On quantum obfuscation

Gorjan Alagic and Bill Fefferman

January 30, 2016

Abstract

Encryption of data is fundamental to secure communication in the modern world. Beyond encryption of data lies *obfuscation*, i.e., encryption of functionality. It is well-known that the most powerful means of obfuscating classical programs, so-called "black-box obfuscation," is provably impossible [6]. For years since, obfuscation was believed to always be either impossible or useless, depending on the particulars of its formal definition. However, several recent results have yielded candidate schemes that satisfy a definition weaker than black-box, and yet still have numerous applications.

In this work, we initialize the rigorous study of obfuscating programs *via quantum-mechanical means*. We define notions of quantum obfuscation which encompass several natural variants. For instance, the input can describe classical or quantum functionality, and the output can be either a classical description or a quantum state. The obfuscator can also satisfy one of a number of obfuscation conditions: black-box, information-theoretic black-box, indistinguishability, and best-possible. We discuss a number of applications, including CPA-secure quantum encryption, quantum fully-homomorphic encryption, and quantum money. We then prove several impossibility results, extending a number of foundational papers on classical obfuscation to the quantum setting. We prove that quantum black-box obfuscation is impossible in a setting where adversaries can possess more than one output of the obfuscator (possibly even on the same input.) In particular, generic transformation of quantum circuits into black-box-obfuscated quantum circuits is impossible. We also show that statistical indistinguishability obfuscation is impossible, up to an unlikely complexity-theoretic collapse. Our proofs involve a new tool: chosen-ciphertext-secure encryption of quantum data, which is possible provided that quantum-secure one-way functions exist.

We emphasize that our results leave open one intriguing possibility: obfuscating a classical or quantum circuit into a single, uncloneable quantum state. This indicates that, in spite of our results, quantum obfuscation may be significantly more powerful than its classical counterpart.

Contents

1 Introduction

The ability to encrypt data is central to modern communications. Basic methods for performing this task with privately-exchanged encryption keys have been known for hundreds of years. More advanced, public-key encryption methods were developed much more recently, beginning with the work of Merkle [29] and Diffie and Hellman [16] in the 1970s. These public-key methods have found widespread practical application in virtually all Internet communications. More advanced theoretical methods for encrypting data, such as fully-homomorphic encryption, have only been discovered recently [21], but show great promise for practical application.

Arguably the most powerful encryption ability is *obfuscation*; this is the ability to *encrypt functionality*. Obfuscation implies (with some caveats) the ability to perform almost any cryptographic task imaginable, including public-key and fully-homomorphic encryption. Unlike in the case of data encryption, our theoretical understanding of obfuscation is still fairly limited.

To understand obfuscation, it is useful to think about an obvious application: protecting intellectual property in software. In this setting, a software developer wishes to distribute their software to end users. However, the code contains a number of trade secrets which the developer does not want to become public. In order to maintain these secrets, the publisher passes the software through an *obfuscation algorithm* (or obfuscator) prior to publishing. In this application, the obfuscator must be an efficient algorithm that satisfies three core properties:

1. *functional equivalence:* the input/output functionality of the input program does not change;
2. *polynomial slowdown:* if the input program is efficient, then the output program is efficient;
3. *obfuscation:* the code of the output program is "hard to understand."

The last condition can be formulated rigorously in a number of ways. One possibility is the so-called "virtual black-box" condition, which says that the obfuscated program is no more useful than an impenetrable box which simply accepts inputs and produces outputs. While this condition appears to be too strong in the classical world, there are other formulations (with varying levels of strength and usefulness) which may be achievable.

The study of encrypting classical data and classical programs is significantly complicated by the advent of *quantum computation.* One well-known consequence of the presence of quantum computers is that certain data encryption schemes, such as those based on the hardness of factoring or the discrete logarithm, are no longer secure. It is conceivable that certain classical obfuscation schemes are also not secure against quantum adversaries. On the other hand, quantum mechanics also appears to enable certain cryptographic tasks (such as information-theoretically secure key exchange) that are impossible classically. It is thus natural to ask what quantum computation means for obfuscation of programs. In particular, we would like to answer the following questions:

- what are some natural formulations of quantum-mechanical program obfuscation?
- is it possible to quantumly obfuscate classical and/or quantum programs?
- which of the classical results about obfuscation carry over to the quantum setting?
- are there applications of quantum obfuscation that are impossible classically?

We remark that, in order to address the above questions, we must also properly address the question of encrypting quantum data—a strictly simpler task than encrypting functionality. While information-theoretic encryption of quantum data has been considered before, in this setting we are interested in encryption of quantum data *with computational assumptions*[1]. This latter subject has not yet received significant attention in literature.

Before continuing, we draw attention to the distinction between obfuscating *programs* and obfuscating *circuits*. While these two forms of obfuscation are closely related, there are some important technical

[1] Note that information-theoretic obfuscation is impossible if the adversary can execute the obfuscated program on all possible inputs; indeed, a computationally unbounded adversary can use this ability to learn everything there is to know about the program.

differences. In this work, as in most theoretical works on obfuscation, we will focus on obfuscation of circuits. We view the circuit model as more convenient; it also tends to be preferred in the theoretical literature on both cryptography and quantum computation.

1.1 Background

We now briefly review the current state of affairs in research on obfuscation. The classical case has been studied significantly. Quantum obfuscation, on the other hand, has received little to no attention.

1.1.1 Classical obfuscation

Ad-hoc obfuscation of software has been a fairly common practice for some time. In fact, simply compiling a program can be viewed as a form of obfuscation. The earliest mention of obfuscation in the modern study of theoretical cryptography appears to be in the famous paper of Diffie and Hellman [16]. There, it was suggested that public-key cryptosystems might be constructible via obfuscation of private-key schemes; this was viewed as a reasonable possibility because writing code in an obfuscated manner seems relatively easy in practice.

The first major result in classical obfuscation was the 2001 proof by Barak et al. that virtual black-box obfuscation is impossible [6, 7]. Their definition is based on the *simulation paradigm*. More precisely, the obfuscation condition (i.e., the third condition in the previous section) states that any efficient algorithm with access to an obfuscated circuit should be simulable by another efficient algorithm with only oracle (i.e., black-box) access to the original functionality. This definition is very natural in the setting of the aforementioned "software intellectual property protection" application: the end user can only learn that which is learnable by simply running the program. Barak et al. proved that there exist circuit families which are unobfuscatable under this definition. They also showed that some of the most sought-after applications of black-box obfuscation are impossible. For instance, they showed that private-key encryption schemes cannot be transformed to public-key ones by obfuscating the encryption circuits in a generic manner.

The years following the Barak et al. result saw some limited progress in theoretical obfuscation. It was proved possible for some limited forms of functionality [15, 34], and some additional limits were placed, e.g., on black-box obfuscation with auxiliary input [23]. An important step in formulating feasible notions of obfuscation was taken by Goldwasser and Rothblum; they defined *indistinguishability obfuscation* and *best-possible obfuscation* [24]. Both of these definitions alter the obfuscation condition, while leaving the functional-equivalence and polynomial-slowdown conditions unchanged. Under indistinguishability, it is required that the obfuscator maps functionally-equivalent circuits to indistinguishable distributions. Under best-possible, the obfuscator maps any circuit to a circuit from which the end user can "learn the least." Both definitions have a perfect, statistical, and computational variant. Goldwasser and Rothblum proved that the two definitions are equivalent, and that the perfect and statistical versions are impossible (unless the PH collapses) [24]. This left one possibility: computational indistinguishability obfuscation. It was widely believed that computational indistinguishability was too weak of a condition to provide any interesting applications.

In 2013, in a breakthrough result, Garg et al. proposed a convincing candidate for computational indistinguishability obfuscation [18]. They proposed an obfuscation scheme for NC1 circuits, based on the presumed hardness of a problem in multilinear maps; they also showed how to use fully-homomorphic encryption (with NC1 decryption circuits) to "bootstrap" their NC1 scheme to obfuscation for all circuits. Around the same time, another breakthrough by Sahai and Waters showed how to use a computational indistinguishability obfuscator to achieve a wide-range of applications, via a new "punctured programs" technique [31]. These applications include chosen-ciphertext-secure public-key encryption, injective trapdoor functions, and oblivious transfer. Sahai and Waters suggested that the applications were so wide-ranging that indistinguishability obfuscation might become a "'central hub' for cryptography" [31]. These two breakthroughs were followed by a flurry of new activity in the area, including several new proposals and applications [8, 11, 12, 13, 20, 27].

1.1.2 Quantum obfuscation

Quantum obfuscation is essentially an unexplored topic, and the present work appears to be the first rigorous treatment of the foundational questions. The question of whether quantum obfuscation is possible was posed as one of Scott Aaronson's "semi-grand challenges" for quantum computation [1]. Since so little work on quantum obfuscation has appeared, our brief discussion will also mention some results that we believe are related.

In [2], Aaronson proposed two relevant results. The first was a *complexity-theoretic no-cloning theorem*, stating that cloning an unknown, random state by means of a black-box "reflection oracle" requires exponentially many queries. The second theorem stated that an oracle exists relative to which "software copy-protection" is possible. Unfortunately, a full version of [2] with proofs never appeared, although the complexity-theoretic no-cloning theorem was eventually proved in a paper on quantum money [3]. In related work, Mosca and Stebila proposed a black-box quantum money scheme, and suggested the possibility of using a quantum circuit obfuscator in place of the black box [30].

More recently, Alagic, Jeffery and Jordan proposed obfuscators for both classical (reversible) circuits and quantum circuits, based on ideas from topological quantum computation [4]. The proposed obfuscator compiles the circuits into braids using certain high-dimensional representations of the braid group, and then applies an algorithm for putting braids into normal form. Although it is efficient, this algorithm does not satisfy any of the aforementioned obfuscation definitions; instead, it satisfies perfect indistinguishability for a restricted set of circuit equivalences. The usefulness of such an obfuscator is unclear at this time.

1.2 Summary of results

In this section, we summarize our results and discussions. These are divided by subject, with quantum encryption covered in Section 2, quantum black-box obfuscation in Section 3, and quantum indistinguishability obfuscation in Section 5.

1.2.1 Quantum encryption

For us, *quantum encryption* will mean the encryption of quantum states under computational assumptions. In this work, the crucial advantages of this form of quantum encryption over its information-theoretic analogues (e.g., the quantum one-time pad) are (i.) reusability of the key, and (ii.) chosen-ciphertext security. The results on quantum encryption which we will present are summarized below, and will be necessary in order to establish some of our results about black-box obfuscation. A complete treatment will appear in [5].

1. **Quantum encryption schemes.** We define a notion of symmetric-key encryption scheme for quantum states, with reusable keys; these schemes consist of three quantum algorithms (key generation, encryption, and decryption) which satisfy correctness: under a fixed key, encryption followed by decryption must be equivalent to the identity.
2. **Chosen-ciphertext security for quantum encryption.** We define a notion of IND-CCA1 (or *indistinguishability of ciphertexts under non-adaptive chosen ciphertext attacks*) for these schemes; this formalizes the idea of a "lunchtime attack," where an adversary has complete access to all aspects of the encryption except the key itself, and is tasked with decrypting a challenge ciphertext later (presumably after lunch.)
3. **An IND-CCA1-secure construction.** We give a construction for an IND-CCA1-secure symmetric-key encryption scheme for quantum states, under the assumption that quantum-secure one-way functions (qOWF) exist. These qOWFs are deterministic classical functions which are easy to compute, but hard to invert for quantum adversaries.

We remark that, in contemporaneous work, Broadbent and Jeffrey also considered IND-CPA-secure public-key and symmetric-key quantum encryption; in addition, they considered partial quantum-homomorphic encryption [14].

1.2.2 Quantum black-box obfuscation

Definitions. Our main results concern definitions, applications, and (im)possibility of quantum obfuscation in the virtual black-box setting. We will begin by defining the following.

1. **Quantum black-box obfuscator.** This is a polynomial-time quantum algorithm $\mathcal{O}$ which accepts quantum circuits C as input, and produces quantum states $\mathcal{O}(C)$ as output. It preserves functionality, in the sense that there is a publicly known way to use $\mathcal{O}(C)$ and any input state $|\psi\rangle$ to produce the state $C|\psi\rangle$. It satisfies a black-box condition, which states that for polynomial-time quantum algorithms, possession of $\mathcal{O}(C)$ can be simulated by black-box access to C. This definition is a natural analogue of the classical black-box definition given in [7].
2. **Quantum "two-circuit" black-box obfuscator.** This obfuscator is precisely as above, except the obfuscation condition is strengthened to hold over arbitrary *pairs of circuits* (C_1, C_2). For us, this definition will be primarily useful because of its role in establishing certain impossibility results.
3. **Information-theoretic quantum black-box obfuscator.** This is a modification of the above definition, in which we posit that *any* adversary with access to $\mathcal{O}(C)$ can be simulated by a *polynomial-time* quantum simulator with black-box access to C. This definition is impossible classically, for obvious reasons: both $\mathcal{O}(C)$ and $\mathcal{O}$ can be copied and reused an arbitrary number of times, enabling unbounded adversaries to discover everything about C.

Impossibility. We prove three impossibility results, which place several important restrictions on quantum obfuscation. Our impossibility proofs are based on the ideas of Barak et al. [7], with several important quantum adaptations, and a new quantum ingredient: the aforementioned IND-CCA1 quantum encryption.

1. **Two-state black-box obfuscation is impossible.** We prove that there exist families of circuit pairs which can reveal a secret if one is in possession of a circuit description for both of them, but not if one only has black-box access. This impossibility persists even if the obfuscation output is a quantum state, as opposed to a circuit description. Unlike the other results, it is also true even if the obfuscated states are *not reusable.*
2. **If qOWFs exist, then obfuscation with more than one output is impossible.** For this proof, we combine the pairs from the circuit families in the two-circuit impossibility proof in order to build a single unobfuscatable family. The ability to execute obfuscated states from this family *on themselves* is crucial here, and has two requirements: (i.) access to more than one obfuscation, even if the obfuscations are quantum states, and (ii.) secure encryption, which in turn requires the existence of qOWFs. This result applies both to quantum black-box obfuscators (as in the first definition above) and the information-theoretic variant (as in the third definition above.)
3. **Classical algorithms for quantum obfuscation are impossible, unconditionally.** This result follows directly from the previous result and Application 1 below. It can be viewed as an extension of the original Barak et al. impossibility result to the case of quantum functionality and quantum adversaries.

Applications. We then move on to discuss potential applications of quantum black-box obfuscators. We emphasize that (with the exception of the first one), all of these applications are still possible *in some form* in spite of the above impossibility result. We view this as a strong indication that quantum obfuscation should be studied further. While some of the applications are analogues of known classical applications (as outlined in [7],) the last is special to the quantum setting. We are certain that many other quantum-specific applications are possible, given the combined advantage of obfuscation and no-cloning.

1. **Quantum-secure one-way functions.** We show that, if there exists a classical probabilistic algorithm for quantum obfuscation, then quantum-secure one-way functions exist. The above impossibility result rules this out, but the implication is nonetheless interesting; for one, it enables

the very proof of the impossibility result itself! The one-way functions are essentially the functions computed by the obfuscator (with fixed randomness) on circuits with a "hidden output." We are unable to extend this application to the setting of efficient quantum algorithms for obfuscation. We leave this as an interesting open problem, and note its connection to developing foundational primitives for quantum encryption.

2. **IND-CPA-secure private-key quantum encryption.** In this application, the obfuscation algorithm can be quantum; moreover, we do not demand the existence of one-way functions or any other primitive.
3. **qOWF imply IND-CPA public-key encryption.** This application combines IND-CCA1-secure private-key encryption (which follows from qOWFs) with obfuscation of the encryption circuits. The result is public-key encryption of quantum states without the need for trapdoor permutations (unlike in [5] and, indirectly, in [14].)
4. **qOWF imply IND-CPA quantum fully homomorphic encryption.** This application combines the previous application, together with obfuscation of a universal decrypt-compute-encrypt circuit. Depending on the properties of the obfuscator, it may also satisfy *compactness* (the requirement that communication between client and server does not scale with the size of the computation.)
5. **Public-key quantum money**. Using circuit obfuscation to produce public-key quantum money was first proposed by Mosca and Stebila [30], using a complexity-theoretic no-cloning theorem proposed by Aaronson [2] and proved by Aaronson and Christiano [3]. We outline the ideas here, and discuss the new limitations placed by our results.

We emphasize that all the above applications except quantum money also work for achieving *classical functionality* from a quantum obfuscator; however, depending on the details of the obfuscator and the application, this may require quantum algorithms for encryption and decryption, or even quantum ciphertexts.

1.2.3 Quantum indistinguishability obfuscation

Lastly, we consider an alternative formulation of obfuscation, motivated by the classical definitions of indistinguishability obfuscation and best-possible obfuscation, as set down by Barak et al. [7] and Goldwasser and Rothblum [24]. We establish quantum analogues of the central results in those classical papers. In this setting, rather than comparing the obfuscation of the circuit to that of a black-box, we compare it to the obfuscations of other, functionally-equivalent circuits. Starting with the new definitions, our results are as follows.

1. **Quantum indistinguishability obfuscator.** Just as in the black-box definition, this is a polynomial-time quantum algorithm $\mathcal{O}$ which accepts quantum circuits C as input, and produces "functionally-equivalent" quantum states $\mathcal{O}(C)$ as output. The obfuscation condition now states that functionally equivalent circuits are mapped to *indistinguishable* states. Based on the kind of indistinguishability deployed in the definition, there are three variants of an indistinguishability obfuscator: perfect, statistical, and computational.
2. **Quantum best-possible obfuscator.** This is an algorithm precisely as above, except for the obfuscation condition: it now states that $\mathcal{O}(C)$ is the state that "leaks least," among all states which are "functionally-equivalent" to C. There are again three variants: perfect, statistical, and computational.
3. **Equivalence of definitions.** We prove that each of the three variants of quantum indistinguishability obfuscation is equivalent to the analogous variant of quantum best-possible obfuscation, so long as the obfuscator is efficient.
4. **Impossibility of perfect and statistical indistinguishability obfuscation.** We end with a quantum version of the main result of [24]: a proof that perfect and statistical quantum indistinguishability

obfuscation is impossible, unless coQMA is contained in QSZK. We remark that an analogous containment in the classical setting (i.e., coMA $\subseteq$ SZK) would imply a collapse of the polynomial-time hierarchy to the second level. Moreover, for the case of obfuscating arbitrary quantum computations (i.e., completely positive, trace-preserving maps), we obtain that statistical quantum indistinguishability obfuscation would imply that PSPACE is contained in QSZK. One consequence of these results is that extending the obfuscator proposed in [4] to full indistinguishability is impossible, barring highly unlikely collapses of complexity classes.

5. **Application: witness encryption for QMA.** Motivated by an analogue discussed in [19, 18], we show that a quantum indistinguishability obfuscator enables witness encryption for QMA. A witness encryption scheme for a language L in QMA encrypts plaintexts x using a particular instance l. The security condition states that, if $l \in L$, then a valid witness w for $l \in L$ allows decryption; on the other hand, if $l \notin L$, then ciphertexts are indistinguishable. While witness encryption has several applications classically [19], the quantum analogue has not been considered previously.

We remark that, in the classical setting, indistinguishability obfuscation also implies functional encryption [18] and many more applications through the very successful "punctured programs" technique developed by Sahai and Waters [31]. We suspect that these results can also be adapted to the quantum setting, but leave them open for now.

1.3 Notation and terminology

In this section, we set down some notation and basic terminology which we will use throughout the rest of the paper.

We will assume that the state space of a classical device can be identified with sets of bitstrings, i.e., $\{0,1\}^n$ for some positive integer n. The notation $x \in_R \{0,1\}$ will mean that x is an n-bit string selected uniformly at random. The set of all bitstrings (of arbitrary length) will be denoted by $\{0,1\}^*$. Classical functions will then be maps $f : \{0,1\}^n \to \{0,1\}^m$ from one set of bitstrings to another. We will also sometimes consider function families, written $f : \{0,1\}^* \to \{0,1\}^*$; these can be thought of as a function family $\{f_n\}_{n>0}$ indexed by the input size n.

A classical circuit C is a sequence of local boolean gates which, when composed together, implement some (in general irreversible) function $f_C : \{0,1\}^n \to \{0,1\}^m$. The input size of C is n, the output size is m, and the number of gates is denoted by $|C|$. A probabilistic circuit is also a circuit, but with the input bits divided into two registers: the input register, and the "coin" register. A normal execution of a probabilistic circuit involves initializing the coin register with completely random bits, and inserting the input into the input register. We will frequently discuss ensembles of circuits; these are infinite families $\{C_n\}_{n>0}$ of circuits, one for each possible input size. We will also sometimes make use of *distributions of circuit ensembles*; these are infinite families $\mathcal{C} = \{\mathcal{C}_n\}_{n>0}$ where each $\mathcal{C}_n$ is a finite family of circuits of input size n, along with a probability distribution $P_{\mathcal{C},n}$. For a bitstring x, the notation $\mathcal{C}(x)$ will then denote the probability distribution (on bitstrings) resulting from running a random circuit from the family $\mathcal{C}_{|x|}$, selected according the distribution $P_{\mathcal{C},|x|}$.

A deterministic classical algorithm $\mathcal{A}$ is simply a circuit ensemble. Running $\mathcal{A}$ on an input bitstring x involves selecting the circuit with the appropriate input size, and executing it with input x. If the circuit ensemble is polynomial-time uniform (i.e., there exists a polynomial-time Turing machine that outputs descriptions of the circuits), we will say that $\mathcal{A}$ is efficient; more precisely, it is then a classical deterministic polynomial-time algorithm, or PT for short. A probabilistic algorithm $\mathcal{A}'$ is an algorithm whose circuits are probabilistic. Running $\mathcal{A}'$ on an input bitstring x involves selecting the circuit with the appropriate input size, initializing its coin register with uniformly random bits, and then executing it with input x. If the circuits of $\mathcal{A}'$ are polynomial-time uniform, we say that $\mathcal{A}'$ is an efficient, or classical probabilistic polynomial-time algorithm (PPT for short.) We will frequently use PPTs to model the most general efficient classical algorithms.

For our purposes, the space of pure states of a quantum device will be identified with a Hilbert space $\mathcal{H}_n \cong (\mathbb{C}_2)^{\otimes n}$ of a finite number n of qubits. We will identify some fixed orthonormal basis (called the

computational basis) of $\mathcal{H}_n$ with the corresponding space $\{0,1\}^n$ of classical states, so that, e.g, $|x\rangle$ for $x \in \{0,1\}^n$ denotes a basis element of $\mathcal{H}_n$. The space of density operators of n qubits will be denoted $\mathfrak{D}(\mathcal{H}_n)$; a state in this space can be interpreted as a probabilistic mixture of pure states, albeit not in a unique way. We will discuss valid quantum transformations of three types. The first are measurements, which act on a state $|\psi\rangle \in \mathcal{H}_n$ by projecting some or all of the qubits into the computational basis states $\{|0\rangle, |1\rangle\}$. The second are unitary maps, i.e., linear operators $U : \mathcal{H}_n \to \mathcal{H}_n$ satisfying $U^\dagger U = \mathbb{1}_n$, where $\mathbb{1}_n$ denotes the n-qubit identity operator. The third are CPTP maps, i.e., completely positive trace-preserving maps $\Phi : \mathfrak{D}(\mathcal{H}_n) \to \mathfrak{D}(\mathcal{H}_m)$. CPTP maps are the most general type of evolution, encompassing unitary maps, measurement, and discarding (or tracing out) of subspaces. For example, a unitary operator $U \in U(\mathcal{H}_n)$ can be expressed as a CPTP map by writing $\rho \mapsto U\rho U^\dagger$, where $\rho \in \mathfrak{D}(\mathcal{H}_n)$.

A quantum circuit C is a sequence of local unitary gates on a fixed number (say n) of qubits; these gates , when composed together, implement some unitary operator $U_C \in U(2^n)$. Definitions of circuit ensembles and distributions over circuit ensembles are defined precisely as in the classical case. A quantum algorithm $\mathcal{A}$ is a (classically-)polynomial-time uniform family of quantum circuits; algorithms can also include measurements and discarding (or tracing-out) of subsystems, so long as these also admit efficient classical descriptions. The input and output size of a quantum algorithm can vary, and will have to be deduced from context. For example, given a QPT $\mathcal{A}$, the expression $\Pr[\mathcal{A}(|0^n\rangle) = 1]$ will take the value zero unless $\mathcal{A}$ has a specific, labeled output qubit which is measured at the end of the computation.

2 Quantum encryption

In this section, we discuss a notion of encryption for quantum states with computational assumptions. Interestingly, this topic has not received significant attention as yet. In Section 2.1, we will recall how to construct a classical function which appears pseudorandom to quantum adversaries, by means of a function which is one-way against quantum adversaries. In Section 2.2, we define a notion of symmetric-key quantum encryption, together with associated notions of IND-CPA and IND-CCA1 security. We then describe a scheme which is IND-CCA1-secure under the assumption that quantum-secure one-way functions exist. While this particular scheme is new, encryption of quantum states with computational assumptions was also recently (and independently) considered by Broadbent and Jeffery [14]. A complete framework for this topic, including considerations about semantic security, will appear in an upcoming work [5].

2.1 Quantum-secure pseudorandomness

We begin with two primitives for encryption: quantum-secure one-way functions, and quantum-secure pseudorandom functions. These are both classical, efficiently computable functions which are in some sense resistant to quantum analysis. In the case of one-way functions, we demand that inversion is hard; in the case of pseudorandom functions, we demand that distinguishing from perfectly random functions is hard.

Definition 1. *A PT-computable function $f : \{0,1\}^* \to \{0,1\}^*$ is a quantum-secure one-way function (qOWF) if for every QPT $\mathcal{A}$,*

$$\Pr_{x \in_R \{0,1\}^n}\left[\mathcal{A}(f(x), 1^n) \in f^{-1}(f(x))\right] \leq \mathrm{negl}(n)\,,$$

where the probability is taken over $x \in_R \{0,1\}^n$ as well as the measurements of $\mathcal{A}$.

Definition 2. *A PT-computable function family $f_k : \{0,1\}^n \to \{0,1\}^m$ is a quantum-secure pseudorandom function (qPRF) if for every QPT $\mathcal{A}$,*

$$\left|\Pr_{k \in_R \{0,1\}^n}[\mathcal{A}^{f_k}(1^n) = 1] - \Pr_{g \in_R \mathcal{F}_{n,m}}[\mathcal{A}^{g}(1^n) = 1]\right| \leq \mathrm{negl}(n)\,,$$

where $\mathcal{F}_{n,m}$ denotes the space of all functions from $\{0,1\}^n$ to $\{0,1\}^m$.

Classically, one-way functions are the fundamental primitive underpinning encryption. A series of basic results shows that one-way functions can be turned into pseudorandom functions, which can then be used for defining probabilistic encryption schemes. This series of results carries over to the quantum-secure case without much of a change (although some proofs are somewhat more involved.) For example, it is known how to construct qPRFs from qOWFs.

Theorem 1. *If quantum-secure one-way functions exist, then so do quantum-secure pseudorandom functions.*

Proof. (Sketch.) It is folklore that the well-known Håstad et al. result that pseudorandom generators can be constructed from any one-way function [26] carries over to the quantum-secure case. Roughly speaking, the reasoning is that the reduction in the proof is done in a "black-box" way, i.e., only by feeding inputs into the adversary and then analyzing the resulting outputs. The quantum-secure case then simply involves replacing PPTs with QPTs in the appropriate places. Proving that the standard GGM construction [22] of PRFs from pseudorandom generators is still secure in the setting of quantum adversaries is more involved; this was established by Zhandry [36]. □

2.2 Symmetric-key encryption of quantum states

It is well-known how to encrypt quantum states with information-theoretic security, via the so-called quantum one-time pad. To encrypt a single-qubit state ρ, we choose two classical bits at random, use them to select a random Pauli matrix $P \in \{\mathbb{1}, X, Y, Z\}$, and perform $\rho \mapsto P\rho P^\dagger$. To encrypt an n-qubit quantum state ρ, we select $r \in_R \{0,1\}^{2n}$ and apply

$$\rho \longmapsto P_r \rho P_r^\dagger \,, \tag{2.1}$$

where P_r denotes the element of the n-qubit Pauli group indexed by r.

One disadvantage of the quantum one-time pad is that parties must share two bits of randomness for every qubit which they wish to transmit securely. In particular, one cannot securely exchange multiple messages with the same key. To address this issue, we must settle for computational security assumptions and use pseudorandomness to select r. A general encryption scheme for quantum states is then defined as follows.

Definition 3. *A symmetric-key quantum encryption scheme is a triple of QPTs:*

- *(key generation)* $\mathsf{KeyGen} : 1^n \longmapsto k \in \{0,1\}^n$*;*
- *(encryption)* $\mathsf{Enc}_k : \mathfrak{D}(\mathcal{H}_m) \longrightarrow \mathfrak{D}(\mathcal{H}_c)$*;*
- *(decryption)* $\mathsf{Dec}_k : \mathfrak{D}(\mathcal{H}_c) \longrightarrow \mathfrak{D}(\mathcal{H}_m)$*;*

where m and c are polynomial functions of n, and the QPTs satisfy $\|\mathsf{Dec}_k \circ \mathsf{Enc}_k - \mathbb{1}_m\|_\diamond \leq \mathrm{negl}(n)$ for all $k \in \mathbf{supp}\,\mathsf{KeyGen}(1^n)$.

Public-key quantum encryption schemes are defined in an analogous manner. The encryption schemes we will need must produce ciphertexts which are computationally indistinguishable. In some cases, the ciphertexts will need to remain indistinguishable even to adversaries which possess oracle access to the encryption algorithm (and sometimes also even the decryption algorithm.) This security notion is captured by the following definition.

Definition 4. *A symmetric-key quantum encryption scheme is IND-secure if for all QPTs $\mathcal{A}, \mathcal{A}'$,*

$$\left|\Pr[(\mathcal{A}' \circ \mathsf{Enc}_k \otimes \mathbb{1}_s \circ \mathcal{A}) \cdot 1^n = 1] - \Pr[(\mathcal{A}' \circ \Xi_{\mathsf{Enc}_k|0^m\rangle\langle 0^m|} \otimes \mathbb{1}_s \circ \mathcal{A}) \cdot 1^n = 1]\right| \leq \mathrm{negl}(n)\,,$$

where $\Xi_\sigma : \rho \mapsto \sigma$ is the "forgetful" map, and s is a polynomial function of n. If $\mathcal{A}$ and $\mathcal{A}'$ have oracle access to Enc_k, then we say that the scheme is IND-CPA secure. If in addition $\mathcal{A}'$ has oracle access to Dec_k, then we say that the scheme is IND-CCA1 secure.

The two QPTs $\mathcal{A}$ and $\mathcal{A}'$ together model the adversary. The definition above captures the idea of a certain "security game" between an adversary and a challenger. The game proceeds in steps: (i.) the key is selected and the adversary receives access to the appropriate oracles, (ii.) after some computation, the adversary transmits the first part of a bipartite state ρ_{ms} to a challenger, (iii.) the challenger either encrypts this or replaces it with the encryption of $|0^m\rangle\langle 0^m|$, and then returns the result to the adversary, and (iv.) the adversary must decide which choice the challenger made. The scheme is considered secure if the adversary can do no better than random guessing. As shown in [5], this definition is equivalent to a security notion called *semantic security*; roughly speaking, this notion captures the idea that anyone that tries to compute anything about a plaintext gains no advantage by possessing its encryption. In addition, Definition 4 is equivalent to several natural variants, where e.g., the challenger chooses to encrypt one of two messages provided by the adversary, or where the game is played over multiple rounds. The latter guarantees security of transmitting multiple ciphertexts produced via encryption with the same key.

We now show how to use qPRFs to construct simple symmetric-key quantum encryption schemes that satisfy all of the above security conditions.

Theorem 2. *If quantum-secure pseudorandom functions exist, then so do IND-CCA1-secure symmetric-key quantum encryption schemes.*

Proof. Let $\{f_k\}$ be a qPRF. For simplicity we assume that each f_k is a map from $\{0,1\}^n$ to $\{0,1\}^{2n}$. Recall that for $r \in \{0,1\}^2 n$, P_r denotes the element of the n-qubit Pauli group indexed by r. Consider the following scheme:

- $\mathsf{KeyGen}(1^n)$: output $k \in_R \{0,1\}^n$;
- $\mathsf{Enc}_k(\rho)$: choose $r \in_R \{0,1\}^n$; output $|r\rangle\langle r| \otimes P_{f_k(r)} \rho P^\dagger_{f_k(r)}$;
- $\mathsf{Dec}_k(|r\rangle\langle r| \otimes \sigma)$: output $P^\dagger_{f_k(r)} \rho P_{f_k(r)}$.

In the decryption algorithm, we may assume that the first register is always measured prior to decrypting. Correctness of the scheme is straightforward to check: decrypting with the same key and randomness simply undoes the Pauli operation.

We now sketch the proof that the scheme is IND-CCA1 secure; a complete proof will appear in [5]. The key observation is that each query to the encryption oracle is no more useful than receiving a pair $(r, f_k(r))$ for $r \in_R \{0,1\}^{2n}$, and that each decryption oracle is no more useful than receiving a pair $(r, f_k(r))$ for a string r of the adversary's choice. Thus the adversary learns at most a polynomial number of values of f_k. Now, if f_k is a perfectly random function, then these values are completely uncorrelated to the one used to encrypt the challenge. The scheme is thus secure simply by the information-theoretic security of the quantum one-time pad. On the other hand, if f_k is a function in a qPRF, Definition 2 guarantees oracle indistinguishability from perfectly random functions. It follows that, if $(\mathcal{A}, \mathcal{A}')$ can break the actual scheme, then by computational indistinguishability they would also break the perfect scheme, which is impossible. □

We emphasize that the above proof shows that, even in the case where the adversary chooses the randomness r used by the Enc_k and Dec_k oracles, the scheme remains secure. Of course, the randomness for the challenge encryption must still be selected by the challenger. Finally, by combining Theorem 1 and Theorem 2, we have the following.

Theorem 3. *If quantum-secure one-way functions exist, then so do IND-CCA1-secure symmetric-key quantum encryption schemes.*

3 Quantum black-box obfuscation

In this section, we discuss the virtual black-box framework for obfuscating quantum computations. We begin in Section 3.1 with a definition of black-box quantum obfuscator, motivated both by the classical

analogue and an intuitive notion of what a "good obfuscator" should achieve. In Section 3.2, we outline several interesting cryptographic consequences that would follow from the existence of such an obfuscator. Finally, in Section 3.3, we prove a few impossibility results which restrict the range of possibilities for the existence of black-box quantum obfuscators. Interestingly, our results leave open some possibilities, which include (restricted versions) of the most interesting applications. Indeed, it is conceivable that quantum obfuscation could be significantly more powerful than its classical counterpart.

3.1 Definitions

Any reasonable notion of obfuscation involves giving the obfuscated circuit $\mathcal{O}(C)$ to an untrusted party. We accept as fundamental the idea that this obfuscated circuit should implement some particular, chosen functionality f_C, and that the object $\mathcal{O}(C)$ allows the untrusted party to execute that functionality. In the black-box formulation of obfuscation, we demand that this is effectively all that the untrusted party will ever be able to do. The rigorous formulation uses the simulation paradigm: anything which can be efficiently learned from the obfuscated circuit, should also be efficiently learnable simply by evaluating f_C some polynomial number of times. This "virtual black-box" notion was first formulated by Barak et al. [7], and proved impossible to satisfy generically in the classical case.

In the quantum case, there are several complications. First, we are considering the obfuscation of quantum functionalities. This implies that the end user (and hence also any adversary) should be in possession of a quantum computer, and likewise for the simulator. Second, it is conceivable that the obfuscation may not just be another quantum circuit, which is simply a classical state describing a quantum computation. The obfuscator might instead output a quantum state, which is then to be employed by the end user to execute the desired functionality in some well-specified manner. These considerations motivate the following definition.

Definition 5. *A **black-box quantum obfuscator** is a quantum algorithm $\mathcal{O}$ and a QPT $\mathcal{J}$ such that whenever C is an n-qubit quantum circuit, the output of $\mathcal{O}$ is an m-qubit state $\mathcal{O}(C)$ satisfying*

1. *(polynomial expansion) $m = poly(n)$;*
2. *(functional equivalence) $\left\| \mathcal{J}(\mathcal{O}(C) \otimes \rho) - U_C \rho U_C^\dagger \right\|_{\mathrm{tr}} \leq \mathrm{negl}(n)$ for all $\rho \in \mathfrak{D}(\mathcal{H}_n)$;*
3. *(virtual black-box) for every QPT $\mathcal{A}$ there exists a QPT $\mathcal{S}^{U_C}$ such that*

$$\left| \Pr\left[\mathcal{A}(\mathcal{O}(C)) = 1\right] - \Pr\left[\mathcal{S}^{U_C}\left(|0^n\rangle\right) = 1\right] \right| \leq \mathrm{negl}(n)\,.$$

We emphasize that while the "interpreter" algorithm $\mathcal{J}$ must be polynomial-time, the obfuscator itself need not be. In applications, it will be necessary to make the obfuscator polynomial-time; on the other hand, our impossibility results will hold even for inefficient obfuscators. One could consider variants of Definition 5 where the interpreter algorithm is fixed once and for all, or where $\mathcal{O}(C)$ itself consists of both a quantum "advice state" and a circuit which the end user should execute on the advice state and the desired input. It is straightforward to show that all of these variants are equivalent, in the sense that a black-box quantum obfuscator of each variant exists if and only if the other variants exist. Since we are primarily concerned with possibility vs impossibility, we will stick with the formulation in Definition 5. We also remark that the interpreter is a natural addition to the classical black-box definition when passing to the quantum case. In order for the definition to make sense, there should be *some efficient way* to use $\mathcal{O}(C)$ to implement U_C; whatever that efficient procedure is, we have here called it an interpreter and denoted it by $\mathcal{J}$.

We also point out that the no-cloning theorem opens up the possibility of *computationally unbounded adversaries*. In the classical case, such an adversary could simply execute the circuit on every input, and thus learn far more than is possible for a polynomial-time black-box simulator. Quantumly, however, a computationally unbounded adversary is restricted both by the no-cloning theorem and the limitations of measurement. The adversary may not be able to acquire multiple copies of the obfuscated state, and the single state may be partially (or completely) destroyed when measured. It is thus not *a priori* clear that an

unbounded adversary could always outmatch a polynomial-time black-box simulator. The appropriate definition is a straightforward modification of Definition 5, where we replace the third condition with the following:

3. *(information-theoretic virtual black-box) for every quantum adversary $\mathcal{A}$ there exists a QPT $\mathcal{S}^{U_C}$ such that*
$$\left|\Pr\left[\mathcal{A}(\mathcal{O}(C))=1\right]-\Pr\left[\mathcal{S}^{U_C}\left(|0^n\rangle\right)=1\right]\right|\leq \mathrm{negl}(n)\,.$$

3.2 Applications of efficient black-box obfuscators

In this section, we motivate the study of quantum black-box obfuscation by giving a few example applications. Unsurprisingly, these applications require that the obfuscation algorithm is itself quantum polynomial-time; strictly speaking, this is not required of Definition 5. Many of these applications are motivated by known classical applications of classical black-box obfuscators. Although our impossibility results will put some restrictions on these applications, they remain interesting. In fact, some of the applications (such as quantum-secure one-way functions) will be used in the impossibility proofs themselves. We point out that, while most of the applications below are written in terms of quantum functionality (e.g., encryption of quantum states), one can just as well consider the weaker case of classical functionality, in this case achieved via quantum means (e.g., via a quantum algorithm for obfuscation.)

3.2.1 Quantum-secure one-way functions

The first application shows that, if there exists a classical algorithm for obfuscating quantum computations, then quantum-secure one-way functions exist. By the results discussed in Section 2, this also implies the existence of quantum-secure pseudorandom generators, quantum-secure pseudorandom functions, and IND-CCA1-secure symmetric-key quantum encryption schemes.

Proposition 1. *If there exists a classical probabilistic algorithm which is a quantum black-box obfuscator, then quantum-secure one-way functions exist.*

Proof. The proof is essentially the same as that of Lemma 3.8 in [7]. For all $a\in\{0,1\}^n$ and $b\in\{0,1\}$, we define
$$U_{a,b}:|x,y\rangle\longmapsto\begin{cases}|a,y\oplus b\rangle & \text{if } x=a;\\ |x,y\rangle & \text{otherwise.}\end{cases}$$
Define a function $f:\{0,1\}^*\to\{0,1\}^*$ by $f(a,b,r)=\mathcal{O}_r(U_{a,b})$ where $\mathcal{O}$ is the obfuscator[2] as in the hypothesis, and $\mathcal{O}_r$ denotes the same algorithm, but with randomness coins initialized to r. Clearly, inverting f requires computing b from $\mathcal{O}_r(U_{a,b})$. Moreover, with only black-box access to $U_{a,b}$ (for uniformly random a,b) the probability of correctly outputing b in polynomial time is at most $1/2+\mathrm{negl}(n)$. By the black-box property of $\mathcal{O}$, we then have
$$\begin{aligned}\Pr_{a,b}[A(f(a,b,r))=b]&=\Pr_{a,b}[A(\mathcal{O}_r(a,b))=b]\\&\leq\Pr_{a,b}\left[S^{U_{a,b}}(1^n)=b\right]+\mathrm{negl}(n)\\&\leq\frac{1}{2}+\mathrm{negl}(n)\,,\end{aligned}$$
which completes the proof. □

We remark that the above proof fails if the obfuscator is a quantum algorithm—even if its output is itself classical. The issue is that one-way functions must be deterministic; while one can turn a classical probabilistic algorithm into a deterministic one by making the coins part of the input, this is not possible quantumly. We leave the problem of constructing cryptographically useful primitives from a fully quantum obfuscator (or even just from a quantum encryption scheme) as an interesting open question.

[2]For simplicity of notation, we omit $\mathcal{J}$ and assume that $f(a,b,r)=\mathcal{O}_r(U_{a,b})$ is in fact a classical circuit for $U_{a,b}$.

3.2.2 CPA-secure private-key quantum encryption

Can we say anything about encryption of data if we know that *quantum* algorithms for quantum black-box obfuscation exist? While we do not know how to extract one-way functions, we can nonetheless produce useful encryption schemes, as follows.

Proposition 2. *If quantum black-box obfuscators exist, then so do IND-CPA-secure symmetric-key quantum encryption schemes.*

Proof. (Sketch.) Let $(\mathcal{O}, \mathcal{J})$ be a quantum black-box obfuscator. We consider an adaptation of the unitary operator $U_{a,b}$ defined above, but now with Pauli group action instead of XOR, and with two n-bit registers:

$$U'_{r,k} : |x, y\rangle \longmapsto \begin{cases} |x, P_r^\dagger y\rangle & \text{if } x = k; \\ |x, y\rangle & \text{otherwise,} \end{cases}$$

Now consider the following scheme for encrypting n-qubit quantum states.

- $\mathsf{KeyGen}(1^n)$: output $k \in_R \{0,1\}^n$;
- $\mathsf{Enc}_k(\rho)$: choose $r \in_R \{0,1\}^n$; output $P_r \rho P_r^\dagger \otimes \mathcal{O}(U_{r,k})$;
- $\mathsf{Dec}_k(\sigma \otimes \tau)$: output the second register of $\mathcal{J}(\tau \otimes |k\rangle\langle k| \otimes \sigma)$.

To check correctness, we apply the functionality-preserving property of the obfuscator. A decryption of a valid encryption with the same key yields

$$\begin{aligned} \mathsf{Dec}_k(\mathsf{Enc}_k(\rho)) &= \mathrm{Tr}_1 \left[\mathcal{J}(\mathcal{O}(U_{r,k}) \otimes |k\rangle\langle k| \otimes P_r \rho P_r^\dagger) \right] \\ &= \mathrm{Tr}_1 \left[U_{r,k}(|k\rangle\langle k| \otimes P_r \rho P_r^\dagger) U_{r,k}^\dagger \right] \\ &= \mathrm{Tr}_1 \left[|k\rangle\langle k| \otimes \rho \right] \\ &= \rho \,. \end{aligned}$$

as desired. IND-CPA security follows from the black-box property of the obfuscator, as follows. Let $\mathcal{A}$ be an adversary with access to the encryption oracle. Since the output of the encryption is a product state, $\mathcal{A}$ can be simulated by an adversary $\mathcal{S}$ that has only the first register of the ciphertext (i.e., $P_r \rho P_r^\dagger$) and black-box access to the unitary $U'_{r,k}$. It's then clear that $\mathcal{S}$ can only succeed in the challenge stage of Definition 4 by discovering the secret input for $U'_{r,k}$ or by guessing the response to the challenge. In any case, $\mathcal{S}$ (and hence also $\mathcal{A}$) succeeds with probability at most $1/2 + \mathrm{negl}(n)$. □

3.2.3 Public-key encryption from private-key encryption

As we now show, combining black-box obfuscation with one-way functions yields even stronger encryption functionality.

Proposition 3. *If quantum black-box obfuscators and quantum-secure one-way functions exist, then so do IND-CPA-secure public-key quantum encryption schemes.*

Proof. (Sketch.) Under the hypothesis, Theorem 3 implies the existence of IND-CCA1-secure symmetric-key encryption schemes for quantum states. Let $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be such a scheme; for concreteness, we may take the scheme described in Theorem 2. For $x \in \{0,1\}^n$, let $\mathsf{Enc}_{(x)}$ denote the encryption circuit for key x; this is the circuit that accepts two input registers (one for randomness, and one for the plaintext) and outputs the ciphertext. Now define a public-key encryption scheme $(\mathsf{KeyGen}', \mathsf{Enc}', \mathsf{Dec}')$ as follows.

- $\mathsf{KeyGen}'(1^n)$: output $sk := k \in_R \{0,1\}^n$ (secret key) and $pk := \mathcal{O}\left(\mathsf{Enc}_{(sk)}\right)$ (public key);
- $\mathsf{Enc}'_{pk}(\rho)$: choose $r \in_R \{0,1\}^n$; output $pk(|r\rangle\langle r| \otimes \rho)$;
- $\mathsf{Dec}'_{sk}(\sigma)$: output $\mathsf{Dec}_{sk}(\sigma)$.

The correctness of this scheme follows directly from the functionality-preserving property of $\mathcal{O}$ and the correctness of the private-key scheme. To prove IND-CPA security for the public-key scheme, we rely on the black-box property. It implies that any QPT adversary $\mathcal{A}$ with access to the public key can be simulated by a QPT $\mathcal{S}$ having only black-box access to $\mathsf{Enc}_{(sk)}$. The QPT $\mathcal{S}$, in turn, can be simulated by a QPT $\mathcal{S}'$ which has both decryption and encryption oracles for the private-key scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$. It may not be immediately obvious that the decryption oracle is necessary; this is the case because black-box access to $\mathsf{Enc}_{(sk)}$ enables $\mathcal{S}$ to select the randomness used for encryption, thus gaining the ability to evaluate pairs $(r, f_{sk}(r))$ where f is the qPRF from the private-key scheme.

Now we have that, if $\mathcal{A}$ can distinguish ciphertexts during the challenge, then so can $\mathcal{S}'$; since the ciphertexts themselves are the same for the public-key scheme and the private-key scheme, this contradicts the IND-CCA1 security of the private-key scheme. □

A few remarks are in order. First, in [5] it is shown that IND-CPA-secure public-key quantum encryption schemes exist under the assumption that quantum-secure trapdoor permutations exist. This is a stronger assumption than one-way functions. Proposition 3 can then be thought of as replacing this strengthening of assumptions with an obfuscator. In [14] it is shown how to use quantum-secure classical public-key encryption to produce quantum public-key encryption (by encrypting the key for the quantum one-time pad); this amounts to the same assumption on primitives as in [5]. An important difference between [5, 14] and Proposition 3 is that the scheme from Proposition 3 may have public keys which are quantum states. Such schemes have not been considered before, and (due to no-cloning) would have significantly different features from their classical counterparts.

An interesting question is if there could be public-key encryption for classical data with classical ciphertexts, but where the encryption procedure is performed by a quantum algorithm. While this question remains open, our impossibility results will show that this cannot be achieved in a generic way via Proposition 3.

3.2.4 Quantum fully homomorphic encryption

We briefly recall the idea of fully homomorphic encryption (FHE). For thorough definitions and the appropriate notions of security in the fully quantum case, see [14]. Without considering all of the details, we will view QFHE as an encryption scheme (just as in Definition 3), but where KeyGen produces an extra "evaluation" key k_{eval}, and there is an "evaluation" algorithm:

- $\mathsf{Eval}_{k_{\text{eval}}} : \mathfrak{D}(\mathcal{H}_m \otimes \mathcal{H}_g) \longrightarrow \mathfrak{D}(\mathcal{H}_m)$.

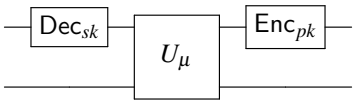
We imagine a party (henceforth, *server*) in possession of k_{eval} and a ciphertext $\mathsf{Enc}_k(\rho)$ provided by another party (henceforth, *client*.) The evaluation algorithm then enables the server to produce the ciphertext $\mathsf{Enc}_k(G_k \rho G_k^\dagger)$, where G is a gate of the server's choice. A classical string describing the choice of gate G (and which qubits $k, k+1, \ldots$ of ρ it should be applied to) is input into the register $\mathcal{H}_g$. In general, we may consider the case where k_{eval} is itself a quantum state. Depending on the details of the scheme, this key may be partly or fully consumed by Eval; indeed, this is the case in [14]. Depending on the consumption rate, this might violate the (classically standard) *compactness* requirement for FHE, namely that the amount of communication between the client and the server should scale only with the size of the ciphertext, and not with the size of the computation the server wishes to perform.

Proposition 4. *If quantum black-box obfuscators and one-way functions exist, then so do IND-CPA-secure quantum fully homomorphic encryption schemes.*

Proof. (Sketch.) We will consider the public-key case, which turns out to be simpler. Let $(\mathcal{O}, \mathcal{J})$ be a quantum obfuscator, and $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ an IND-CPA-secure public-key scheme. We adapt KeyGen to produce an evaluation key, and describe the evaluation algorithm. We will require a universal circuit U_μ for performing gates on m-qubit states; this circuit accepts two inputs: an m-qubit state, and a description of a gate and indices of the qubits to which the gate should be applied. In our usage, m will be the number of qubits of the ciphertext state.

- $\mathsf{KeyGen}'(1^n)$: output $\mathsf{KeyGen}(1^n) = (sk, pk)$ and $k_{\text{eval}} = \mathcal{O}(\mathsf{Enc}_{pk} \circ U_\mu \circ \mathsf{Dec}_{sk})$;
- $\mathsf{Eval}_{k_{\text{eval}}} : \rho \otimes |G\rangle\langle G| \longmapsto \mathcal{J}(k_{\text{eval}} \otimes \rho \otimes |G\rangle\langle G|)$.

where $|G\rangle\langle G|$ is again just a classical string instructing U_μ to apply the desired gate. A circuit for $\mathsf{Enc}_{pk} \circ U_\mu \circ \mathsf{Dec}_{sk}$ is given below; the gate register is represented by the bottom wire.



We now want to show that $(\mathsf{KeyGen}', \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ is a public-key QFHE scheme. The homomorphic property follows directly from the definition of Eval and the functionality-preserving property of the obfuscator. The security of the encryption scheme follows from IND-CPA security of $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ and the black-box property of $(\mathcal{O}, \mathcal{J})$. The black-box property implies that each execution of the Eval algorithm is no more useful than providing the server with an encryption of $G\rho G^\dagger$. However, in the IND-CPA setting, the adversary can already use the CPA oracle to produce encryptions of *arbitrary* plaintexts of her choice (as opposed to just ones which are modifications of the plaintext provided by the client.) There is one additional wrinkle: by repeatedly applying gates (or even just the identity), the adversary can also produce multiple encryptions during the challenge round. However, as shown in [14], single-message IND-CPA is equivalent to multiple-message IND-CPA. By the assumption that $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ is IND-CPA secure, it follows that the homomorphic scheme is also secure.

We remark that, in general, the encryption procedure Enc_{pk} may require an external source of randomness. This is certainly the case in classical encryption, but may not be required if the Enc algorithm is allowed to perform measurements. In any case, since we are starting with an IND-CPA public-key scheme, the adversary already has access to the public key and the ability to encrypt with randomness of her choice; the ability to choose randomness in Eval is of no additional benefit. □

3.2.5 Public-key quantum money

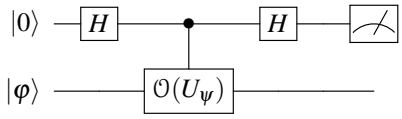
Quantum money. The idea of "quantum money" first arose in work by Wiesner [35]. The core idea is simple: use a quantum state for representing currency in such a way that the no-cloning theorem of quantum mechanics prevents counterfeiting. These ideas were refined and developed further in several works [2, 3, 10, 17, 30]; some of these works also included explicit proposals based on various hardness assumptions.

Informally, a *quantum money scheme* consists of two algorithms: Mint, which produces quantum states, and Verify, which accepts an input state and then either accepts or rejects. If the different states produced by Mint are distinguishable, then we refer to them as *bills*; if they are indistinguishable, then we call them *tokens* (if Verify consumes them) or *coins* (if Verify does not consume them.) In all quantum money schemes, we imagine an authority (typically called the bank) which runs Mint repeatedly to produce money; in addition, the Verify algorithm should accept only on states produced by the bank. Depending on the particular scheme, this might only be true if Verify is executed by the bank (private-key money), or it might be true for any party (public-key money.)

In this language, Wiesner's original idea [35] was for a private-key scheme for bills, which is as follows. Each execution of Mint produces two random classical bitstrings $r, s \in \{0,1\}^{2n}$ as well as an n-qubit quantum state $|\psi_r\rangle$, with each qubit initialized in one of the states $|0\rangle, |1\rangle, |+\rangle, |-\rangle$, as determined by the bits of r. The bank records the pair (r, s) in a secret table, and publishes $(s, |\psi_r\rangle)$. The bank verifies by using s to look up the correct r in the table, and then performing the measurements in the correct basis and checking the results against r.

Public-key money from circuit obfuscation. While private-key money schemes are relatively straightforward to construct, public-key proposals appear to be much more difficult, and require computational assumptions. In analogy to its role in producing public-key encryption schemes from private-key

ones (Proposition 2), an obfuscator can sometimes be used to turn private-key money schemes to public-key ones. The use of an obfuscator to create a particular quantum money scheme was considered by Mosca and Stebila [30]. Their scheme (in our language) is as follows. Each execution of Mint produces a Haar-random n-qubit quantum state $|\psi\rangle$, together with the obfuscation $\mathcal{O}(U_\psi)$ of a circuit[3] for $U_\psi = \mathbb{1} - 2|\psi\rangle\langle\psi|$. The bill consists of the pair $(\mathcal{O}(U_\psi), |\psi\rangle)$. Verify$(|\varphi\rangle)$ consists of executing the following:



and accepting iff the measurement returns 1. It's easy to check that the above succeeds only on valid states; moreover, in that case, the state $|\psi\rangle$ is output in the second register, so that verification can be repeated. To show resistance of the above scheme to counterfeiting, one can use Aaronson's Complexity-Theoretic No-Cloning Theorem [2], which states that cloning the state $|\psi\rangle$ while in possession of oracle access to $|U_\psi\rangle$ requires $\Omega(2^{n/2})$ queries. The first published proof of this theorem (as well as its first appearance in the form required here) was in [3].

Unfortunately, we will later show that obfuscation of quantum circuits in the form required by Mosca and Stebila is impossible. What remains possible is a setting in which both $|\psi\rangle$ and $\mathcal{O}(U_\psi)$ are quantum states, and another circuit (which is publicly known and independent of $|\psi\rangle$) is used for verification. Moreover, as we will also show, any black-box obfuscation scheme which outputs states that can be efficiently cloned is also impossible. We thus conjecture the following.

Conjecture 1. *If quantum black-box obfuscators exist, then so do public-key quantum money schemes.*

If the relevant obfuscation is a consumable state, then this would result in a token scheme. If it can be reused to perform verification repeatedly[4], then the result would be a bills scheme. We remark that, in any case, all of the public-key money states discussed above should be authenticated by the bank; otherwise a merchant would only know that he was handed *some* pair (state, circuit) where the circuit executed on the state outputs "accept"—a clearly inadequate state of affairs.

3.3 Impossibility results

3.3.1 Impossibility of two-circuit obfuscation

Barak et. al. [7] showed that black-box obfuscation is impossible by constructing an explicit circuit family that cannot be black-box obfuscated. We begin with a similar result in the quantum setting. We show that quantum black-box obfuscation is impossible in any setting where the adversary can gain access to two outputs of the obfuscator on *different* inputs. We formalize this notion by defining a "black-box two-circuit obfuscator," defined just as in Definition 5 but with the following strengthening of the virtual black-box condition:

3. *(two-circuit virtual black-box) for every pair of quantum circuits C_1 and C_2 and every quantum adversary $\mathcal{A}$ there exists a quantum simulator $\mathcal{S}^{U_{C_1},U_{C_2}}$ and a negligible ε_2 such that*

$$\left|\Pr[\mathcal{A}(\mathcal{O}(C_1)\otimes\mathcal{O}(C_2))=1] - \Pr\left[\mathcal{S}^{U_{C_1},U_{C_2}}\left(|0\rangle^{\otimes|C_1|+|C_2|}\right)=1\right]\right| \leq \varepsilon_2(n,\min\{|C_1|,|C_2|\}).$$

We now show that there exists a family of circuits which is unobfuscatable under the above definition. We emphasize that our result holds even when the outputs of the obfuscator are quantum states, and even if these states are *single-use only*, i.e., if the interpreter $\mathcal{J}$ irrevocably destroys the obfuscated state during use.

[3] For most $|\psi\rangle$, the circuit U_ψ will not have polynomial length. However, as pointed out by [2], one can instead select $|\psi\rangle$ from an approximate t-design without a significant loss in security.

[4] For example, if successful verification also outputs another state which is sufficiently close to the original state.

We first define a *circuit-pair family* to be an ensemble of distributions over pairs of circuits. More precisely, if $\mathcal{C}$ is a circuit-pair family, then there exists a Turing machine M which, on input a positive integer parameter n (in unary), outputs a classical description of a pair of circuits (C_n, D_n) drawn at random from some distribution $\mathcal{C}_n$ on pairs of poly(n)-size circuits. If M is polynomial-time, then we say that $\mathcal{C}$ is a *poly-time circuit-pair family.*

We also define a *state-pair family* analogously. If $\mathcal{C}'$ is a state-pair family, then there exists a (not necessarily polynomial-time) quantum algorithm which, on input n in unary, outputs a pair of density operators (ρ_n, σ_n) drawn at random from some distribution $\mathcal{C}'_n$ on quantum states on poly(n)-many qubits. Given a circuit-pair family $\mathcal{C}$ and a state-pair family $\mathcal{C}'$, we say that $\mathcal{C}'$ is an obfuscation of $\mathcal{C}$ if there exists a computable map $\mathcal{C} \to \mathcal{C}'$ assigning to each circuit a corresponding state, in a manner that satisfies the two-circuit obfuscation definition above.

With these definitions, we can now state our first impossibility result.

Theorem 4. *There exists a poly-time quantum circuit-pair family $\mathcal{C}$ such that no state-pair family is an obfuscation of $\mathcal{C}$.*

Proof. Let $(\mathcal{O}, \mathcal{J})$ be a black-box quantum two-circuit obfuscator. The poly-time quantum circuit-pair family $\mathcal{C}$ consists of quantum circuits for implementing the following pairs of unitary operators. Each pair is parameterized by an input size n, as well as bitstrings a, b chosen uniformly at random from $\{0,1\}^n$.

$$U_{a,b} : |x, y\rangle \longmapsto \begin{cases} |x, y \oplus b\rangle & \text{if } x = a; \\ |x, y\rangle & \text{otherwise.} \end{cases} \tag{3.1}$$

$$V_{a,b} : |C, z\rangle \longmapsto \begin{cases} |C, z \oplus 1\rangle & \text{if } C(a) = b; \\ |C, z\rangle & \text{otherwise.} \end{cases} \tag{3.2}$$

The registers indexed by x and y are of size n. The register indexed by C accepts a circuit description (under some fixed encoding), and needs to be able to handle inputs of size $|\mathcal{O}(C_{a,b})|$ (i.e. of size equal to the number of qubits in the state $\mathcal{O}(C_{a,b})$). Here $C_{a,b}$ is a fixed, explicit poly(n)-size circuit for $U_{a,b}$. The second register of $V_{a,b}$ has size one.

Note that both of these unitaries can be implemented by efficient quantum circuits. We choose some particular set of such circuits, and henceforth denote them by $C_{a,b}$ and $D_{a,b}$, respectively. The idea for the proof is as follows. Consider an adversary $\mathcal{A}$ which is ignorant of the randomly selected a and b, and consider two scenarios: in the first, $\mathcal{A}$ is given access to *any* pair of *circuits* that implement $U_{a,b}$ and $V_{a,b}$; in the second, $\mathcal{A}$ only has oracle access to $U_{a,b}$ and $V_{a,b}$. The point is that, in the first case, $\mathcal{A}$ can execute $V_{a,b}$ on a circuit for $U_{a,b}$; provided that the latter is not too long, $\mathcal{A}$ will achieve something that is impossible to do with only black-box access. Specifically, it is only in the first case that $\mathcal{A}$ will be able to tell if the first circuit/oracle implements $U_{a,b}$, or if it has surreptitiously been replaced by the identity operator!

Things are somewhat complicated by the fact that the obfuscator outputs states instead of circuits. We will need to enable $\mathcal{A}$ to execute these states on one another. It will thus be necessary to replace $D_{a,b}$ with a related circuit $D'_{a,b}$. Roughly speaking, this circuit will check if its input, when interpreted as a quantum advice state to the algorithm $\mathcal{J}$, maps the input a to the output b. A precise description follows. First, $D'_{a,b}$ will have three registers: an input register of m qubits, a work register of $2n$ qubits, and an output register of 1 qubit initialized in the $|0\rangle$ state. When given as input a quantum state ρ on m qubits, it will initialize the first n bits of the work register to $|a\rangle$, then execute the appropriate unitary circuit of $\mathcal{J}$ on $\rho \otimes |a\rangle$. Finally, if the output register of the latter computation contains $|b\rangle$, $D'_{a,b}$ will flip the contents of the output register. We remark that, by a simple counting argument over circuits, this occurs for only an exponentially small fraction of possible input states ρ.

Recall that the $2n$-qubit identity operator is denoted by $\mathbb{1}_{2n}$, and is implemented by the obvious circuit which we will denote by I_{2n}. We observe that, for every QPT algorithm $\mathcal{S}$ there exists a polynomial t and a negligible ε_1 so that:

$$\left|\Pr\left[\mathcal{S}^{U_{a,b},D'_{a,b}}\left(|0\rangle^{\otimes t(n)}\right)=1\right]-\Pr\left[\mathcal{S}^{Id_{2n},D'_{a,b}}\left(|0\rangle^{\otimes t(n)}\right)=1\right]\right|\leq \varepsilon_1(n)\,. \tag{3.3}$$

Here the probability is taken over the uniformly random choice of a and b as well as all of the measurement outcomes of $\mathcal{S}$. The above is an easy corollary of the tightness of the Grover bound for unstructured quantum search [9]. Indeed, given the definitions of $U_{a,b}$ and $D'_{a,b}$, it's clear that with only polynomial queries and no knowledge of a or b, $\mathcal{S}$ is faced precisely with unstructured search for an exponentially small "marked space." This marked space is only encountered if $\mathcal{S}$ correctly guesses a, or correctly guesses an obfuscation of a circuit that maps a to b.

Now consider the QPT algorithm $\mathcal{A}$ that, given as input the obfuscated states $\mathcal{O}(C)$ and $\mathcal{O}(D)$, simply executes the quantum algorithm $\mathcal{J}$ on their tensor product, accepting if and only if the outcome is 1. Notice that this succeeds with constant probability $\alpha > 0$ if C is functionally equivalent to $C_{a,b}$ and D is functionally equivalent to $D'_{a,b}$. On the other hand, this same algorithm $\mathcal{A}$ accepts with at most negligible probability when C is functionally equivalent to I_{2n} (and D is still functionally equivalent to $D'_{a,b}$); indeed, this only happens is if $a=b$. Thus there exists a negligible function ε_2 so that:

$$\left|\Pr\left[\mathcal{A}(\mathcal{O}(D'_{a,b}),\mathcal{O}(I_{2n}))=1\right]-\Pr\left[\mathcal{A}\left(\mathcal{O}(D'_{a,b})\otimes\mathcal{O}(C_{a,b})\right)=1\right]\right|\geq \alpha-\varepsilon_2(n)\,. \tag{3.4}$$

To complete the proof, we explicitly define the poly-time circuit-pair family $\mathcal{C}$. The distribution $\mathcal{C}_n$ is generated by choosing a,b uniformly at random from $\{0,1\}^n$, and then choosing a bit $r\in 0,1$ at random; if $r=0$, we output the circuit pair $(C_{a,b},D'_{a,b})$, and otherwise we output $(I_{2n},D'_{a,b})$. For this distribution, equations (3.3) and (3.4) together show that no state-pair family is an obfuscation of $\mathcal{C}$. □

3.3.2 Generalizing the impossibility result

Our goal in this section is to extend the two-circuit impossibility proof from the prior section to the case of obfuscating a single circuit. For our impossibility proof, we require an additional condition on the obfuscator: that each of its outputs is reusable a polynomial number of times. This is a natural condition which is automatically satisfied by classical obfuscators (as well as quantum obfuscators with classical outputs), since their outputs can be perfectly copied.

Definition 6. *A **reusable-black-box quantum obfuscator** is a quantum algorithm $\mathcal{O}$ and a QPT $\mathcal{J}$ such that whenever C is an n-qubit quantum circuit, $\mathcal{O}(C)$ is an m-qubit quantum state satisfying*

1. *(polynomial slowdown) $m=poly(n,|C|)$;*
2. *(functional equivalence) $\left\|\mathcal{J}(\mathcal{O}(C)\otimes\cdot)-C\cdot C^\dagger\right\|_\diamond\leq \mathrm{negl}(n,|C|)$;*
3. *(reusability) after execution of $\mathcal{J}$, an output register contains a state which satisfies (2.);*
4. *(virtual black-box) for every QPT adversary $\mathcal{A}$ there exists a QPT simulator $\mathcal{S}^{U_C}$ such that:*

$$\left|\Pr[\mathcal{A}(\rho_{(i)})=1]-\Pr\left[\mathcal{S}^{U_C}\left(|0\rangle^{\otimes|C|}\right)=1\right]\right|\leq \mathrm{negl}(n,|C|)\,.$$

We remark that reusability can be achieved in any number of ways: by providing a state which partially survives uses by the interpreter $\mathcal{J}$, by providing sufficiently many copies, or by providing a means of cloning the state. We prove impossibility of the above definition in any setting where the adversary receives two copies of the obfuscator output, even on identical inputs. This is automatically satisfied if the obfuscator provides multiple copies in order to satisfy reusability, or if the state is (even approximately) cloneable. The key new obstacle is to prove impossibility even though the functionality for both copies is *the same.*

To state the result, we define (in analogy to circuit-pair families and state-pair families) a *circuit family* to be an ensemble of distributions over circuits, and a *state family* to be an ensemble of distributions over states. A state family $\mathcal{C}'$ is said to be an obfuscation of a circuit family $\mathcal{C}$ if there exists a computable map $\mathcal{C}\to\mathcal{C}'$ assigning to each circuit a corresponding state, in a manner that satisfies Definition 6. With these definitions, we will prove the following theorem.

Theorem 5. *If quantum-secure one-way functions exist, then there exists a quantum circuit family $\mathcal{C}$ such that no state family is a reusable-black-box quantum obfuscation of $\mathcal{C}$.*

Since the full proof of Theorem 5 is somewhat lengthy and involved, we will first prove a simpler case, showing that quantum circuits cannot be obfuscated into quantum circuits, under any of the definitions considered so far—even the strongest one, Definition 5.) This corollary (stated below as Theorem 6) is arguably the most direct quantum generalization of the impossibility result of [6]. Once we have proved it, we will explain in detail how the proof should be adapted in order to achieve Theorem 5.

Theorem 6. *If quantum-secure one-way functions exist, then there exists a quantum circuit family $\mathcal{C}$ such that no quantum circuit family is a black-box obfuscation of $\mathcal{C}$.*

Proof. Let $\mathcal{O}$ be a black-box quantum obfuscator satisfying Definition 5, such that its outputs are classical bitstrings. Since these states are used to describe an efficiently implementable quantum computation, we can assume that these bitstrings are in fact quantum circuits under some particular encoding.

To construct the unobfuscatable circuit family, we will need a notion of combining the functionality of two quantum circuits into one.

Definition 7. *The* **combined quantum circuit** *of a finite collection $\{C_1, C_2, ..., C_k\}$ of n-qubit quantum circuits is the circuit that has two registers (a control register of $\log k$ qubits, and an input register of n qubits) and, controlled on the value of the first register, applies the respective quantum circuit to the input register.*

Notice that if each circuit C_i in the collection is polynomial size, and k is bounded by a polynomial in n, then the associated combined quantum circuit is also of polynomial size. We will denote the operation of combining circuits with #. For example, the combined circuit of two circuits C_1 and C_2 is denoted $C_1 \# C_2$.

Now recall the two circuits $C_{a,b}$ and $D_{a,b}$ from Section 3.3.1, as well as the circuit I_{2n}, which simply implements the identity operator on $2n$ qubits. Consider the combined quantum circuits $C_{a,b} \# D_{a,b}$ and $I_{2n} \# D_{a,b}$, sampled by selecting a and b uniformly at random from $\{0,1\}^n$. We again choose $C = C_{a,b}$ or $C = I_{2n}$, each with probability $1/2$, and ask the adversary to determine which is the case. Using the same reasoning as in the proof of Theorem 4 from Section 3.3.1, these combined quantum circuit distributions are indistinguishable from the perspective of any QPT simulator that is ignorant of a and b, and is given only black-box access to $U_{C \# D_{a,b}}$. On the other hand, unlike in the prior proof, it is not immediately apparent how to distinguish the two possibilities given a circuit description of $\mathcal{O}(C \# D_{a,b})$. Still, the idea is simple. Suppose we have two copies of the circuit. We can hard-wire the control register of one copy to implement $D_{a,b}$, and hard-wire the control register of the other copy to implement C. If we then run the first copy on the second, the result should be equivalent to implementing $D_{a,b}$ on input C, which will determine the nature of C and conclude the proof just as in Theorem 4.

Unfortunately, this idea does not work as stated, because the two circuit copies have the same size. Since they are functionally nontrivial, their input size is much smaller than their description, making it impossible to run one on the other. The core difficulty is that $D_{a,b}$ cannot be made large enough to universally execute circuits of size $|\mathcal{O}(C \# D_{a,b})| > p(|D_{a,b}|)$ where p is the running time of $\mathcal{O}$. This issue arises in the classical proof as well, and is resolved as follows. First, note that we could have $D_{a,b}$ simply provide a and b, thus offloading the gate-by-gate execution of C to the algorithm $\mathcal{A}$ in the black-box definition. Unfortunately, this would also provide the simulator $\mathcal{S}$ with a and b, enabling it to simulate $\mathcal{A}$. The resolution is to have $D_{a,b}$ provide *encryptions of* a and b, as well as a *quantum fully-homomorphic encryption (QFHE) oracle* for homomorphically applying the gates of C. We emphasize that the functionality and security of the QFHE oracle crucially depends on the obfuscation property; in particular, an actual QFHE scheme is not required for the proof.

Concretely, we prove the following Lemma, which is a quantum analogue of Lemma 3.6 (including Claim 3.6.1) from [7].

Lemma 7. *If quantum-secure one-way functions exist, then for each $n \in \mathbb{N}$ and $a, b \in \{0,1\}^n$, there exists a distribution $\mathcal{D}_{a,b}$ over circuits such that:*

1. *There exists a PPT algorithm that, given $n \in \mathbb{N}$ and $a,b \in \{0,1\}^n$, samples from $\mathcal{D}_{a,b}$;*
2. *There is a QPT algorithm $\mathcal{A}$ so that $C|a\rangle|0^n\rangle = |a\rangle|b\rangle$ implies $\mathcal{A}^{U_D}(C,1^n) = a$; this holds for all $n \in \mathbb{N}$, all $a,b \in \{0,1\}^n$, any $D \in \textbf{\textit{supp}}(\mathcal{D}_{a,b})$, and every n-qubit circuit C;*
3. *For any QPT S, $\Pr[S^{U_D}(1^n) = a] \leq \mathrm{negl}(n)$, where the probability is over $a,b \in \{0,1\}^n$, $D \sim \mathcal{D}_{a,b}$, and the measurements of S.*

Proof. The distribution $\mathcal{D}_{a,b}$ will be sampled by choosing $k, r \in_R \{0,1\}^{2n}$. The bitstring k is to serve as a private key for the IND-CCA1-secure symmetric-key quantum encryption scheme from Theorem 2. Each circuit $D \in \mathbf{supp}(\mathcal{D}_{a,b})$ will be a combination (again via #) of the following three circuits.

1. $E_{K,a}$; this outputs $\mathsf{Enc}_k(|a\rangle)$, executed with randomness r.
2. $\mathsf{Hom}_K(G,\rho)$; on input a gate description G and a state ρ, it outputs $[\mathsf{Enc}_k \circ G \circ \mathsf{Dec}_k](\rho)$.
3. $B_{k,a,b}$; on input ρ, it outputs $|a\rangle$ if $\mathsf{Dec}_k(\rho) = |b\rangle$ and $|0^n\rangle$ otherwise.

We remark that the Hom oracle requires randomness in order to re-encrypt the state. This is handled in the usual way: we expand the input in some register via a (quantum-secure) pseudo-random function; these exist by the assumption of quantum-secure one-way functions and Theorem 1.

Clearly, given a and b, $\mathcal{D}_{a,b}$ can be sampled efficiently by choosing k uniformly at random and outputting the combined quantum circuit $D_{k,a,b} := E_{k,a}\#\mathsf{Hom}_k\#B_{k,a,b}$. This establishes Property 1 from the Lemma. For Property 2, let $\mathcal{A}$ be the algorithm that, on input a quantum circuit C, (i.) uses the first two circuits comprising $D_{k,a,b}$ to homomorphically simulate C gate-by-gate on a, and then (ii.) plugs the final state into $B_{K,a,b}$.

To complete the proof of Lemma 7, we must verify Property 3, i.e., that no QPT simulator algorithm that has black-box access to each of the three algorithms comprising $D_{K,a,b}$ can discover a with non-negligible probability. This amounts to showing that

$$\left|\Pr[\mathcal{S}^{\mathsf{Hom}_k,\mathsf{Enc}_k}(\mathsf{Enc}_k(|0\rangle)) = 1] - \Pr\left[\mathcal{S}^{\mathsf{Hom}_k,\mathsf{Enc}_k}(\mathsf{Enc}_k(|a\rangle)) = 1\right]\right| \leq \mathrm{negl}(n)\,, \tag{3.5}$$

where the probabilities are over $k \in_R \{0,1\}^n$ and the measurement outcomes of $\mathcal{S}$. We proceed by contradiction, and assume that there's a QPT $\mathcal{S}$ that violates the claim.

First, we replace the responses to all of $\mathcal{S}$'s queries to the Hom_k oracle with encryptions of $|0^n\rangle$, and deploy a hybrid argument to show that the success probability of $\mathcal{S}$ is not significantly affected. To this end, consider the following hybrids of the computation of $\mathcal{S}$ on input $\mathsf{Enc}_k(|a\rangle)$: in the i-th hybrid, the first i oracle queries of $\mathcal{S}$ are answered using Hom_k, and the rest are answered with $\mathsf{Enc}_k(|0^n\rangle)$. Notice that any gap in distinguishing between the i and $i+1$st hybrid must be due to the $i+1$st query $\mathcal{S}$ makes to Hom_k. We can use this to create a CCA1 adversary $\mathcal{T}$ which breaks the encryption scheme, as follows. The QPT $\mathcal{T}$ simulates $\mathcal{S}$ and replies to its first i oracle queries by means of $\mathcal{T}$'s Enc_k and Dec_k oracles. Upon receiving the challenge ciphertext, $\mathcal{T}$ passes it to $\mathcal{S}$ as the response to its $i+1$st oracle query. Finally, $\mathcal{T}$ answers the remaining oracle queries of $\mathcal{S}$ with $\mathsf{Enc}_K(|0^n\rangle)$. We conclude that, if $\mathcal{S}$ violated (3.5), then $\mathcal{T}$ succeeds with non-negligible probability. This establishes (by contradiction) that we can replace the oracle queries of $\mathcal{S}$ with $\mathsf{Enc}_k(|0\rangle)$. With this replacement, $\mathcal{S}$ can distinguish an encryption of $|0^n\rangle$ from an encryption of $|a\rangle$, when given access to only an encryption oracle, which again contradicts IND-CCA1 security of the scheme. □

Now we are ready to describe the unobfuscatable family of quantum circuits and complete the proof of Theorem 6. First, for a fixed $a,b \in \{0,1\}^n$ we let $\mathcal{D}_{a,b}$ be the distribution over circuits constructed in Lemma 7. Then consider the following two distributions over circuits:

1. $\mathcal{F}_n$: Choose $a,b \in_R \{0,1\}^n$, sample a circuit $D_{a,b}$ from $\mathcal{D}_{a,b}$ and output $C_{a,b}\#D_{a,b}$
2. $\mathcal{G}_n$: Choose $a,b \in_R \{0,1\}^n$, sample a circuit $D_{a,b}$ from $\mathcal{D}_{a,b}$ and output $I_{2n}\#D_{a,b}$

By Property 2 of Lemma 7 there exists an algorithm $\mathcal{A}$ that, on input $\mathcal{O}(C_0)$, accepts if C_0 was sampled from $\mathcal{F}_n$ and rejects if C was sampled from $\mathcal{G}_n$. Thus there exists a constant α and a negligible function ε_1 so that:

$$\left|\Pr\left[\mathcal{A}(\mathcal{O}(\mathcal{F}_n))=1\right]-\Pr\left[\mathcal{A}\left(\mathcal{O}(\mathcal{G}_n)\right)=1\right]\right|\geq\alpha-\varepsilon_1(n)\,.$$

On the other hand, by Property 3 of Lemma 7, we know that for every QPT S there exists some negligible function ε_2 so that:

$$\left|\Pr\left[\mathcal{S}^{\mathcal{F}_n}\left(|0\rangle^{\otimes n}\right)=1\right]-\Pr\left[\mathcal{S}^{\mathcal{G}_n}\left(|0\rangle^{\otimes n}\right)=1\right]\right|\leq\varepsilon_2(n)\,.$$

We conclude that the circuit family formed by taking the union of $\mathcal{F}_n$ with $\mathcal{G}_n$ (and assigning them each equal probability) is an unobfuscatable circuit family. □

We now return to the proof of Theorem 5, and show how to extend the above proof to the case where the obfuscator outputs reusable quantum states.

Proof. (of Theorem 5) Our proof will still use the same distribution $\mathcal{D}_{a,b}$ over circuits, which were provided by Lemma 7 and described above, but with some slight modifications. The goal will still be to take two copies of $\mathcal{O}(C_0)$ for any circuit C_0 sampled from that distribution, and give an algorithm $\mathcal{A}$ that can "execute one copy on the other." This will enable us to distinguish if C_0 is from the distribution $\mathcal{F}_n$, or the distribution $\mathcal{G}_n$ (just as above), a task which is impossible with only black-box access.

However, executing one copy of $\mathcal{O}(C_0)$ on another is now somewhat more complicated, due to the fact that we no longer have explicit circuit descriptions in hand, and must instead use the interpreter $\mathcal{J}$ (with some register initialized to $\mathcal{O}(C_0)$) whenever we want to run C_0. To do this, we will need to describe a new distribution $\mathcal{D}'_{a,b}$ of circuits, closely related to the distribution $\mathcal{D}_{a,b}$ from Lemma 7.

Attempt 1. To warm up, a first attempt at describing $\mathcal{A}$ and the modified circuits $D'_{a,b}$ from the distribution $\mathcal{D}'_{a,b}$ is as follows. First, we simply define $D'_{a,b}$ to be a composition of circuits which simply output both a and b. Set $C_0 = C\#D'_{a,b}$, and suppose that inputting $|0\rangle$ in the first register executes the first circuit in the combination, while $|1\rangle$ executes the second circuit in the combination. The algorithm $\mathcal{A}$ is in possession of two copies of $\mathcal{O}(C_0)$. It performs:

1. run $\mathcal{J}(\mathcal{O}(C_0)\otimes|1\rangle|0^n\rangle)$ to retrieve $|a\rangle|b\rangle$ (by functional equivalence of $\mathcal{O}$);
2. run $\mathcal{J}(\mathcal{O}(C_0)\otimes|0\rangle|a\rangle)$ (now using the other copy of $\mathcal{O}(C_0)$);
3. compare the result to $|b\rangle$.

This does exactly what we want, except of course that the black-box simulator $\mathcal{S}$ will also be able to retrieve a and perform this experiment. Our valiant attempt failed.

Attempt 2. Undeterred, we now try a more sophisticated approach, returning to the idea of encryption and homomorphic execution. We now ask that (as before) $\mathcal{D}'_{a,b}$ outputs an encryption $\mathsf{Enc}(a)$ of a (when given the flag A), implements a Hom oracle (when given the flag H), and checks if the input is $\mathsf{Enc}(b)$ (when given the flag B). We again set $C_0 = C\#D'_{a,b}$ and give $\mathcal{A}$ two copies of $\mathcal{O}(C_0)$; for convenience we denote them $\mathcal{O}(C_0)$ and $\mathcal{O}(C_0)'$. Now, $\mathcal{A}$ performs:

1. run $\mathcal{J}\left(\mathcal{O}(C_0)\otimes[|1A\rangle\otimes|0^n\rangle]\right)$ to retrieve $\mathsf{Enc}(a)$;
2. run $\mathcal{J}\left(\mathcal{O}(C_0)\otimes[|1H\rangle\otimes\mathcal{O}(C_0)'\otimes|0\rangle\otimes\mathsf{Enc}(a)]\right)$ to "homomorphically run C";
3. run $\mathcal{J}\left(\mathcal{O}(C_0)\otimes[|1B\rangle\otimes\rho]\right)$ where ρ is the output of the previous step; output the result.

The first and last step are largely self-explanatory: we start with the encryption of a, and check at the end that we have the encryption of b. What happened in the second step? We tried to homomorphically evaluate[5] C on $\mathsf{Enc}(a)$. By functional equivalence of $\mathcal{O}$, we executed the first copy of C_0 on input $|1H\rangle\otimes\mathcal{O}(C_0)'\otimes\mathsf{Enc}(a)$; this specifies that $D'_{a,b}$ should run the Hom oracle with input $\mathcal{O}(C_0)'\otimes\mathsf{Enc}(a)$. To make this sensible, we can redefine Hom to accept two registers, and homomorphically evaluate the

[5] Selecting C was done by passing the bit flag $|0\rangle$ into the control register.

appropriate circuit of $\mathcal{J}$; the result is that, whenever Hom is called on $\mathcal{O}(C) \otimes \mathsf{Enc}(z)$ for a circuit C and state z, the output is $\mathsf{Enc}(C(z))$.

This attempts looks like it succeeds, but there is a disastrous flaw: Hom must now accept inputs with at least as many qubits as $\mathcal{O}(C_0)$, which is significantly bigger than the circuit description allowed for Hom itself (since it is a part of $D'_{a,b}$ and hence also of C_0).

Attempt 3. In the final attempt, we will repair the flaw of Attempt 2. The key is to again offload some of the execution, but this time from the Hom oracle to the main algorithm $\mathcal{A}$. More precisely, we will expand step (2) in Attempt 2, and execute it gate-by-gate. In this iteration, Hom is back to its original version, and is used only to apply two-qubit gates. It accepts n input qubits, decrypts them, applies the desired gate (as specified in another register), and then re-encrypts. In addition, will also expand $D'_{a,b}$ to provide an Enc oracle (when given the flag E); we can do this for free, by equation (3.5) in Lemma 7. The final algorithm $\mathcal{A}$ will proceed as follows. Here we have let m denote the number of qubits of $\mathcal{O}(C_0)'$, and we let J_m be the circuit of $\mathcal{J}$ for executing m-qubit obfuscated states.

1. run $\mathcal{J}\big(\mathcal{O}(C_0) \otimes [|1E\rangle \otimes \mathcal{O}(C_0)'_{(k)}]\big)$ for all $k \in [m]$, to encrypt all qubits of $\mathcal{O}(C_0)'$;
2. run $\mathcal{J}\big(\mathcal{O}(C_0) \otimes [|1A\rangle \otimes |0^n\rangle]\big)$ to retrieve $\mathsf{Enc}(a)$;
3. let $j = 0$ and let $\sigma_j := \mathsf{Enc}(a)$;
4. let G_j be the jth gate in the description of J_m;
   (a) let s,t be the qubits G_j acts on; assume s is a qubit of $\mathcal{O}(C_0)'$ and t is a qubit of σ_j;[6]
   (b) set $\sigma_{j+1} = \mathcal{J}\big(\mathcal{O}(C_0) \otimes [|1H\rangle \otimes \big|G_j, s, t\big\rangle \otimes \mathsf{Enc}(\mathcal{O}(C_0)'_{(s)}) \otimes \sigma_j]\big)$;[7]
   (c) if $j = |J_m|$, continue; otherwise increment j by 1 and go to step 4.
5. run $\mathcal{J}\big(\mathcal{O}(C_0) \otimes [|1B\rangle \otimes \sigma_j]\big)$ and output the result.

A few remarks are in order. First, the reusability of the state $\mathcal{O}(C)$ was crucial in our ability to repeatedly execute the Hom oracle in step 4.(b). Second, one checks that by the functional equivalence of the obfuscator and the definition of the Hom oracle, if $C_0 = C_{a,b} \# D'_{a,b}$ then the state σ_j when step 5. is reached will indeed be $\mathsf{Enc}(b)$. Third, note that the "compactness" issue of Hom from Attempt 2 has been resolved, and the input to C_0 in step 4.(b) is now of size n (plus a constant.)

Finally, despite all of the difficulties in defining the algorithm $\mathcal{A}$ appropriately, the hardness of the corresponding search problem for the black-box simulator $\mathcal{S}$ is essentially unchanged from the proof of Theorem 6. The only difference is that $D'_{a,b}$ now also provides an encryption oracle; the encryption scheme we selected is certainly secure in this setting.

To finish the proof, we again build a circuit family by choosing $C_{a,b} \# D'_{a,b}$ or $I_{2n} \# D'_{a,b}$, each with equal probability, for random a and b. By the above arguments, this circuit family is unobfuscatable. This concludes the proof of Theorem 5. □

4 Quantum indistinguishability obfuscation

(Gorjan: This is a copy of the old section (which is still below.) In this copy, I am trying to do things using CPTP maps from the get-go. The basic issue is that I think it's weird to only ask for obfuscation of unitary circuits. If the goal is to obfuscate actual quantum computations, then any discarding or measurement should be obfuscated as well—otherwise they are leaking information about the computation! Further, I am using ensembles rather than individual circuits, and I'm trying to simplify the definitions by first defining a "translator" (i.e., an obfuscator with no obfuscation condition) and then tacking on different obfuscation conditions.)

In this section, we analyze a different definition of quantum obfuscation, motivated by classical definitions established by Goldwasser and Rothblum [24]. As opposed to the black-box approach, these

[6] This assumption is only made for simplicity of the algorithm description; the other possibilities are similar.

[7] The notation $\mathcal{O}(C_0)'_{(s)}$ is meant to indicate that only the s-th qubit of that state is to be placed in the input register.

definitions assess the quality of an obfuscation in relative terms, e.g., as compared to other functionally-equivalent circuits (or, in the quantum case, states).

4.1 Ensembles of circuits and states

(Gorjan: It might be better to put this near the start, and use it for both VBB and IO sections.) We first set down some basic terminology for dealing with infinite collections of circuits and states. We take the point of view that any quantum circuit C can include unitary gates, as well as measurement gates and instructions for discarding qubits. We overload notation so that C denotes both the circuit itself (i.e., a classical description of a set of wires and gates) as well as the CPTP map that the circuit implements.

A **circuit ensemble** is an infinite collection $\{C_s : s \in S\}$ of circuits indexed by some set $S \subset \{0,1\}^*$, such that both the size and the number of qubits of C_s are bounded by some fixed polynomial function of $|s|$. We will denote ensembles by italicized capital letters (optionally, with the indexing set in the subscript.) Elements of the ensemble will be denoted by the corresponding non-italicized capital letter (optionally, with the index in the subscript.) So, for example, we may write $\mathcal{C}_S := \{C_s : s \in S\}$. A **uniform circuit ensemble** is a circuit ensemble together with a deterministic Turing Machine which, on input $s \in S$, outputs a classical description of C_s. Given two circuit ensembles $\mathcal{C}_S$ and $\mathcal{D}_S$, we say that $\mathcal{C}_S$ is functionally equivalent to $\mathcal{D}_S$ if the circuits themselves are functionally equivalent, i.e., if $\|C_s - D_s\|_\diamond \leq \mathrm{negl}(|s|)$ for all $s \in S$.

Similarly, we define a **state ensemble** to be an infinite collection $\{\rho_x : x \in X\}$ of density operators indexed by some set $X \subset \{0,1\}^*$, such that $\rho_x \in \mathfrak{D}(\mathcal{H}_{p(|x|)})$ where p is bounded by some fixed polynomial function. We remark that a circuit ensemble is a special case of a state ensemble, where each state is a classical string describing wires, gates, and so on. A **uniform state ensemble** will be a state ensemble $\{\rho_x : x \in X\}$ together with a uniform circuit ensemble $\{C_x : x \in X\}$ such that $\rho_x = C_x|0^m\rangle$ for appropriately chosen m. We remark that a QPT is defined by choosing a uniform circuit ensemble, and that the set of possible outputs of a QPT are a uniform state ensemble. In particular, if $\mathcal{S}$ is a state ensemble and $\mathcal{A}$ is a QPT, then (ignoring some uninteresting bookkeeping) we may write $\mathcal{A}(\mathcal{S})$ to denote the state ensemble that results from running $\mathcal{A}$ on inputs from $\mathcal{S}$. Note that if $\mathcal{S}$ is uniform, then $\mathcal{A}(\mathcal{S})$ is also uniform.

Next, we consider three different notions of distinguishability for state ensembles, in order of decreasing power. Let $\mathcal{R} = \{\rho_x : x \in X\}$ and $\mathcal{S} = \{\sigma_x : x \in X\}$ be two state ensembles. We say that $\mathcal{R}$ and $\mathcal{S}$ are **perfectly indistinguishable** if $\rho_x = \sigma_x$ for every $x \in X$. While this is a fairly unnatural notion for general quantum states, it is more reasonable (and is easy to test for individual cases) if ρ_x and σ_x happen to all be classical. A weaker notion is **statistical indistinguishability**, which demands that $\|\rho_x - \sigma_x\|_{\mathrm{tr}} \leq \mathrm{poly}(|x|)$ for all but finitely many $x \in X$. We say that $\mathcal{R}$ and $\mathcal{S}$ are **computationally indistinguishable** if for every QPT $\mathcal{A}$,

$$|\Pr[\mathcal{A}(\rho_x) = 1] - \Pr[\mathcal{A}(\sigma_x) = 1]| \leq \mathrm{negl}(|x|)$$

for all but finitely many x; here the probabilities are taken over the coins and measurements of $\mathcal{A}$. One may also consider a non-uniform version of the above definition, in which $\mathcal{A}$ is a non-uniform circuit family which is allowed access to an auxiliary state ensemble $\{\xi_x : x \in X\}$. All of our results hold for both the uniform and the non-uniform setting; we will focus on the uniform setting for convenience.

4.2 Definitions, and an application

We begin with the notion of a quantum indistinguishability obfuscator. As before, the interpreter and the obfuscated state must be efficient, while the obfuscation algorithm itself might not be. We also assume that all "interpreter" algorithms $\mathcal{J}$ have two registers: an advice register (where the obfuscated state is to be inserted), and an input register (where the input is to be inserted.) It will thus be convenient to write, e.g., $\mathcal{J}_\rho$ for the CPTP map corresponding to executing $\mathcal{J}$ with the advice register initialized to the state ρ.

Definition 8. *A **quantum translator** is a quantum algorithm $\mathcal{O}$ and a QPT $\mathcal{J}$ such that whenever $\mathcal{C}$ is a circuit ensemble and $C \in \mathcal{C}$ is an n-qubit circuit,*

1. *(polynomial slowdown)* $\mathcal{O}(C)$ *has at most* $\mathrm{poly}(n)$ *qubits;*
2. *(functional equivalence)* $\left\|\mathcal{J}_{\mathcal{O}(C)} - C\right\|_\diamond \leq \mathrm{negl}(n)$*;*

Definition 9. ***A quantum statistical (resp., computational) indistinguishability obfuscator*** *is a quantum translator* $(\mathcal{O},\mathcal{J})$ *such that whenever* $\mathcal{C}_S$ *and* $\mathcal{D}_S$ *are functionally-equivalent circuit ensembles with* $|C_s| = |D_s|$ *for all* $s \in S$*, then* $\mathcal{O}(\mathcal{C}_S)$ *and* $\mathcal{O}(\mathcal{D}_S)$ *are statistically (resp., computationally) indistinguishable.*

Note that an obfuscator which simply outputs circuit descriptions is included as a special case of the above; in that case, $\mathcal{O}(C)$ is always a quantum circuit, and $\mathcal{J}$ is a universal circuit which executes $\mathcal{O}(C)$ on the state given in the input register. One may also define a quantum perfect-indistinguishability obfuscator, where the obfuscated states $\mathcal{O}(C_1)$ and $\mathcal{O}(C_2)$ are identical. We remark that the condition of equal length can be relaxed to any fixed polynomial (e.g., $|C_1|$ can be of length at most $|C_2|^2$.)

Next, we consider the question of functional equivalence for state ensembles. If we fix a quantum translator $(\mathcal{O},\mathcal{J})$, then the QPT $\mathcal{J}$ defines a notion of implementing functionality via states. We then say that two state ensembles $\mathcal{R}_X$ and $\mathcal{S}_X$ are functionally equivalent if

$$\left\|\mathcal{J}_{\rho_x} - \mathcal{J}_{\sigma_x}\right\|_\diamond \leq \mathrm{negl}(|x|)$$

for all but finitely many $x \in X$, $\rho_x \in \mathcal{R}_X$ and $\sigma_x \in \mathcal{S}_X$. This allows us to consider the relative strength of obfuscation in one state ensemble versus another, as follows.

Definition 10. ***A quantum statistical (resp., computational) best-possible obfuscator*** *is a quantum translator* $(\mathcal{O},\mathcal{J})$ *such that for any QPT* $\mathcal{A}$ *there exists a QPT* $\mathcal{S}$ *with the following property: for every circuit ensemble* $\mathcal{C}_X$ *and any state ensemble* $\mathcal{R}_X$ *which is functionally-equivalent to* $\mathcal{O}(\mathcal{C}_X)$ *and has same-size states*[8]*, we have that* $\mathcal{A}(\mathcal{O}(\mathcal{C}_X))$ *and* $\mathcal{S}(\mathcal{R}_X)$ *are statistically (resp., computationally) indistinguishable.*

This definition captures the relative "leakage" of the obfuscated state ensemble: among all functionally-equivalent state ensembles, it leaks the least.

(Gorjan: Should this definition somehow vary over the interpreters $\mathcal{J}$ as well?)

(Gorjan: — Stopped editing here. —)

5 Quantum indistinguishability obfuscation

In this section, we analyze a different definition of quantum obfuscation, motivated by classical definitions established by Goldwasser and Rothblum [24]. As opposed to the black-box approach, these definitions assess the quality of an obfuscation in relative terms, e.g., as compared to other functionally-equivalent circuits (or, in the quantum case, states).

5.1 Definitions, and an application

We begin with the notion of a quantum indistinguishability obfuscator. As before, the interpreter and the obfuscated state must be efficient, while the obfuscation algorithm itself might not be.

Definition 11. ***A quantum indistinguishability obfuscator*** *is a quantum algorithm* $\mathcal{O}$ *and a QPT* $\mathcal{J}$ *such that whenever* C *is an* n*-qubit quantum circuit* C*,* $\mathcal{O}(C)$ *is an* m*-qubit quantum state satisfying*

1. *(polynomial slowdown)* $m = poly(n,|C|)$*.*
2. *(functional equivalence)* $\left\|\mathcal{J}(\rho\otimes\cdot) - C\cdot C^\dagger\right\|_\diamond \leq \mathrm{negl}(n,|C|)$*;*
3. *(indistinguishability) if circuits* C_1, C_2 *satisfy* $|C_1| = |C_2|$ *and* $\|U_{C_1} - U_{C_2}\| \leq \mathrm{negl}(n)$*, then the states* $\mathcal{O}(C_1)$ *and* $\mathcal{O}(C_2)$ *are indistinguishable.*

[8] meaning that, for each $x \in X$, the corresponding states in the two ensembles have the same number of qubits

Note that an obfuscator which simply outputs circuit descriptions is included as a special case of the above; in that case, $\mathcal{O}(C)$ is always a quantum circuit, and $\mathcal{J}$ is a universal circuit which executes $\mathcal{O}(C)$ on the state given in the input register. We consider three versions of the notion of indistinguishability in the third condition above. These are perfect indistinguishability (i.e., the states are identical), statistical indistinguishability (i.e., the states have negligible trace distance), and computational indistinguishability (i.e., the states are not distinguishable by QPT adversaries.) We remark that the condition of equal length can be relaxed to any fixed polynomial (e.g., $|C_1|$ can be of length at most $|C_2|^2$.)

We also define a notion of **quantum best-possible obfuscator**. This is an algorithm $\mathcal{O}$ and QPT $\mathcal{J}$ precisely as in Definition 9, but now with condition (3) replaced with the following:

3. *(best-possible) for every pair C_1, C_2 satisfying $|C_1| = |C_2|$ and $\|U_{C_1} - U_{C_2}\| \leq \text{negl}(n)$ and every QPT $\mathcal{A}$, there exists a QPT $\mathcal{S}$ such that*

$$\left|\Pr[\mathcal{A}(\mathcal{O}(C_1)) = 1] - \Pr\left[\mathcal{S}\left(C_2\right) = 1\right]\right| \leq \text{negl}(n)\,.$$

The intuition behind the above definition is the following: any information $\mathcal{A}(\mathcal{O}(C_1))$ that is "leaked" by the obfuscation $\mathcal{O}(C_1)$ can actually be recovered from any functionally equivalent, similarly-sized circuit C_2. In this sense, the obfuscation $\mathcal{O}(C_1)$ leaks the least amount of information possible. When using the best-possible definition, we will frequently refer to $\mathcal{A}$ as a "learner" (and even denote it by $\mathcal{L}$) to emphasize its role in the intuition. While Definition 9 is not as naturally motivated, it is easier to formulate. As we will later show, the two definitions are equivalent for efficient obfuscators. We will thus be free to work with the definition that is more natural for whatever task is at hand.

Before proving an impossibility result, we mention an application of indistinguishability obfuscation; this application survives the impossibility result, and is motivated by analogous application of classical indistinguishability obfuscators.

Application: quantum witness encryption. The classical idea of witness encryption was first studied in [19]; its connection to indistinguishability obfuscation was first considered in [18]. In the quantum case, we set up the problem as follows. Suppose Alice wishes to encrypt a quantum state ρ, but not to a particular key or for a particular person; instead, the encryption is tied to a challenge question, and anyone that can answer the question correctly can decrypt the plaintext.

More formally, we consider a QMA language L, and would like to enable Alice to encrypt her state ρ using a particular problem instance x. If x is a "yes" instance, then we'd like to allow Bob, who holds a witness state, to be able to decrypt Alice's message. Likewise, if x is a no instance, we demand that no QPT algorithm can distinguish between encryptions of any two quantum states on the same number of qubits. Interestingly, the definition says nothing about the case where x is a yes instance but a witness is not known. While this may seem counterintuitive, the classical primitive has a number of natural applications (e.g., public-key encryption and identity encryption, see [19]).

We now show that Witness Encryption for QMA is possible, assuming the existence of a quantum computational indistinguishability obfuscator. It is well-known (see, e.g., [25]) that QMA contains languages L which have a poly-time uniform circuit family $\mathcal{C}_L$ with completeness $1-2^{-\Omega(n)}$ and soundness $2^{-\Omega(n)}$. Given an instance x of such a language L, and a quantum state ρ, consider the quantum circuit $Q_{x,\rho}(\sigma)$, which runs the appropriate circuit from $\mathcal{C}_L$ and outputs ρ on accept, and $|0^n\rangle$ otherwise. We claim the computational-indistinguishability obfuscation $\mathcal{O}(Q_{x,\rho})$ is a valid witness encryption for L and ρ. Correctness of decryption is clear, since the ability to provide a valid witness for L allows Bob to use $\mathcal{O}(Q_{x,\rho})$ to obtain Alice's state ρ. Of course, if x is not in L, then no witness will suffice; more precisely, $\|Q_{x,\rho_1} - Q_{x,\rho_2}\| \leq 2^{-\Omega(n)}$ for any two quantum states ρ_1, ρ_2 on the same number of qubits. By the indistinguishability condition of Definition 9, the obfuscation of Q_{x,ρ_1} will be computationally indistinguishable from the obfuscation of Q_{x,ρ_2}.

5.2 Equivalence of indistinguishability and best-possible

In what follows, for the sake of simplicity we omit the perfect, statistical, and classical variants of the definitions; one can arrive at these versions simply by replacing quantum indistinguishability of the relevant ensembles to one of the other notions. We say that two quantum circuit families $\mathcal{C}'$ and $\mathcal{C}''$ are equivalent if they consist of functionally equivalent circuits of the same size; more precisely, for every n, $|\mathcal{C}'_n| = |\mathcal{C}''_n| = 1$ and $|C'_n| = |C''_n|$ and $U_{C'_n} = U_{C''_n}$. We begin with the following straightforward observation.

Proposition 1. *There exists an inefficient algorithm which is a perfect indistinguishability obfuscator for all quantum circuits, and whose outputs are quantum circuits.*

Proof. The algorithm simply picks the lexicographically first circuit which implements the same unitary as the given circuit. Iterating lexicographically through all polynomial-size circuits can be done in PSPACE, and equivalence-checking can be done in QMA $\subset$ PSPACE as well. □

Next, we show that best-possible implies indistinguishability, for both efficient and inefficient obfuscators. In our proofs, we will need to explicitly refer to uniform circuit families of circuits, as defined above.

Proposition 2. *If $\mathcal{O}$ is a best-possible quantum obfuscator for a circuit family $\mathcal{C}$, then it is also a quantum indistinguishability obfuscator for $\mathcal{C}$.*

Proof. Let $\mathcal{C}'$ and $\mathcal{C}''$ be uniform equivalent subfamilies of $\mathcal{C}$, and let $\mathcal{L}$ be the trivial learner that simply implements the identity operator. By the best-possible property, there is a simulator $\mathcal{S}$ such that $\mathcal{S}(\mathcal{C}'')$ is quantum indistinguishable from $\mathcal{L}(\mathcal{O}(\mathcal{C}')) = \mathcal{O}(\mathcal{C}')$. By the same property, we also have that $\mathcal{S}(\mathcal{C}'')$ is quantum indistinguishable from $\mathcal{L}(\mathcal{O}(\mathcal{C}'')) = \mathcal{O}(\mathcal{C}'')$. By the transitivity property of indistinguishability, it follows that $\mathcal{O}(\mathcal{C}')$ is indistinguishable from $\mathcal{O}(\mathcal{C}'')$. □

The other direction requires efficiency of the obfuscator.

Proposition 3. *If $\mathcal{O}$ is an efficient quantum indistinguishability obfuscator for a circuit family $\mathcal{C}$, then it is also an efficient quantum best-possible obfuscator for $\mathcal{C}$.*

Proof. Let $\mathcal{C}'$ and $\mathcal{C}''$ be equivalent subfamilies of $\mathcal{C}$, and let $\mathcal{L}$ be a learner whose output on $\mathcal{C}'$ is the ensemble $\mathcal{L}(\mathcal{O}(\mathcal{C}'))$. We define a simulator by setting $\mathcal{S} = \mathcal{L} \circ \mathcal{O}$; its output on $\mathcal{C}''$ is then the ensemble $\mathcal{L}(\mathcal{O}(\mathcal{C}''))$. Since the ensembles $\mathcal{O}(\mathcal{C}')$ and $\mathcal{O}(\mathcal{C}'')$ are quantum indistinguishable, so are their images under $\mathcal{L}$. □

5.3 Impossibility of statistical obfuscators

In this section, we show that efficient perfect or statistical indistinguishability obfuscation is impossible. We begin by recalling the following computational problems and corresponding completeness results.

Problem 1. Identity Check.
Input: an n-qubit quantum circuit C and parameters a,b so that $b - a \geq 1/poly(n)$.
Promise: $\min_\alpha \|U - e^{i\alpha} I\|$ *is less than a or greater than b.*
Output: YES in the former case and NO in the latter.

Theorem 8. *The problem Identity Check is coQMA-complete [28].*

Given an m-qubit state ρ and indices $l, k \geq 0$, we let $\mathrm{Tr}_{l,k}[\rho]$ denote the result of tracing out qubits l through k of ρ. We adopt the convention that nothing is traced out (i.e., $\mathrm{Tr}_{l,k}[\rho] = \rho$) if $l > m$.

Definition 12. Quantum State Distinguishability
Input: m-qubit quantum circuits C_0 and C_1, positive integer $k \leq m$ and parameters a, b with $a < b^2$.
Promise: let $\rho_i = \mathrm{Tr}_{(k+1,m)}[C_i|0^m\rangle\langle 0^m|C_i^\dagger]$; then $\|\rho_0 - \rho_1\|_{\mathrm{tr}}$ is less than a or greater than b.
Output: YES in the former case and NO in the latter.

Theorem 9. *The problem Quantum State Distinguishability is QSZK-complete [32].*

As a matter of fact, we will only need the containment in QSZK for the Quantum State Distinguishability problem. We prove the following.

Theorem 10. *If there exists a polynomial-time quantum statistical indistinguishability obfuscator, then coQMA is contained in QSZK.*

Proof. We will actually show coQMA $\subset$ BQP$^{\text{QSZK}}$; since BQP is contained in QSZK, the result will follow. Let a and b satisfy $b-a = 1/\text{poly}(n)$. We will solve Identity Check using a subroutine that solves Quantum State Distinguishability.

Let C be the input, i.e., a classical description of an n-qubit quantum circuit. Create an identity circuit D with an equal number of inputs as C, and of equal length to C. Let O_C be a circuit that initializes a register with the classical state $|C\rangle$ containing the classical description of C, and applies the circuit of $\mathcal{O}$ which corresponds to the input length $|C|$. Likewise, let O_D be a circuit that initializes a register with the classical state $|D\rangle$ containing the classical description of D, and applies the circuit of $\mathcal{O}$ which corresponds to the input length $|D| = |C|$. Note that, after tracing out ancillas, the outputs of these circuits are given by

$$\text{Tr}_{\text{anc.}}\left[O_C|0\rangle\langle 0|O_C^\dagger\right] = \mathcal{O}(C) \qquad \text{and} \qquad \text{Tr}_{\text{anc.}}\left[O_D|0\rangle\langle 0|O_D^\dagger\right] = \mathcal{O}(D)\,.$$

Now apply the subroutine for solving quantum state distinguishability to the pair (O_C, O_D). If it says "close", we output YES; otherwise we output NO. Let's show that this has solved (a,b)-identity-check. Note that the states $\mathcal{O}(C)$ and $\mathcal{O}(D)$ must have the same number of qubits, and denote that number by m.

- **completness.** In this case, the obfuscated states satisfy $\|\mathcal{O}(C) - \mathcal{O}(D)\|_{\text{tr}} \leq \alpha$. By the definition of the induced trace norm, this implies that $\|\mathcal{J}^n_{\mathcal{O}(C)} - \mathcal{J}^n_{\mathcal{O}(D)}\|_\diamond \leq \alpha$. By functional equivalence for C and D and the triangle inequality, it follows that $\|U_C - U_D\|_\diamond = \|U_C - I\|_\diamond \leq \alpha$, as desired.
- **soundness.** In this case, the obfuscated states satisfy $\|\mathcal{O}(C) - \mathcal{O}(D)\|_{\text{tr}} \geq \beta$. We claim that this implies $\|U_C - U_D\|_\diamond > b$. Suppose this is not the case, i.e., that these operators are in fact close; then by the indistinguishability property, it would follow that $\mathcal{O}(C)$ and $\mathcal{O}(D)$ are close as well, a contradiction.

The above amounts to a BQP$^{\text{QSZK}}$ protocol for a coQMA-hard problem, thus placing coQMA in QSZK. □

6 Discussion

We briefly list some open questions, and leave more detailed discussions for a future version.

- Can you achieve single-copy quantum black-box obfuscation with quantum states, either for classical or quantum functionality? Is information-theoretic security possible if we give up on reusability?
- Can our black-box impossibility proofs be adapted to show that the state *must be consumable*, even if only partly? Is it conceivable that, if we restrict to only classical functionality, that the advice state is *perfectly preserved* with each execution?
- Are there quantum versions of the recent classical breakthroughs for computational indistinguishability obfuscation? Can we achieve quantum circuit-to-circuit obfuscation under that condition? Which of the classical applications carry over to the quantum setting?
- Our impossibility proofs apply to unitary circuits, and thus trivially also to circuits that include measurements, or arbitrary CPTP maps. If we *require* the ability to obfuscate such computations as well, are stronger impossibility results possible?

References

[1] Scott Aaronson. Ten semi-grand challenges for quantum computing theory. http://www.scottaaronson.com/writings/qchallenge.html, July 2005. Retrieved 09/15.

[2] Scott Aaronson. Quantum copy-protection and quantum money. In *Computational Complexity, 2009. CCC'09. 24th Annual IEEE Conference on*, pages 229–242. IEEE, 2009.

[3] 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. ACM, 2012.

[4] Gorjan Alagic, Stacey Jeffery, and Stephen Jordan. Circuit obfuscation using braids. In *Proceedings of TQC 2014*, volume 27, pages 141–160, 2014.

[5] Gorjan Alagic, Anne Broadbent, Bill Fefferman, Tommaso Gagliardoni, Christian Schaffner, and Michael StJules. Computational security for quantum encryption. *To appear.*, 2015.

[6] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. In *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '01, pages 1–18, London, UK, UK, 2001. Springer-Verlag. ISBN 3-540-42456-3. URL http://dl.acm.org/citation.cfm?id=646766.704152.

[7] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. *J. ACM*, 59(2):6:1–6:48, May 2012. ISSN 0004-5411. doi:10.1145/2160158.2160159. URL http://doi.acm.org/10.1145/2160158.2160159.

[8] Boaz Barak, Sanjam Garg, Yael Tauman Kalai, Omer Paneth, and Amit Sahai. Protecting obfuscation against algebraic attacks. In PhongQ. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 221–238. Springer Berlin Heidelberg, 2014. ISBN 978-3-642-55219-9. doi:10.1007/978-3-642-55220-5_13. URL http://dx.doi.org/10.1007/978-3-642-55220-5_13.

[9] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh V. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997. doi:10.1137/S0097539796300933. URL http://dx.doi.org/10.1137/S0097539796300933.

[10] CharlesH. Bennett, Gilles Brassard, Seth Breidbart, and Stephen Wiesner. Quantum cryptography, or unforgeable subway tokens. In David Chaum, RonaldL. Rivest, and AlanT. Sherman, editors, *Advances in Cryptology*, pages 267–275. Springer US, 1983. ISBN 978-1-4757-0604-8. doi:10.1007/978-1-4757-0602-4_26. URL http://dx.doi.org/10.1007/978-1-4757-0602-4_26.

[11] Nir Bitansky, Ran Canetti, Henry Cohn, Shafi Goldwasser, Yael Tauman Kalai, Omer Paneth, and Alon Rosen. The impossibility of obfuscation with auxiliary input or a universal simulator. In JuanA. Garay and Rosario Gennaro, editors, *Advances in Cryptology CRYPTO 2014*, volume 8617 of *Lecture Notes in Computer Science*, pages 71–89. Springer Berlin Heidelberg, 2014. ISBN 978-3-662-44380-4. doi:10.1007/978-3-662-44381-1_5. URL http://dx.doi.org/10.1007/978-3-662-44381-1_5.

[12] Dan Boneh and Mark Zhandry. Multiparty key exchange, efficient traitor tracing, and more from indistinguishability obfuscation. In JuanA. Garay and Rosario Gennaro, editors, *Advances in Cryptology CRYPTO 2014*, volume 8616 of *Lecture Notes in Computer Science*, pages 480–499. Springer Berlin Heidelberg, 2014. ISBN 978-3-662-44370-5. doi:10.1007/978-3-662-44371-2_27. URL http://dx.doi.org/10.1007/978-3-662-44371-2_27.

[13] Zvika Brakerski and GuyN. Rothblum. Virtual black-box obfuscation for all circuits via generic graded encoding. In Yehuda Lindell, editor, *Theory of Cryptography*, volume 8349 of *Lecture Notes in Computer Science*, pages 1–25. Springer Berlin Heidelberg, 2014. ISBN 978-3-642-54241-1. doi:10.1007/978-3-642-54242-8_1. URL http://dx.doi.org/10.1007/978-3-642-54242-8_1.

[14] Anne Broadbent and Stacey Jeffery. Quantum homomorphic encryption for circuits of low T-gate complexity. *Crypto 2015 (to appear)*, December 2015.

[15] Ran Canetti and RonnyRamzi Dakdouk. Obfuscating point functions with multibit output. In Nigel Smart, editor, *Advances in Cryptology EUROCRYPT 2008*, volume 4965 of *Lecture Notes in Computer Science*, pages 489–508. Springer Berlin Heidelberg, 2008. ISBN 978-3-540-78966-6. doi:10.1007/978-3-540-78967-3_28. URL http://dx.doi.org/10.1007/978-3-540-78967-3_28.

[16] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE transactions on Information Theory*, 22(6):644–654, 1976.

[17] Edward Farhi, David Gosset, Avinatan Hassidim, Andrew Lutomirski, and Peter Shor. Quantum money from knots. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pages 276–289, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1115-1. doi:10.1145/2090236.2090260. URL http://doi.acm.org/10.1145/2090236.2090260.

[18] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 40–49, Oct 2013. doi:10.1109/FOCS.2013.13.

[19] Sanjam Garg, Craig Gentry, Amit Sahai, and Brent Waters. Witness encryption and its applications. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 467–476, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2029-0. doi:10.1145/2488608.2488667. URL http://doi.acm.org/10.1145/2488608.2488667.

[20] Sanjam Garg, Craig Gentry, Shai Halevi, and Daniel Wichs. On the implausibility of differing-inputs obfuscation and extractable witness encryption with auxiliary input. In JuanA. Garay and Rosario Gennaro, editors, *Advances in Cryptology CRYPTO 2014*, volume 8616 of *Lecture Notes in Computer Science*, pages 518–535. Springer Berlin Heidelberg, 2014. ISBN 978-3-662-44370-5. doi:10.1007/978-3-662-44371-2_29. URL http://dx.doi.org/10.1007/978-3-662-44371-2_29.

[21] Craig Gentry. *A Fully Homomorphic Encryption Scheme*. PhD thesis, Stanford, CA, USA, 2009. AAI3382729.

[22] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, 1986. ISSN 0004-5411. doi:http://doi.acm.org/10.1145/6490.6503.

[23] S. Goldwasser and Y.T. Kalai. On the impossibility of obfuscation with auxiliary input. In *Foundations of Computer Science, 2005. FOCS 2005. 46th Annual IEEE Symposium on*, pages 553–562, Oct 2005. doi:10.1109/SFCS.2005.60.

[24] Shafi Goldwasser and GuyN. Rothblum. On best-possible obfuscation. In SalilP. Vadhan, editor, *Theory of Cryptography*, volume 4392 of *Lecture Notes in Computer Science*, pages 194–213. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-70935-0. doi:10.1007/978-3-540-70936-7_11. URL http://dx.doi.org/10.1007/978-3-540-70936-7_11.

[25] David Gosset and Daniel Nagaj. Quantum 3-sat is qma1-complete. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 756–765. IEEE Computer Society, 2013. ISBN 978-0-7695-5135-7. doi:10.1109/FOCS.2013.86. URL http://dx.doi.org/10.1109/FOCS.2013.86.

[26] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28:1364–1396, March 1999. ISSN 0097-5397. doi:http://dx.doi.org/10.1137/S0097539793244708. URL http://dx.doi.org/10.1137/S0097539793244708.

[27] Susan Hohenberger, Amit Sahai, and Brent Waters. Replacing a random oracle: Full domain hash from indistinguishability obfuscation. In PhongQ. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 201–220. Springer Berlin Heidelberg, 2014. ISBN 978-3-642-55219-9. doi:10.1007/978-3-642-55220-5_12. URL http://dx.doi.org/10.1007/978-3-642-55220-5_12.

[28] Dominik Janzing, Pawel Wocjan, and Thomas Beth. Non-identity check is qma-complete. In *International Journal of Quantum Information*, 2005.

[29] Ralph C. Merkle. Secure communications over insecure channels. *Commun. ACM*, 21(4):294–299, April 1978. ISSN 0001-0782. doi:10.1145/359460.359473. URL http://doi.acm.org/10.1145/359460.359473.

[30] Michele Mosca and Douglas Stebila. Quantum coins. *Error-Correcting Codes, Finite Geometries and Cryptography*, 523:35–47, 2010.

[31] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: Deniable encryption, and more. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 475–484, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2710-7. doi:10.1145/2591796.2591825. URL http://doi.acm.org/10.1145/2591796.2591825.

[32] John Watrous. Limits on the power of quantum statistical zero-knowledge. In *43rd Symposium on Foundations of Computer Science (FOCS 2002), 16-19 November 2002, Vancouver, BC, Canada, Proceedings*, page 459. IEEE Computer Society, 2002. ISBN 0-7695-1822-2. doi:10.1109/SFCS.2002.1181970. URL http://dx.doi.org/10.1109/SFCS.2002.1181970.

[33] John Watrous. Zero-knowledge Against Quantum Attacks. In *STOC '06*, pages 296–305. ACM, 2006. URL http://doi.acm.org/10.1145/1132516.1132560.

[34] Hoeteck Wee. On obfuscating point functions. In *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 523–532, New York, NY, USA, 2005. ACM. ISBN 1-58113-960-8. doi:10.1145/1060590.1060669. URL http://doi.acm.org/10.1145/1060590.1060669.

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

[36] Mark Zhandry. How to Construct Quantum Random Functions. In *FOCS 2012*, pages 679–687. IEEE, 2012.

The definition of computationally indistinguishable ensembles of quantum states, as set down by Watrous [33].

Definition 13. *[33] Let $S \subset \{0,1\}^*$ be an infinite set, let $m : \{0,1\}^* \to \mathbb{N}$ be a polynomially-bounded function, and let ρ_x and ξ_x be mixed states on $m(x)$ qubits for each $x \in S$. Then the ensembles $\{\rho_x : x \in S\}$ and $\{\xi_x : x \in S\}$ are **computationally indistinguishable** if, for every choice of*

1. *polynomials p and q;*

2. *polynomially-bounded function* $k : \{0,1\}^* \to \mathbb{N}$*;*

3. *a collection* $\{\rho_x : x \in S\}$*, where* σ_x *is a mixed state on* $k(x)$ *qubits;*

4. *a quantum circuit* Q *of size at most* $p(|x|)$ *and type* $(m(x)+k(x),1)$*;*

it holds that

$$|\langle 1|Q(\rho_x \otimes \sigma_x)|1\rangle - \langle 1|Q(\xi_x \otimes \sigma_x)|1\rangle| \leq \frac{1}{q(|x|)}$$

for all but finitely many $x \in S$*.*