\left[f(x_{S+1}) \leftarrow \mathsf{Eval}^\mathcal{O}(1^\lambda, \rho_f^{S+1}, x_{S+1}) \ : \ \substack{\rho_f^1 \leftarrow \mathsf{CP}(1^\lambda, d_f) \\ x_i \leftarrow \mathcal{D}_X,\ 1 \le i \le S+1 \\ \rho_f^{i+1} \leftarrow \mathsf{Eval}^\mathcal{O}(1^\lambda, \rho_f^i, x_i),\ 1 \le i \le S} \right] \geq 1 - \mathsf{negl}(\lambda). \tag{8}$$

[10] In [CKP15], the number of test executions (T) has an extra factor of N. Our modified analysis could also be applied to their construction to get rid of the factor of N.

Recall that $\widetilde{\mathsf{Eval}}$ emulates $\mathsf{Eval}^{\mathcal{O}}$, using the database $D = D_{\mathsf{Eval}}$ to answer oracle queries and using the independent random answers r_j when the query is not in D. This is equivalent to running $\mathsf{Eval}^{\mathcal{O}_D}(1^\lambda, \rho_f, x)$, where $\mathcal{O}_D$ is defined as a random oracle conditioned to be consistent with D. Let D_{CP} be the set of queries made by CP in step 2 of $\widetilde{\mathsf{CP}}$. Let E be the event that $\mathsf{Eval}(1^\lambda, \rho_f, x)$ does not make any oracle query in $D_{\mathsf{CP}} \setminus D_{\mathsf{Eval}}$ during the execution of $\widetilde{\mathsf{Eval}}$.[11] Keep in mind that $\widetilde{\mathsf{Eval}}$ emulates $\mathsf{Eval}^{\mathcal{O}}$ at the time of the $(S+1)$st test execution. Hence, E is equivalently the event that $\mathsf{Eval}(1^\lambda, \rho_f^{S+1}, x_{S+1})$ makes no query in $D_{\mathsf{CP}} \setminus D_{\mathsf{Eval}}$, where ρ_f^{S+1} and x_{S+1} are as in eq. (8). Conditioned on E, the emulation of $\widetilde{\mathsf{Eval}}$ is flawless, i.e. $\mathsf{Eval}^{\mathcal{O}}(1^\lambda, \rho_f, x)$ and $\mathsf{Eval}^{\mathcal{O}_D}(1^\lambda, \rho_f, x)$ are perfectly indistinguishable. Hence, we can lower-bound the correctness of $\widetilde{\mathsf{Eval}}$ by lower-bounding the probability of E:

$$
\begin{aligned}
&\mathsf{Pr}\left[f(x) \leftarrow \widetilde{\mathsf{Eval}}(1^\lambda, \widetilde{\rho_f}, x) \ : \ \substack{\widetilde{\rho_f} \leftarrow \widetilde{\mathsf{CP}}(1^\lambda, d_f) \\ x \leftarrow \mathcal{D}_X}\right] \\
&= 1 - \mathsf{Pr}\left[y \neq f(x) \ : \ \substack{\widetilde{\rho_f} \leftarrow \widetilde{\mathsf{CP}}(1^\lambda, d_f) \\ x \leftarrow \mathcal{D}_X \\ y \leftarrow \widetilde{\mathsf{Eval}}(1^\lambda, \widetilde{\rho_f}, x)}\right] \\
&= 1 - \mathsf{Pr}\left[y \neq f(x_{S+1}) \ : \ \substack{\rho_f \leftarrow \mathsf{CP}^{\mathcal{O}}(1^\lambda, d_f) \\ S \xleftarrow{\$} \{0,1,\dots,T-1\} \\ x_i \leftarrow \mathcal{D}_X,\ 1 \leq i \leq S+1 \\ \rho_f^{i+1} \leftarrow \mathsf{Eval}^{\mathcal{O}}(1^\lambda, \rho_f^i, x_i),\ 1 \leq i \leq S \\ y \leftarrow \mathsf{Eval}^{\mathcal{O}_D}(1^\lambda, \rho_f^{S+1}, x_{S+1})}\right] \\
&\geq 1 - \left(\mathsf{Pr}\left[E\right] \cdot \mathsf{Pr}\left[y \neq f(x_{S+1}) \ : \ \substack{\rho_f \leftarrow \mathsf{CP}^{\mathcal{O}}(1^\lambda, d_f) \\ S \xleftarrow{\$} \{0,1,\dots,T-1\} \\ x_i \leftarrow \mathcal{D}_X,\ 1 \leq i \leq S+1 \\ \rho_f^{i+1} \leftarrow \mathsf{Eval}^{\mathcal{O}}(1^\lambda, \rho_f^i, x_i),\ 1 \leq i \leq S \\ y \leftarrow \mathsf{Eval}^{\mathcal{O}_D}(1^\lambda, \rho_f^{S+1}, x_{S+1})} \ \middle| \ E\right] + \mathsf{Pr}\left[\neg E\right]\right) \\
&\overset{(8)}{\geq} 1 - \mathsf{Pr}\left[\neg E\right] - \mathsf{negl}(\lambda).
\end{aligned}
\tag{9}
$$

Therefore, the following claim will suffice for the proof together with eq. (9):

Claim 23. $\mathsf{Pr}\left[\neg E\right] \leq \frac{\varepsilon(\lambda)}{2}$.

Proof. Consider the following experiment, which consists of T executions of $\mathsf{Eval}^{\mathcal{O}}$ with random inputs:

1. Let $\mathcal{O}$ be a random oracle.
2. Compute $\rho_f \leftarrow \mathsf{CP}^{\mathcal{O}}(1^\lambda, d_f)$.
3. Set $\rho_f^1 := \rho_f$. For $i = 1$ to T:
   (a) Sample $x_i \leftarrow \mathcal{D}_X$.
   (b) Compute $\rho_f^{i+1} \otimes |y_i\rangle\langle y_i| \leftarrow \mathsf{Eval}^{\mathcal{O}}(1^\lambda, \rho_f^i, x_i)$. Let D_{Eval}^i be the set of inputs queried by Eval in this step.

Let $w_i = \left|\left(D_{\mathsf{Eval}}^i \cap D_{\mathsf{CP}}\right) \setminus \bigcup_{1 \leq j < i} D_{\mathsf{Eval}}^j\right|$ be the random variable corresponding to the number of *new* queries from D_{CP} made by $\mathsf{Eval}^{\mathcal{O}}$ at the ith step above. Let $p_i = \mathsf{Pr}\left[w_i \geq 1\right]$ be the probability

[11] Here by a slight abuse of notation we only consider the input $x \in X$ of the query and not the answer $y \in Y$.

that a new query is made at the ith step. Recall that $\widetilde{\mathsf{Eval}}$ chooses the number of test executions uniformly at random, hence $\mathsf{Pr}\left[\neg E\right] = \frac{1}{T}\sum_{i=1}^{T} p_i$ is the probability that $w_i \geq 1$ for a random $i \in [T]$. Now, by linearity of expectation we have

$$\sum_{i=1}^{T} p_i \leq \sum_{i=1}^{T} \mathbb{E}\left[w_i\right] = \mathbb{E}\left[\sum_{i=1}^{T} w_i\right] = \mathbb{E}\left[\left| D_{\mathsf{CP}} \cap \left(\bigcup_{j=1}^{T} D_{\mathsf{Eval}}^{j}\right)\right|\right] \leq |D_{\mathsf{CP}}| \leq M.$$

Therefore, $\mathsf{Pr}\left[\neg E\right] = \frac{1}{T}\sum_{i=1}^{T} p_i \leq \frac{M}{T} \leq \frac{\varepsilon}{2}$ as desired. □

Copy-Protection Security: Suppose there is an adversary $\left(\widetilde{\mathcal{A}}, \widetilde{\mathcal{B}}, \widetilde{\mathcal{C}}\right)$ which succeeds in the pirating experiment for $(\widetilde{\mathsf{CP}}, \widetilde{\mathsf{Eval}})$ with probability ε, i.e.

$$\mathsf{Pr}\left[b = 1 \ : \ b \leftarrow \mathsf{PirExp}_{\mathcal{D}_{\mathcal{F}}, \mathfrak{D}_X}^{\widetilde{\mathsf{CP}}, \widetilde{\mathsf{Eval}}}\left(1^{\lambda}, \left(\widetilde{\mathcal{A}}, \widetilde{\mathcal{B}}, \widetilde{\mathcal{C}}\right)\right)\right] = \varepsilon.$$

Using $\left(\widetilde{\mathcal{A}}, \widetilde{\mathcal{B}}, \widetilde{\mathcal{C}}\right)$, we will construct an adversary $\left(\mathcal{A}^{\mathcal{O}}, \mathcal{B}^{\mathcal{O}}, \mathcal{C}^{\mathcal{O}}\right)$ which succeeds in the pirating experiment for $(\mathsf{CP}, \mathsf{Eval})$ with probability ε. This will immediately imply δ-security of $(\widetilde{\mathsf{CP}}, \widetilde{\mathsf{Eval}})$ by the δ-security of $(\mathsf{CP}, \mathsf{Eval})$.

We set $\mathcal{B}^{\mathcal{O}}$ and $\mathcal{C}^{\mathcal{O}}$ to be identical to $\widetilde{\mathcal{B}}$ and $\widetilde{\mathcal{C}}$, respectively, so that they do not make any queries. We define $\mathcal{A}^{\mathcal{O}}$, which has oracle access to $\mathcal{O}$, as follows:

1. Given a copy-protected program $\rho_f =: \rho_f^1$, pick $S \xleftarrow{\$} \{0, 1, \ldots, T-1\}$ and repeat for $i = 1$ to S:
   (a) Sample $x_i \leftarrow \mathcal{D}_X$
   (b) Run $\rho_f^{i+1} \otimes |y_i\rangle\langle y_i| \leftarrow \mathsf{Eval}^{\mathcal{O}}(1^{\lambda}, \rho_f^i, x_i)$, forwarding the oracle queries of Eval to $\mathcal{O}$ and recording them with their answers in a database D_{Eval}^i.
2. Sample random oracle answers $r_1, r_2, \ldots, r_N \xleftarrow{\$} Y$.
3. Set $\widetilde{\rho_f} := \rho_f^{S+1} \otimes |D_{\mathsf{Eval}}\rangle\langle D_{\mathsf{Eval}}| \bigotimes_{j=1}^{N} |r_j\rangle\langle r_j|$, where $D_{\mathsf{Eval}} := \bigcup_{i=1}^{S} D_{\mathsf{Eval}}^i$.
4. Run $\widetilde{\mathcal{A}}$ on the state $\widetilde{\rho_f}$.

By construction, the bipartite state received by $\mathcal{B}^{\mathcal{O}}$ and $\mathcal{C}^{\mathcal{O}}$ is identical to that received by $\mathcal{B}$ and $\mathcal{C}$. This is because the on-the-fly simulation done by $\widetilde{\mathsf{CP}}$ is perfectly indistinguishable from the real oracle answers provided by $\mathcal{O}$. Therefore,

$$\begin{aligned}&\mathsf{Pr}\left[b = 1 \ : \ b \leftarrow \mathsf{PirExp}_{\mathcal{D}_{\mathcal{F}}, \mathfrak{D}_X}^{\mathsf{CP}, \mathsf{Eval}}\left(1^{\lambda}, (\mathcal{A}, \mathcal{B}, \mathcal{C})\right)\right] \\ =& \mathsf{Pr}\left[b = 1 \ : \ b \leftarrow \mathsf{PirExp}_{\mathcal{D}_{\mathcal{F}}, \mathfrak{D}_X}^{\widetilde{\mathsf{CP}}, \widetilde{\mathsf{Eval}}}\left(1^{\lambda}, \left(\widetilde{\mathcal{A}}, \widetilde{\mathcal{B}}, \widetilde{\mathcal{C}}\right)\right)\right] = \varepsilon\end{aligned}$$

as desired.

□

References

[Aar04] Scott Aaronson. Limitations of quantum advice and one-way communication. In *Proceedings of the 19th IEEE Annual Conference on Computational Complexity*, CCC '04, page 320–332, USA, 2004. IEEE Computer Society.

[Aar09] Scott Aaronson. Quantum copy-protection and quantum money. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 229–242. IEEE, 2009.

[ABDS21] Gorjan Alagic, Zvika Brakerski, Yfke Dulek, and Christian Schaffner. Impossibility of quantum virtual black-box obfuscation of classical circuits. In *CRYPTO*, 2021.

[AC12] Scott Aaronson and Paul Christiano. Quantum money from hidden subspaces. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 41–60, 2012.

[AGKZ20] Ryan Amos, Marios Georgiou, Aggelos Kiayias, and Mark Zhandry. One-shot signatures and applications to hybrid quantum/classical authentication. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 255–268, 2020.

[AK21] Prabhanjan Ananth and Fatih Kaleoglu. Uncloneable encryption, revisited. In *TCC*, 2021.

[AKL$^+$22] Prabhanjan Ananth, Fatih Kaleoglu, Xingjian Li, Qipeng Liu, and Mark Zhandry. On the feasibility of unclonable encryption, and more. In *CRYPTO*, 2022.

[ALL$^+$21] Scott Aaronson, Jiahui Liu, Qipeng Liu, Mark Zhandry, and Ruizhe Zhang. New approaches for quantum copy-protection. In *CRYPTO*, 2021.

[ALP21] Prabhanjan Ananth and Rolando L La Placa. Secure software leasing. In *Eurocrypt*, 2021.

[BDF$^+$11] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In *International conference on the theory and application of cryptology and information security*, pages 41–69. Springer, 2011.

[BI20] Anne Broadbent and Rabib Islam. Quantum encryption with certified deletion. In *Theory of Cryptography Conference*, pages 92–122. Springer, 2020.

[BJ15] Anne Broadbent and Stacey Jeffery. Quantum homomorphic encryption for circuits of low t-gate complexity. In *Annual Cryptology Conference*, pages 609–629. Springer, 2015.

[BJL$^+$21] Anne Broadbent, Stacey Jeffery, Sébastien Lord, Supartha Podder, and Aarthi Sundaram. Secure software leasing without assumptions. In *TCC*, 2021.

[BL20] Anne Broadbent and Sébastien Lord. Uncloneable quantum encryption via oracles. In *TQC*, 2020.

[Bra18] Zvika Brakerski. Quantum fhe (almost) as secure as classical. In *Annual International Cryptology Conference*, pages 67–95. Springer, 2018.

[BV16] Nir Bitansky and Vinod Vaikuntanathan. Indistinguishability obfuscation: from approximate to exact. In *Theory of Cryptography Conference*, pages 67–95. Springer, 2016.

[CDM20] Orestis Chardouvelis, Nico Doettling, and Giulio Malavolta. Rate-1 secure function evaluation for bqp. Cryptology ePrint Archive, Report 2020/1454, 2020. https://ia.cr/2020/1454.

[CKP15] Ran Canetti, Yael Tauman Kalai, and Omer Paneth. On obfuscation with random oracles. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *Theory of Cryptography*, pages 456–467, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[CLLZ21] Andrea Coladangelo, Jiahui Liu, Qipeng Liu, and Mark Zhandry. Hidden cosets and applications to unclonable cryptography. In *CRYPTO*, 2021.

[CMP20] Andrea Coladangelo, Christian Majenz, and Alexander Poremba. Quantum copy-protection of compute-and-compare programs in the quantum random oracle model. *arXiv preprint arXiv:2009.13865*, 2020.

[Die82] DGBJ Dieks. Communication by epr devices. *Physics Letters A*, 92(6):271–272, 1982.

[GKVW19] Rishab Goyal, Venkata Koppula, Satyanarayana Vusirikala, and Brent Waters. On perfect correctness in (lockable) obfuscation. 2019.

[GKW17] Rishab Goyal, Venkata Koppula, and Brent Waters. Lockable obfuscation. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 612–621. IEEE, 2017.

[Got02] Daniel Gottesman. Uncloneable encryption. *arXiv preprint quant-ph/0210062*, 2002.

[GZ20] Marios Georgiou and Mark Zhandry. Unclonable decryption keys. *IACR Cryptol. ePrint Arch*, 877(2020):3, 2020.

[KNY21] Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Secure software leasing from standard assumptions. In *TCC*, 2021.

[Mah18] Urmila Mahadev. Classical homomorphic encryption for quantum circuits. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 332–338. IEEE, 2018.

[RS19] Roy Radian and Sattath. Semi-quantum money. *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, Oct 2019.

[Wie83] Stephen Wiesner. Conjugate coding. *ACM Sigact News*, 15(1):78–88, 1983.

[Win99] A. Winter. Coding theorem and strong converse for quantum channels. *IEEE Transactions on Information Theory*, 45(7):2481–2485, 1999.

[WZ82] William K Wootters and Wojciech H Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982.

[WZ17] Daniel Wichs and Giorgos Zirdelis. Obfuscating compute-and-compare programs under lwe. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 600–611. IEEE, 2017.

[Zha19a] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 239–268, Cham, 2019. Springer International Publishing.

[Zha19b] Mark Zhandry. Quantum lightning never strikes the same state twice. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 408–438. Springer, 2019.