Public-Key Quantum Money with a Classical Bank

Omri Shmueli*

Abstract

Quantum money is a main primitive in quantum cryptography, that enables a bank to distribute to parties in the network, called wallets, unclonable quantum banknotes that serve as a medium of exchange between wallets. While quantum money suggests a theoretical solution to some of the fundamental problems in currency systems, it still requires a strong model to be implemented; quantum computation and a quantum communication infrastructure. A central open question in this context is whether we can have a quantum money scheme that uses "minimal quantumness", namely, local quantum computation and only classical communication.

Public-key semi-quantum money (Radian and Sattath, AFT 2019) is a quantum money scheme where the algorithm of the bank is completely classical, and quantum banknotes are publicly verifiable on any quantum computer. In particular, such scheme relies on local quantum computation and only classical communication. The only known construction of public-key semi-quantum is based on quantum lightning (Zhandry, EUROCRYPT 2019), which is based on a computational assumption that is now known to be broken.

In this work, we construct public-key semi-quantum money, based on quantum-secure indistinguishability obfuscation and the sub-exponential hardness of the Learning With Errors problem. The technical centerpiece of our construction is a new 3-message protocol, where a classical computer can delegate to a quantum computer the generation of a quantum state that is both, unclonable and publicly verifiable.

*Tel Aviv University, omrishmueli@mail.tau.ac.il. Supported by ISF grants 18/484 and 19/2137, by Len Blavatnik and the Blavatnik Family Foundation, by the European Union Horizon 2020 Research and Innovation Program via ERC Project REACT (Grant 756482), and by the Clore Israel Foundation.

Contents

1 Introduction

Mediums of exchange have been a central part of modern society, with the most popular of them being currency systems. Currency systems, divide into two different categories: cash-based currency systems and cashless currency systems. At the center of every cash system there is a banknote - a physical object that is (1) verifiable publicly, and (2) hard to counterfeit. In contrast, cashless systems swap the use of physical banknotes with a database of assets, governed by a middleman that approves or declines every transaction request. The main advantage of cash systems is that transactions are local. Local transactions are private and do not require communication with a third party. As a result, such systems are usually highly efficient and unbounded in their ability to handle any number of transactions simultaneously. These properties of cash become even more desirable when considering decentralizing a currency system[1]. On the opposite side, there are three main disadvantages of cash systems. First, in theory, any piece of information can be copied and our ability to prevent counterfeiting is limited. Second, banknotes are physical objects and usually take large amounts of space. Third, cash exchange requires physical contact, so, long-distance transactions are not accessible.

More than five decades ago, in a 1969 paper by Wiesner [Wie83], quantum money was introduced for the first time as an alternative cash-based currency system, suggesting a theoretical solution to all three issues above. In a quantum money system, a banknote is made up of *quantum*, rather than classical, information. The reasoning underlying such design is the no-cloning theorem [WZ82], which asserts the striking guarantee that, according to the laws of quantum mechanics, some quantum states cannot be cloned by any physical procedure, be it unbounded in its resources. Specifically, if a banknote is made up from a quantum unclonable state then the banknote is unclonable and cannot be counterfeited. Also, quantum money opened the possibility of cash that takes negligible amounts of physical space and can be sent remotely over communication channels.

Due to its desirable properties and due to the fascinating technical challenges it provides, quantum money has come a long way since Wiesner's seminal work. Today, quantum money serves as a precursor in the field of quantum cryptography, acting both as a central primitive and a breeding ground for new cryptographic techniques. Naturally, Wiesner's version of quantum money lacked many of the properties that are needed to make such idea feasible. Following works on quantum money overcame significant barriers, exploring different functionalities of quantum money and, more generally, unclonable cryptography, c.f. [BBBW83, MS06, LAF+09, Aar09, MS10, AC12, FGH+12, PYJ+12, BDS16, JLS18, BDG19, Zha19, HS20, BS20, RZ20, CLLZ21, BSS21]. Notably, are the works of Aaronson and Christiano [AC12] and Zhandry [Zha19] that achieve public-key quantum money i.e. where the receiver of a quantum banknote can locally verify it, without interacting with any other party, let alone the bank. This made quantum money behave like actual cash (i.e. where banknotes are both locally sent and locally verified) for the first time. Still, even with these advancements we are a long way to go from realizable quantum money schemes.

Semi-quantum money. A central question that remains open in this framework, is whether or not the bank can be a classical algorithm in a public-key quantum money system. This question was asked in the past in different variations, in particular, Radian and Sattath [Rad19] define this notion as Public-Key Semi-Quantum Money, along with other notions of quantum money where the bank is completely classical. Compared to public-key (fully) quantum money, in a public-key semi-quantum money scheme the bank has two additional abilities.

- **Classical Certificates of Destruction (CCoD) for Banknotes:** Any quantum wallet can return

[1] A central problem currently preventing cryptocurrencies from being adopted on a world-wide scale is the long transaction times. This is a direct cause of the combination between two design needs: (1) every transaction needs to update the database of assets, and (2) the database is decentralized in cryptocurrencies, and with each and every one of its updates the whole network needs to reach a consensus on it.

to the classical bank a valid quantum banknote it is holding. Specifically, a quantum wallet can derive a classical certificate crt from its quantum banknote, that guarantees that the banknote has been destroyed and cannot pass the public quantum verification anymore. When the bank receives crt, it can then consider that banknote as returned to the balance of the wallet that sent it.

- **Classical Minting:** The bank can execute a classical minting protocol, where it lets a quantum wallet generate, by itself, exactly one copy of a quantum banknote which is publicly verifiable by all quantum wallets.

The question of a classical bank has numerous consequences (as mentioned in previous works [Gav12, Rad19]), two main examples are below.

Sending banknotes over long distances. Sending quantum banknotes over long distances using quantum channels is tricky. In fact, we do not know how to guarantee the security of banknotes against some basic attacks. To be more precise, independently of the ability of any quantum error correcting code to protect a quantum state, when a single copy of a quantum state is sent through a channel and communication is cut at the right time (from some reason, malicious or not), the state is lost. States sent from the bank can still be safe: The bank can first send the quantum banknote, wait for a classical confirmation signal from the receiver, and then sign the state using a classical signature (e.g. the state can have a classical part that the bank can sign on). In contrast, for a wallet sending a banknote, due to the communication shutdown attack described, we don't know how to guarantee that a state will arrive to its destination without assuming the physical safety of the channel.

The above means that in a solution where there is CCoD (even where the bank does not have the ability of classical minting, and needs to quantumly generate banknotes by itself), a quantum wallet can locally generate a classical certificate crt which can then be sent to the bank over a classical channel (and classical channels are not susceptible to the shutdown attack described, as information can be copied and re-sent). Consequently, when wanting to send a quantum banknote, the wallet can choose between two options. First option is direct exchange, where the banknote is passed physically to another wallet, and the other wallet can verify it locally and quantumly without needing a middleman. Second option is long-distance transaction, where the wallet generates the CCoD crt, sends it to the bank, which can then send a new quantum banknote to the receiving wallet.

Public-key quantum money on a classical communication network. If we add classical minting along to the CCoD mechanism, it follows that all communication between the wallet and bank is classical. This gives us a scheme where the only quantum communication is between wallets, can be local and does not require a quantum communication network. Apart from the fact that a classical communication infrastructure already exists for both cabled and wireless communication, classical information is more stable and classical communication is likely to be more efficient[2].

Previous work on making the bank more classical and decreasing its quantum computational work have produced exciting research in recent years [Gav12, BDS16, Zha19, Rad19, AGKZ20, VZ21, CLLZ21]. In particular, the work of Ben-David and Sattath [BDS16] combined with the work of Coladangelo, Liu, Liu and Zhandry [CLLZ21] show how to construct public-key quantum money with CCoDs, but no classical minting. On the side of classical minting, Zhandry [Zha19] introduces the idea of Quantum Lightning, which is essentially a non-interactive and reusable classical delegation of sampling states that are unclonable and publicly verifiable. In particular, Quantum Lightning gives a solution to the classical minting problem of public-key quantum money (but does not necessarily provide classical proofs of destruction of banknotes). Zhandry [Zha19] gave a construction of Quantum Lightning based on a new computational assumption. The security of Zhandry's construction was later called into question when Roberts showed that the computational assumption is broken [Rob21]. Radian

[2]The conjectured efficiency gap between classical and quantum communication is a consequence of the better algorithmic efficiency and lower rate of classical error correcting codes, compared to their quantum counterparts.

and Sattath explain in [Rad19] how Quantum Lightning with a certificate of destruction mechanism[3] is at least as strong as public-key semi-quantum money, when adding some basic cryptographic primitives like signature schemes. To date, we still have no secure constructions of Quantum Lightning under studied assumptions, and no solution to the classical minting problem of public-key quantum money. In general terms, the main question we focus on in this work is,

Can a classical computer delegate to a quantum computer the generation of a quantum state, that is both publicly verifiable and unclonable?

1.1 Results

We resolve the open question and construct a public-key semi-quantum money scheme, that is, having both CCoD mechanism and classical minting. Our first assumption is the existence of indistinguishability obfuscation (iO) for classical circuits secure against quantum polynomial-time attacks. Our second assumption is that the Learning With Errors [Reg09] problem has sub-exponential indistinguishability against quantum computers[4], that is, there exists some constant $\delta \in (0,1)$ such that for every quantum polynomial-time algorithm, Decisional LWE cannot be solved with advantage greater than $2^{-\lambda^{\delta}}$, where $\lambda \in \mathbb{N}$ is the security parameter of LWE.

Formally, we have the following main Theorem.

Theorem 1.1. *Assume that Decisional LWE has sub-exponential indistinguishability and that indistinguishability obfuscation for classical circuits exists with security against quantum polynomial time distinguishers. Then, there is a public-key semi-quantum money scheme (as in Definition 3.3).*

The remaining of the paper is as follows. In Section 2 we explain the main ideas in our construction. The Preliminaries are given in Section 3. In Section 4 we present our construction of public-key semi-quantum money and its proof of correctness, and in Section 5 we give the security proof of the scheme.

2 Technical Overview

In this section we explain the main technical ideas in our construction. The structure of the overview is as follows: in Section 2.1 we start with reviewing the known techniques for classical delegation of unclonable state generation and discuss the challenge of public verification of such states. In Section 2.2 we describe our new technique of publicly verifiable unclonable state generation, without a security proof. In Section 2.3 we prove the security of our scheme.

2.1 The Lightning Strike Paradigm and Bolt Verifiability

Our goal in this overview will be to explain how to classically delegate the sampling of a state which is both unclonable and publicly verifiable - we will gradually explain how this is done. Let us first recall the known methods for the relaxed challenge of classical delegation of unclonable state sampling *without verification*.

That is, we currently want to construct a protocol (with security parameter $\lambda \in \mathbb{N}$), between a classical sender Sen and a quantum receiver Rec, where at the end of the interaction Rec holds a quantum state $|\psi\rangle_s$ and Sen holds a classical string $s \in \{0,1\}^{\lambda}$ that uniquely identifies the state $|\psi\rangle_s$ (i.e. the state can be

[3]The certificate of destruction mechanism is for the unclonable states generated by the lightning. In [Rad19], such mechanism is called bolt-to-certificate property of the quantum lightning.

[4]Note that this assumption is weaker than assuming that Decisional LWE is hard for sub-exponential time quantum algorithms, which is considered a standard cryptographic assumption.

generated, possibly inefficiently, from the string s). Security will say that no quantum polynomial-time malicious Rec^* can interact with Sen and output two copies of $|\psi\rangle_s$, for any s. To this end of constructing the sampling protocol we will have what we'll call a sampling scheme. This sampling scheme will help us understand the basic sampling challenge without verification, the challenge of adding verification, and the previous work on the subject.

First Version of the Sampling Scheme. The first version of the scheme is (χ, G) with the following syntax:

- $\mathsf{gk} \leftarrow \chi(1^\lambda)$: A classical polynomial-time algorithm which samples a classical key gk.
- $(\beta, |\psi\rangle_\beta) \leftarrow G(\mathsf{gk})$: A quantum polynomial-time algorithm that given the classical key gk samples a pair of one classical string $\beta \in \{0,1\}^\lambda$ and one copy of a quantum state $|\psi\rangle_\beta$.

The scheme has an unclonability property, which asserts that for any quantum polynomial-time malicious receiver Rec^*, the probability to obtain a key $\mathsf{gk} \leftarrow \chi$ and successfully output two copies of $|\psi\rangle_\beta$ (for any β) is negligible.

A scheme (χ, G) having these properties easily gives rise to the relaxed sampling protocol (without verification) in the following manner:

1. Sen samples $\mathsf{gk} \leftarrow \chi$ and sends gk to Rec.
2. Rec computes $(\beta, |\psi\rangle_\beta) \leftarrow G(\mathsf{gk})$, and sends the classical part β to Sen.

One can verify that the above protocol satisfies the requirements of the relaxed sampling protocol, and that by the unclonability property of the sampling scheme, the sampling protocol is secure and the generated state cannot be cloned.

Remark 2.1 (Sampled Unclonable States as Quantum Proofs of Entropy). A unique property (previously observed by [Zha19]) of the sampling scheme is that the quantum part $|\psi\rangle_\beta$ is effectively a *quantum proof of entropy* for the string β. More elaborately, assume there was a quantum polynomial-time adversary $\mathcal{A}$ that given a sampled key gk, maliciously produces a pair $(\beta^*, |\psi\rangle_{\beta^*})$ (by a computation that might be independent from the circuit G) such that β^* is sampled with some noticeable probability $\tilde{\varepsilon}$. This means that the pair $(\beta^*, |\psi\rangle_{\beta^*})$ can be sampled *twice* with still a noticeable probability $\tilde{\varepsilon}^2$. In such case in particular it would be possible to clone $|\psi\rangle_{\beta^*}$, in contradiction to the unclonability property of (χ, G).

The above means that as long as β is obtained together with the quantum part $|\psi\rangle_\beta$, it could have been sampled only with a negligible probability, or in other words: The quantum part $|\psi\rangle_\beta$ serves as a quantum proof for the entropy in β.

With accordance to the entropy in β, we think of the purification of the computational process $(\beta, |\psi\rangle_\beta) \leftarrow G(\mathsf{gk})$ as the formation of a lightning, and on the act of measuring the left register to obtain β as a lightning strike - a natural probabilistic event with outcome (in our case, β) that is unique with overwhelming probability[5]. The collapsed quantum state $|\psi\rangle_\beta$ is called the lightning bolt, and β is called the identifier.

There are natural examples for implementations of sampling schemes (χ, G) that can generate lightning strikes. A known cryptographic example is that χ is a sampler for the public key of a collision resistant function, and G is computing the sampled collision-resistant function H in superposition. The right output register of G (usually containing $|\psi\rangle_\beta$) is the input register to H and the left output register of G (usually containing β) is the output register of H. Measuring the left register we get β which is a uniform image $y \in \{0,1\}^\lambda$ of H and the right register $|\psi\rangle_\beta$ collapses to a uniform superposition

[5]Thinking of lightning storms in nature, we generally view the probability of two lightning strikes hitting the same point as extremely small when the area of possible strikes is uniform i.e. made up of the same material and have the same distance from the formation of the storm. The interpretation of such computational process as a lightning strike was first given in [Zha19].

$|H_y\rangle := \sum_{x:H(x)=y} |x\rangle$ of the y-preimages in H. The sampled state $|H_y\rangle$ is unclonable, because if we managed to generate two copies of it we can find a collision in H with non-negligible probability: By simply measuring the two copies, with at least probability $1/2$ the measurement outcomes are different and we have a y-collision.

The Problem of Lightning Bolt Verification. A classical sender Sen that lets the quantum receiver Rec generate lightning strikes is not a problem, but enabling *efficient* and *public* verification of lightning bolts is a different game. Our goal until the end of Section 2.2 of the technical overview will be to implement a stronger sampling scheme (χ, G, C_V, V) as follows:

- $(\mathsf{gk}, \mathsf{sk}) \leftarrow \chi(1^\lambda)$: A classical polynomial-time algorithm which samples a classical key gk, together with a secret key sk.
- $(\beta, |\psi\rangle_\beta) \leftarrow G(\mathsf{gk})$: Same as before.
- $\mathsf{vk} \leftarrow C_V(\mathsf{sk}, \beta)$: A classical polynomial-time algorithm that given a secret key sk and a valid string β, generates a public verification key vk.
- $V(\mathsf{vk}, |\psi\rangle_\beta) \in \{0, 1\}$: A quantum polynomial-time algorithm that given a verification key vk and a quantum state $|\psi\rangle_\beta$ outputs a rejection/acceptance bit.

The above strengthened scheme implies our final goal of a sampling protocol for unclonable and publicly verifiable states:

1. Sen samples $(\mathsf{gk}, \mathsf{sk}) \leftarrow \chi$ and sends gk to Rec.
2. Rec computes $(\beta, |\psi\rangle_\beta) \leftarrow G(\mathsf{gk})$ and sends β to Sen.
3. Sen computes $\mathsf{vk} \leftarrow C_V(\mathsf{sk}, \beta)$, and sends vk to Rec.

Like before, this scheme has an unclonable property which says that for any quantum polynomial-time malicious receiver Rec* that interacts with Sen in the protocol, the probability to output two copies of $|\psi\rangle_\beta$ (for any β) is negligible. Using the last algorithm V, the state $|\psi\rangle_\beta$ is now verifiable for anyone holding vk. For now, we will think of vk as an ideal obfuscation of some classical circuit, that is, the sender enables quantum oracle access to some classical efficient function, which is in turn used by the verification V. We will later move to public verification in the standard model, without oracles or ideal obfuscation.

Keeping in mind the previous example of hash functions, given H and the image y, we don't know what kind of vk the sender Sen can classically send to Rec in order to allow the classical verification of the quantum state $|H_y\rangle$. Given H, y, one can check that the obtained quantum state $|\psi\rangle_\beta$ is *some* superposition of preimages of y, by computing H in superposition with input state $|\psi\rangle_\beta$ and watching the output y. The challenge is to check the quantumness of $|\psi\rangle_\beta$ i.e. whether or not it contains more than a single entry in the superposition of y-preimages. This question has proved to be non-trivial and was asked previously. In particular, Unruh shows [Unr16b, Unr16a] that under the Learning with Errors assumption there are collision resistant hash functions where a single preimage $|x'\rangle$ of y and the entire superposition $\sum_{x:H(x)=y} |x\rangle$ are indistinguishable.

Noisy Trapdoor Claw-Free Functions and Learnable Verification. One general method that we know of, where lightning bolts can be efficiently verified, is when the lightning bolt has some "global structure", for example, if it is a subspace state. Specifically, the circuit G is such that when we measure to get β, the collapsed state $|\psi\rangle_\beta$ is a state of the form $|S_\beta\rangle := \sum_{u \in S_\beta} |u\rangle$, for some subspace $S_\beta \subseteq \{0, 1\}^\lambda$. In that case, by the following known method, the bolt $|S_\beta\rangle$ can be verified with quantum oracle access to the classical membership functions for the subspace S_β and its dual $S_\beta^\perp$. Specifically, given a quantum register R we execute the following procedure to check whether it contains $|\psi\rangle_\beta := |S_\beta\rangle$:

1. Execute on register R membership check for the subspace S_β in superposition.
2. Execute Quantum Fourier Transform (QFT) on register R.
3. Execute on register R membership check for the subspace $S_\beta^\perp$ in superposition.
4. If both subspace membership checks are verified, accept the state as valid.

The above implies that if the classical sender Sen knows how to efficiently compute the classical circuits for membership checks of the two subspaces S_β, $S_\beta^\perp$ given β, verification (at least with respect to oracles) is possible - let us not forget that in order to compute S_β, $S_\beta^\perp$ we can use the secret key sk.

For sampling states with subspace structure we have one known tool in the literature - Noisy Trapdoor Claw-Free (NTCF) functions [BCM+18]. In a nutshell, NTCFs allow the sampling of a key pair $(\mathsf{gk}, \mathsf{sk}) \leftarrow \chi$ and a quantum algorithm G such that when $G(\mathsf{gk})$ is computed and the left register is measured to get β, the right register R collapses to a quantum state exponentially close in trace distance to $|x_0^\beta\rangle + |x_1^\beta\rangle$ for some pair of strings $x_0^\beta, x_1^\beta \in \{0,1\}^\lambda$. The strings x_0^β, x_1^β are called the claw of β, and the security of the NTCF asserts that for every β in the support of $G(\mathsf{gk})$, one cannot efficiently find both strings in the claw (hence, "claw-free" function) with a non-negligible probability. Due to the claw-freeness of G, lightning bolts based on NTCFs are unclonable: If we had two copies of the bolt $|x_0^\beta\rangle + |x_1^\beta\rangle$ then with probability $1/2$ we have a claw (x_0^β, x_1^β) of some β by measuring both copies.

Regarding the verifiability of claw states: First, the set of two strings x_0^β, x_1^β can be thought of as the coset $S^\beta := \{0, x_0^\beta + x_1^\beta\} + x_0^\beta$ (i.e. the 1-dimensional subspace $\{0, x_0^\beta + x_1^\beta\}$ with constant shift x_0^β). So, the NTCF bolt is verifiable given quantum oracle access to the classical membership functions to S_β and its dual (the QFT-based verification algorithm above can be slightly modified to handle cosets rather than only subspaces, this is still within the range of known techniques). Second, the structure of NTCFs guarantees that given the secret key sk, we can efficiently compute from any valid β the claw (x_0^β, x_1^β). Since the coset S_β and its dual are efficiently computable from the claw (x_0^β, x_1^β), the sender can enable verification of claw states using sk.

To partially summarize, claw states are unclonable when no oracle is present, and when the membership oracles for the cosets are accessible, they are verifiable. These observations on NTCFs are not new. In particular, as part of their work, Radian and Sattath [Rad19] construct *private-key* semi-quantum money (i.e. where the bank is completely classical, but verification of banknotes can only be done with the assistance of the bank) based on NTCFs.

There is a catch to the above NTCF-based bolts. While *separately*, bolts from NTCFs are (1) unclonable and (2) verifiable given an oracle, these two properties cannot co-exist. Formally, claw states are in fact clonable whenever the membership oracles are accessible. In a nutshell, this follows because oracle access to the coset $S_\beta^\perp$ is *learnable*. Here is how: Given one copy of a claw state $|x_0\rangle + |x_1\rangle$, by measuring the bolt we get x_b for some bit $b \in \{0,1\}$. We then can execute $H^{\otimes\lambda}$ on x_b to get $\sum_{d\in\{0,1\}^\lambda}(-1)^{\langle x_b,d\rangle}|d\rangle$, insert that superposition into the membership check $S_\beta^\perp$, measure the result and let the state collapse with accordance to the measurement outcome. S_β is 1-dimensional and thus $S_\beta^\perp$ has $\lambda - 1$ dimensions and covers half of all $\{0,1\}^\lambda$. With probability $1/2$, the state collapses to the superposition $\sum_{d\in S_\beta^\perp}(-1)^{\langle x_b,d\rangle}|d\rangle$. Since $d \in S_\beta^\perp$ we have $\langle d, x_{\neg b}\rangle = \langle d, x_b\rangle$ and it can be verified by the reader that this state is $H^{\otimes\lambda}\cdot(|x_0\rangle + |x_1\rangle)$. By executing Hadamard again and measuring we have $x_{\neg b}$ with probability $1/2$ and thus a claw. To conclude, we do not know of a secure way to make lightning bolts based on NTCFs publicly verifiable.

2.2 Our Technique - An Alternative Lightning Bolt

We re-examine the lightning strike paradigm. Our aim is to give a different suggestion for a quantum circuit G (and more broadly, a full sampling scheme (χ, G, C_V, V)) that generates lightning strikes. Crucially, we ask that unlike the case of NTCFs, our bolts should be publicly verifiable.

We start with an observation on hybrid Quantum Fully Homomorphic Encryption (QFHE) schemes [BJ15, DSS16, Mah20, Bra18] which are a template for constructing QFHE. In a hybrid QFHE scheme, any ciphertext of any λ-qubit state $|\psi\rangle$ consists of a quantum part, which is a quantum one-time pad (QOTP) encryption $|\psi\rangle^{(x,z)} := \left(\otimes_{i\in[\lambda]} X^{x_i}\right)\cdot\left(\otimes_{i\in[\lambda]} Z^{z_i}\right)\cdot|\psi\rangle$ of $|\psi\rangle$ using classical keys $x, z \in \{0,1\}^\lambda$, and a classical part which is classical Fully Homomorphic Encryption (FHE) encryptions $\mathsf{ct}_{x,z}$ of the pad itself. The process of homomorphic evaluation of a quantum circuit Q involves changing the pad from the initial x, z to some other x', z':

$$\left(\mathsf{ct}_{x',z'}, Q\left(|\psi\rangle\right)^{(x',z')}\right) \leftarrow \mathsf{QHE.Eval}\left((\mathsf{ct}_{x,z}, |\psi\rangle^{(x,z)}), Q\right) \quad .$$

Our starting observation is that in all known hybrid QFHE schemes, in the quantum homomorphic evaluation process, the pad transformation $(x, z) \to (x', z')$ is a randomized function, at least when the evaluation is executed honestly.

The above is clearly not a proof that the pad *has* to be randomized, and it is also provably not always true - it depends on the evaluated quantum circuit Q. For example, for any Q a Clifford circuit it is a known fact that it can be computed homomorphically on any hybrid-encrypted quantum state, where the pad transformation $(x, z) \to (x', z')$ is deterministic [BJ15]. Keeping this transformation deterministic is however not known to be possible when we deal with general quantum circuits, in particular, when we need to homomorphically evaluate Toffoli gates. We'll next see that it is not known for a reason, because it is impossible.

Hybrid Quantum Homomorphic Evaluation is a Lightning Strike. We want to show that the process of quantum homomorphic evaluation itself is a lightning strike. Formally, we claim there exists a quantum circuit Q^* such that when given along with a QFHE encryption $(\mathsf{ct}_{x,z}, |s\rangle^{(x,z)})$ of a random string $s \in \{0,1\}^\lambda$, the following process generates a lightning strike:

$$\left(\underbrace{\mathsf{ct}_{x',z'}}_{\text{Identifier}}, \underbrace{Q^*\left(|s\rangle\right)^{(x',z')}}_{\text{Bolt}}\right) \leftarrow \underbrace{\mathsf{QHE.Eval}\left(\underbrace{(\mathsf{ct}_{x,z}, |s\rangle^{(x,z)})}_{\mathsf{gk}}, Q^*\right)}_{\text{Bolt generator } G} \quad ,$$

which means that it is computationally impossible to generate twice the quantum part $Q^*\left(|s\rangle\right)^{(x',z')}$ of the ciphertext, that corresponds to the same classical part $\mathsf{ct}_{x',z'}$.

The quantum circuit Q^* we suggest is this: Given an input string $s \in \{0,1\}^\lambda$ and zero-initialized ancilla $|0^{(1+\lambda)}\rangle$, generate $|+\rangle = |0\rangle + |1\rangle$ by executing Hadamard gate on the first qubit ancilla register. Then execute λ parallel Toffoli gates where for gate $i \in [\lambda]$, the two controls are the first ancilla qubit and the i-th bit of $|s\rangle$, and the target qubit is the $(1+i)$-th qubit of the ancilla (which we know is $|0\rangle$ before the Toffoli gate). The reader can verify that the obtained state is $|s\rangle \otimes \left(|0, 0^\lambda\rangle + |1, s\rangle\right)$. Q^* traces out $|s\rangle$ and outputs $|0, 0^\lambda\rangle + |1, s\rangle$.

Finally, assume toward contradiction that some adversary $\mathcal{A}$ gets QFHE encryption $(\mathsf{ct}_{x,z}, |s\rangle^{(x,z)})$ for a random s and outputs twice the quantum part of the encryption along with the classical FHE encryption of the pad of the evaluated ciphertext, that is,

$$\begin{aligned}
&Q^*\left(|s\rangle\right)^{(x',z')} \otimes Q^*\left(|s\rangle\right)^{(x',z')} \otimes \mathsf{ct}_{x',z'} \\
&= Z^{\otimes z'}\cdot\left(|x' + 0^{1+\lambda}\rangle + |x' + (1,s)\rangle\right) \otimes Z^{\otimes z'}\cdot\left(|x' + 0^{1+\lambda}\rangle + |x' + (1,s)\rangle\right) \otimes \mathsf{ct}_{x',z'} \quad .
\end{aligned}$$

We can toss $\mathsf{ct}_{x',z'}$ and measure the first bolt to get the classical measurement $x' + (b, b \cdot s)$ for some $b \in \{0,1\}$. The point is that regardless of the value of b, when we add the classical string $x' + (b, b \cdot s)$ to the bolt it cancels the shift x' but does not disturb the rest of the state, because it is still in uniform superposition. This means that the measured $x' + (b, b \cdot s)$ acts as a decryption key:

$$\begin{aligned}
&C_{+x'+(b,b\cdot s)}\left(Z^{\otimes z'}\cdot\left(|x'+0^{1+\lambda}\rangle+|x'+(1,s)\rangle\right)\right)\\
&=Z^{\otimes z'}\cdot\left(|x'+0^{1+\lambda}+x'+(b,b\cdot s)\rangle+|x'+(1,s)+x'+(b,b\cdot s)\rangle\right)\\
&=Z^{\otimes z'}\cdot\left(|0^{1+\lambda}\rangle+|1,s\rangle\right)\quad.
\end{aligned}$$

When the remaining, post-processed bolt is measured we get the secret string s with probability $1/2$ and violate the security of the QFHE.

Unlearnable Verification through High-Dimensional Subspaces. Additionally to the no-cloning guarantee, the lightning bolt generated can be seen as a uniform superposition over the 1-dimensional coset $S + x'$ (S is $\{0^\lambda, s\}$) with phase $(-1)^{\langle u,z'\rangle}$ for all u in the superposition. By having quantum oracle access to $S + x'$ and $S^\perp + z'$ it follows that similarly to how we verified NTCF-based bolts, such states are verifiable by the QFT-based verification algorithm. Such oracle access can be computed efficiently by the sender Sen, as it can get $\mathsf{ct}_{x',z'}$ from Q which made the homomorphic evaluation, and enable access to membership checks for $S + x'$ and $S^\perp + z'$. Unfortunately, also similarly to the case for NTCFs, when quantum oracle access to $S + x'$ and $S^\perp + z'$ is enabled, the lightning bolts become clonable by the same attack (it is a nice exercise to execute the very similar attack and see how to clone a QFHE-based bolt).

So, we know how to create lightning bolts from QFHE, but not how to publicly verify them. Rethinking our attacks on the public verification of both the NTCF and QFHE bolts, it can be seen that at the core of the attacks is the fact that the dimension of S was small, which made the dimension of $S^\perp$ almost full, which in turn made oracle access to $S^\perp$ learnable. Indeed, we have reason to believe that increasing the dimension of the primal subspace S to a degree, can aid in hiding the description of S and $S^\perp$. In their seminal work, Aaronson and Christiano [AC12] show that for a random $\frac{\lambda}{2}$-dimensional subspace S, getting quantum oracle access to the membership functions for S and $S^\perp$ still hide S, even given the quantum state $|S\rangle := \sum_{u\in S}|u\rangle$.

Putting the Pieces Together. The key advantage of our QFHE technique over NTCF bolts is the ability to generate bolts where the underlying subspace S can have a large dimension, as we next see. Recall the circuit Q^* we homomorphically evaluated earlier in order to create a bolt. Under the encryption, given input $s \in \{0,1\}^\lambda$ what the circuit Q^* really does is generating a subspace state $|S\rangle = \sum_{u\in S}|u\rangle$ for $S = \{0^\lambda, s\}$[6]. We then showed that if the specific quantum circuit homomorphically evaluated is Q^*, than the state is unclonable. This proof only uses the fact that Q^* generates subspace states, it is not sensitive to the dimension of the subspace, as long as it isn't too large.

More precisely, we can take Q^* the homomorphically evaluated circuit to be a generating circuit for a subspace state $|S\rangle$, for a subspace S with a larger dimension $\frac{\lambda}{2}$. Instead of encrypting a random $s \in \{0,1\}^\lambda$, the QFHE contains a generating matrix $\mathbf{M}_S \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$ for a random $\frac{\lambda}{2}$-dimensional subspace. It can be verified by the reader that if an adversary $\mathcal{A}$ generates twice the quantum part of the ciphertext, which is twice $Q^*(|\mathbf{M}_S\rangle)^{(x',z')} = \sum_{u\in S}(-1)^{\langle z',u\rangle}|x'+u\rangle$ then by the same trick we used to prove the unclonability of the previous QFHE bolts (i.e. measuring one of the copies and using the measurement result as a decryption key for the x'-part in the other copy of the bolt), we generate $\sum_{u\in S}(-1)^{\langle z',u\rangle}|u\rangle$. This follows because given a uniform superposition over any subspace, adding to the state any element in the subspace, the quantum state stays the same. By measuring $\sum_{u\in S}(-1)^{\langle z',u\rangle}|u\rangle$ we get a uniform sample in S. Now, because S is a random subspace of dimension $\frac{\lambda}{2}$ it takes a tiny

[6]The circuit Q^* we described earlier generated $|0,0^\lambda\rangle + |1,s\rangle$ for the simplicity of the first example, but having s it could have just output $|0^\lambda\rangle + |s\rangle$ which is indeed the superposition over $S = \{0^\lambda, s\}$.

fraction of the entire space of possible strings $\{0,1\}^\lambda$. Consequently, by the hiding of the QFHE, getting a sample from S should be hard.
To summarize what we saw until now, the sender Sen samples a random $\frac{\lambda}{2}$-dimensional subspace $S \subseteq \{0,1\}^\lambda$ and sends the QFHE encryption $(\mathsf{ct}_{x,z}, |\mathbf{M}_S\rangle^{(x,z)})$. The quantum receiver computes,

$$\left(\underbrace{\mathsf{ct}_{x',z'}}_{\text{Identifier}}, \underbrace{Q^* \left(|\mathbf{M}_S\rangle\right)^{(x',z')}}_{\text{Bolt}}\right) \leftarrow \underbrace{\mathsf{QHE.Eval}\left(\underbrace{(\mathsf{ct}_{x,z}, |\mathbf{M}_S\rangle^{(x,z)})}_{\mathsf{gk}}, Q^*\right)}_{\text{Bolt generator } G},$$

to generate a lightning bolt, and sends $\mathsf{ct}_{x',z'}$ to Sen. By decrypting $(x',z') = \mathsf{QHE.Dec}(\mathsf{sk}, \mathsf{ct}_{x',z'})$, Sen knows $S + x'$ and $S^\perp + z'$ and can enable quantum oracle access to them (by sending ideal obfuscations to their membership circuits).

2.3 Security in the Standard Model

It remains to explain two things: one is how Sen enables public verification of the bolt in the standard model (what vk can it send without ideal obfuscation), and second, how given this public verification in the standard model the state is still unclonable[7].

Subspace Hiding Obfuscation. In the last version of the protocol, after Rec generates the bolt and identifier $\left(\mathsf{ct}_{(x',z')}, C(|\mathbf{M}_S\rangle)^{(x',z')}\right)$ it sends the identifier $\mathsf{ct}_{(x',z')}$ to the sender in the second message of the sampling protocol. The sender can then decrypt to get the pad $(x',z') = \mathsf{QHE.Dec}(\mathsf{sk}, \mathsf{ct}_{(x',z')})$ and then have the descriptions of $S + x'$ and $S^\perp + z'$. In order for the sender to enable verification in the standard model we would like to use a known obfuscation technique, instrumental in public verification of quantum money states: the subspace hiding property [Zha19] of indistinguishability obfuscators (iO).

Subspace hiding (formally stated in Theorem 6.3 in [Zha19]) says that if injective one-way functions exist and we use iO to obfuscate a classical membership check for some coset $S + x$, if the dimension of S is bounded by $(1-\epsilon)\cdot\lambda$, where λ is the full dimension and $\epsilon \in (0,1)$ is some constant, then the obfuscation of $S + x$ is indistinguishable from an obfuscation of $T + x$ for T some random $(1-\epsilon')\cdot\lambda$-dimensional superspace of S with $\epsilon' < \epsilon$ a constant. Informally this means that when the dimension of the subspace is sufficiently small we can hide it with iO. To hide both a subspace and its dual, taking the dimension of S to be $\lambda/2$ seems ideal. It is natural to try let the sender send obfuscations of the coset membership circuits $\mathsf{O}_{S+x'} \leftarrow \mathsf{iO}(C_{S+x'})$, $\mathsf{O}_{S^\perp+z'} \leftarrow \mathsf{iO}(C_{S^\perp+z'})$ as vk, the means for public quantum verification. We examine this possibility next.

Is Plain Subspace Hiding Sufficient for Bolt Public Verification? Recent works [Zha19, CLLZ21] have shown that subspace hiding is indeed sufficient in order to publicly and securely verify unclonable states, under the following conditions:

1. The state is of the form $|S\rangle^{(x,z)} := \sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u\rangle$ for $\lambda/2$-dimensional S and any $x,z \in \{0,1\}^\lambda$. This seems to be our case as well.

2. There is no generation protocol i.e. there is a quantum bank and it generates the quantum state by itself, and sends it ready to the receiver. This isn't our case.

Indeed the fact that the bank is the author of the banknotes comes up in the security argument of such schemes. We very roughly explain how: In the security argument of previous works, cloning the state $|S\rangle^{(x,z)}$ for an unknown S, x, z, which is an information-theoretically impossible task, is reduced to

[7] In the body of this work we prove a stronger property than only no-cloning of the bolt, that it has a CCoD mechanism. For the simplicity and because the arguments are identical, we focus only on no-cloning during the technical overview.

cloning this state when also given the obfuscations O_{S+x}, $\mathsf{O}_{S^\perp+}$. In the reduction we first move from sending O_{S+x}, $\mathsf{O}_{S^\perp+z}$ to O_{T_0+x}, O_{T_1+z} using subspace hiding, and then fix T_0 a superspace of S, T_1 a superspace of $S^\perp$, $t_x := x + s$ and $t_z := z + s^\perp$ for any $x, z \in \{0,1\}^\lambda$, $s \in S$, $s^\perp \in S^\perp$. The free variables at this point are S, which is subject to $T_1^\perp \subseteq S \subseteq T_0$, and $x, s, z, s^\perp$, which are subject to $t_x = x + s$, $t_z = z + s^\perp$. Even given the fixing of T_0, T_1, t_x, t_z, cloning is still hard for the adversary as the free variables still have sufficient entropy, linear in the security parameter.

The point is that in order for the reduction to move to this setting where T_0, T_1, t_x, t_z are fixed (and in particular t_x, t_z are fixed) we exactly use the fact that *the bank can sample x, z by itself* in the original construction. In our setting, the bank only samples the starting pad x, z but does not have control over the final padding x', z' of the generated bolt - as we already saw, the creation process of x', z' is both randomized and happens on the computer of the receiver.

At the end of section 2.2 it was shown that if the adversary $\mathcal{A}$ clones the QFHE bolt then it breaks the security of the QFHE by getting a uniform sample from S. This stays the main direction of our reduction. To prove security under public verification we need to carry the reduction again, but in a setting where we send $\mathcal{A}$ the verification circuits $\mathsf{O}_{S+x'}$, $\mathsf{O}_{S^\perp+z'}$ (or, something indistinguishable from them) without knowing the QFHE secret key that is used in the original protocol to decrypt $(x', z') = \mathsf{QHE.Dec}(\mathsf{sk}, \mathsf{ct}_{(x',z')})$. At this point we get stuck: The subspace hiding guarantee lets us swap $\mathsf{O}_{S+x'}$, $\mathsf{O}_{S^\perp+z'}$ with $\mathsf{O}_{T_0+x'}$ and $\mathsf{O}_{T_1+z'}$ for random superspaces $S \subseteq T_0, S^\perp \subseteq T_1$. However, it is still unclear how to send something indistinguishable from any of these obfuscations without knowing the actual pads x', z'.

Knowing the Pads Versus Containing the Pads. We suggest a different security argument, combined with a stronger subspace hiding guarantee. We start with explaining the security argument, and we will later explain how the stronger subspace hiding follows from the same assumptions as in [Zha19].

Our first observation is that there is a redundancy of information when computing the obfuscations $\mathsf{O}_{T_0+x'}$ and $\mathsf{O}_{T_1+z'}$: For example, the straightforward way to compute the membership check for $T_0 + x'$ is to subtract x' from the input string $u \in \{0,1\}^\lambda$ and then check whether the result is in T_0. A different, yet functionally equivalent procedure C_0, will initially compute B_0 a basis for the dual subspace $T_0^\perp$ of T_0, and compute $y_{x'} := B_0 \cdot x'$. Given input $u \in \{0,1\}^\lambda$, the circuit C_0 only checks if $B_0 \cdot u = y_{x'}$. The circuit C_1 is defined analogously as a function of T_1, z'. Because these circuits are functionally equivalent, we can move from $\mathsf{O}_{T_0+x'}$ and $\mathsf{O}_{T_1+z'}$ to O_{C_0} and O_{C_1} by the standard security of indistinguishability obfuscation.

How does this help exactly? Assume we fix T_0, T_1 and let S be a random $\frac{\lambda}{2}$-dimensional subspace subjected to $T_1^\perp \subseteq S \subseteq T_0$. Now, while to simulate $\mathsf{O}_{T_0+x'}$ and $\mathsf{O}_{T_1+z'}$ we need to know x' and z', two strings of length λ, to simulate O_{C_0} and O_{C_1} we need to know $y_{x'}$, $y_{z'}$, two strings of reduced length $\lambda - \dim(T_0)$ (recall T_0 and T_1 are of the same dimension). By simply trying to guess the strings $y_{x'}$, $y_{z'}$ we can succeed in simulating O_{C_0}, O_{C_1} with probability $\left(2^{\dim(T_0)-\lambda}\right)^2$.

Unfortunately, according to the earlier subspace hiding guarantee this isn't useful as it is. The dimension of T_0 is bounded by $(1-\epsilon') \cdot \lambda$ for a constant $\epsilon' \in (0,1)$, which means that $2^{\dim(T_0)-\lambda}$ is exponentially small. The same is true for T_1. However, by looking at the actual proof of the subspace hiding property of indistinguishability obfuscators in [Zha19], one can observe that the exact same proof actually proves a stronger statement than what was claimed - the dimension of the random superspace T of S can be even larger $\lambda - \lambda^\delta$ for any constant $\delta \in (0,1)$, under the exact same computational assumptions. We explain how this is true in Section 3.1, in the proof of Lemma 3.1.

Finally, assuming the stronger subspace hiding property we complete our security reduction. If for every constant $\delta \in (0,1)$ we could sample random $(\lambda - \lambda^\delta)$-dimensional subspaces T_0, T_1 subject to $S \subseteq T_0$, $S^\perp \subseteq T_1$ and obfuscations of them are indistinguishable from S and $S^\perp$, we can amp up the security of the QFHE and get our reduction to work: Assume that the QFHE has sub-exponential

advantage security, that is, there is some constant $\delta' \in (0,1)$ such that any quantum polynomial-time algorithm cannot distinguish encryptions of different messages m, m' with advantage better than $2^{-\lambda^{\delta'}}$. By taking the subspace dimension parameter to be $\delta = \delta'/2$ we create a gap between the probability that the random guess $y'_{x'}$, $y'_{z'}$ for $y_{x'}$, $y_{z'}$ is correct (which happens with probability $\approx 2^{-\lambda^{\delta}}$) and the probability that the adversary should find a random string in S (by the increased security of the QFHE and the fact that the dimension of S is $\frac{\lambda}{2}$, $\mathcal{A}$ should not be able to do this with probability greater than $\approx 2^{-\lambda^{\delta'}}$), and because $\delta = \delta'/2$ implies $2^{-\lambda^{\delta'}} << 2^{-\lambda^{\delta}}$, we get our contradiction. So, the third and last message of the minting protocol is indistinguishability obfuscations of the classical membership checks $C_{S+x'}$, $C_{S^{\perp}+z'}$. This finishes the main technical ideas in our construction. For the full and formal details, see Section 4 which contains the construction, and Section 5 which contains the security proof of the construction.

3 Preliminaries

We rely on standard notions of classical Turing machines and Boolean circuits:

- A PPT algorithm is a probabilistic polynomial-time Turing machine.
- For a PPT algorithm M, we denote by $M(x;r)$ the output of M on input x and random coins r. For such an algorithm and any input x, we write $m \in M(x)$ to denote the fact that m is in the support of $M(x;\cdot)$.

We follow standard notions from quantum computation.

- A QPT algorithm is a quantum polynomial-time Turing machine.
- An interactive algorithm M, in a two-party setting, has input divided into two registers and output divided into two registers. For the input, one register I_m is for an input message from the other party, and a second register I_a is an auxiliary input that acts as an inner state of the party. For the output, one register O_m is for a message to be sent to the other party, and another register O_a is again for auxiliary output that acts again as an inner state. For a quantum interactive algorithm M, both input and output registers are quantum.

The Adversarial Model. Throughout, efficient adversaries are modeled as quantum circuits with non-uniform quantum advice (i.e. quantum auxiliary input). Formally, *a polynomial-size adversary* $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$, consists of a polynomial-size non-uniform sequence of quantum circuits $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, and a sequence of polynomial-size mixed quantum states $\{\rho_\lambda\}_{\lambda\in\mathbb{N}}$.

For an interactive quantum adversary in a classical protocol, it can be assumed without loss of generality that its output message register is always measured in the computational basis at the end of computation. This assumption is indeed without the loss of generality, because whenever a quantum state is sent through a classical channel then qubits decohere and are effectively measured in the computational basis.

Indistinguishability in the Quantum Setting.

- Let $f : \mathbb{N} \to [0,1]$ be a function.
  - f is negligible if for every constant $c \in \mathbb{N}$ there exists $N \in \mathbb{N}$ such that for all $n > N$, $f(n) < n^{-c}$.
  - f is noticeable if there exists $c \in \mathbb{N}, N \in \mathbb{N}$ such that for every $n \geq N$, $f(n) \geq n^{-c}$.
  - f is overwhelming if it is of the form $1 - \mu(n)$, for a negligible function μ.

- We may consider random variables over bit strings or over quantum states. This will be clear from the context.
- For two random variables X and Y supported on quantum states, quantum distinguisher circuit D with, quantum auxiliary input ρ, and $\mu \in [0,1]$, we write $X \approx_{\mathsf{D},\rho,\mu} Y$ if

$$|\Pr[\mathsf{D}(X;\rho)=1] - \Pr[\mathsf{D}(Y;\rho)=1]| \leq \mu.$$

- Two ensembles of random variables $\mathcal{X} = \{X_i\}_{\lambda\in\mathbb{N},i\in I_\lambda}$, $\mathcal{Y} = \{Y_i\}_{\lambda\in\mathbb{N},i\in I_\lambda}$ over the same set of indices $I = \cup_{\lambda\in\mathbb{N}} I_\lambda$ are said to be *computationally indistinguishable*, denoted by $\mathcal{X} \approx_c \mathcal{Y}$, if for every polynomial-size quantum distinguisher $\mathsf{D} = \{\mathsf{D}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ there exists a negligible function $\mu(\cdot)$ such that for all $\lambda \in \mathbb{N}, i \in I_\lambda$,

$$X_i \approx_{\mathsf{D}_\lambda,\rho_\lambda,\mu(\lambda)} Y_i \ .$$

- The trace distance between two distributions X, Y supported over quantum states, denoted $\mathrm{TD}(X,Y)$, is a generalization of statistical distance to the quantum setting and represents the maximal distinguishing advantage between two distributions supported over quantum states, by unbounded quantum algorithms. We thus say that ensembles $\mathcal{X} = \{X_i\}_{\lambda\in\mathbb{N},i\in I_\lambda}, \mathcal{Y} = \{Y_i\}_{\lambda\in\mathbb{N},i\in I_\lambda}$, supported over quantum states, are statistically indistinguishable (and write $\mathcal{X} \approx_s \mathcal{Y}$), if there exists a negligible function $\mu(\cdot)$ such that for all $\lambda \in \mathbb{N}, i \in I_\lambda$,

$$\mathrm{TD}\left(X_i, Y_i\right) \leq \mu(\lambda) \ .$$

In what follows, we introduce the cryptographic tools used in this work.

3.1 Indistinguishability Obfuscation

We use indistinguishability obfuscators for classical circuits, that are secure against quantum polynomial-time adversaries.

Definition 3.1. *An indistinguishability obfuscation scheme* iO *is a PPT algorithm that gets as input a security parameter* $\lambda \in \mathbb{N}$ *and a classical circuit* C*, and outputs a classical circuit. It has the following guarantees.*

- ***Correctness:*** *For every classical circuit* C *and security parameter* $\lambda \in \mathbb{N}$*, the programs* $\mathsf{iO}(1^\lambda, C)$ *and* C *are functionally equivalent.*
- ***Indistinguishability:*** *For every polynomial* $\mathrm{poly}(\cdot)$*:*

$$\{\mathsf{iO}(1^\lambda, C_0)\}_{\lambda,C_0,C_1} \approx_c \{\mathsf{iO}(1^\lambda, C_1)\}_{\lambda,C_0,C_1} \ ,$$

where $\lambda \in \mathbb{N}$*,* C_0, C_1 *are two* $\mathrm{poly}(\lambda)$*-size classical circuits with the same functionality.*

In [Zha19], it is shown that indistinguishability obfuscation schemes have the property of *subspace hiding*. This is proven in Theorem 6.3 in [Zha19]. We observe that a stronger statement can be derived from the exact same proof of Zhandry, when one small observation is added. This stronger statement is given in Lemma 3.1 below. We write the proof for the lemma below for the sake of completeness.

Lemma 3.1. *Let* iO *an indistinguishability obfuscation scheme, and assume that injective one-way functions exist. Let* $S = \{S_\lambda\}_{\lambda\in\mathbb{N}}$ *a subspace* $S \subseteq \{0,1\}^\lambda$*. For a subspace* S'*, denote by* $C_{S'}$ *a classical circuit that checks membership in* S'*. Then, for every constant* $\delta \in (0,1]$ *we have the following indistinguishability,*

$$\{\mathsf{O}_{S_\lambda}|\mathsf{O}_{S_\lambda} \leftarrow \mathsf{iO}(C_{S_\lambda})\}_{\lambda\in\mathbb{N}} \approx_c \{\mathsf{O}_T|\mathsf{O}_T \leftarrow \mathsf{iO}(C_T), T \leftarrow \mathcal{S}_{S_\lambda}\}_{\lambda\in\mathbb{N}} \ ,$$

where $\mathcal{S}_{S_\lambda}$ *is the set of all subspaces of dimension* $\lambda - \lambda^\delta$ *that contain* S_λ*, and* T *is a uniform sample from that set.*

Proof. We prove the claim by a hybrid argument. Specifically, we will only need to prove two things:

- The claim is correct when the dimension of T the random superspace of S is $\dim(S)+1$, as long as $\dim(S)+1 \leq \lambda - \lambda^\delta$.
- The size of the output obfuscated circuit O_T is bigger then the original circuit by at most an additive polynomial size.

The reason this will be sufficient is because we can perform this argument a linear amount of times as long as the upper bound on the dimension of T holds. At the end of applying the argument we use

We next show that the claim is correct whenever $\dim(T) = \dim(S)+1$ by a hybrid argument.

- Hyb_0 : The adversary $\mathcal{A}$ gets the obfuscation $\mathsf{O}(S)$ of the original base subspace S. The circuit C_S is appropriately padded so that all the programs received by the adversary in the following hybrids have the same length.
- Hyb_1 : In this hybrid, the adversary receives an obfuscation of the following function. Let $\hat{P}$ be an obfuscation under iO of the simple program Z that always outputs 0 on inputs in $\{0,1\}^{\lambda-\dim(S)}$. Let $\mathbf{B} \in \{0,1\}^{(\lambda-\dim(S))\times\lambda}$ a matrix whose rows are a basis for $S^\perp$, the space orthogonal to S. This basis can be computed by Gaussian elimination. Then $\hat{S}$ is the obfuscation under iO of the function

$$Q(x) = \begin{cases} 1 & \text{if } \mathbf{B}\cdot x = 0^{\lambda-\dim(S)} \\ 1 & \text{if } \hat{P}(\mathbf{B}\cdot x) = 1 \\ 0 & \text{Otherwise} \end{cases}$$

  Since $\hat{P}$ always outputs 0, the program Q program still accepts if and only if the input is in S. Therefore, Hyb_0 and Hyb_1 are indistinguishable by the security of the outer iO invocation.

- Hyb_2 : This hybrid is the same as Hyb_1, except that $\hat{P}$ is the obfuscation under iO of the function which is defined for $y \in \{0,1\}^*$,

$$P_y(x) = \begin{cases} 1 & \text{if } \mathsf{OWF}(x) = y \\ 0 & \text{Otherwise} \end{cases}$$

  Here, OWF is an injective one-way function, and $y = \mathsf{OWF}(x^*)$ for a random $x^* \in \{0,1\}^{\lambda-\dim(S)}$.

  At this point in the proof we slightly deviate from the proof of Theorem 6.3 in [Zha19]. By the guarantee that $\dim(S)+1 \leq \lambda-\lambda^\delta$, it follows that the row dimension of $\mathbf{B}$, is $\lambda-\dim(S) \geq \lambda^\delta+1$. This means that the security parameter of the one-way function OWF, which is the length of the random input x^*, is exactly $\lambda^\delta+1$. Note that this is still enough to invoke the security of the one-way function.

  Notice that because OWF is injective, the only point on which Z and P_y differ is x^*, and finding x^* requires inverting OWF. Therefore, if iO was a *differing inputs obfuscator*, the obfuscations

of Z and P_y would be indistinguishable. Since Z and P_y differ in only a single input, the results of [BCP14] show that iO *is* a differing inputs obfuscator for these circuits. This implies that Hyb_1 and Hyb_2 are indistinguishable.

Notice now, that $Q(\cdot)$ decides membership in the subspace S' of vectors $u \in \{0,1\}^\lambda$ such that $\mathbf{B} \cdot u$ is in the span of x^* (which is just $\{0, x^*\}$). Except with negligible probability, $x^* \neq 0^{\lambda - \dim(S)}$, and so S' has dimension $\dim(S) + 1$ and also contains S.

- Hyb_3 : In this hybrid, a random x^* is chosen, S' is constructed as above, and then obfuscated. Since $Q(\cdot)$ decides membership in S', the programs being obfuscated in Hyb_2 and Hyb_3 are the same, so these two hybrids are indistinguishable by the security of the indistinguishability obfuscation iO.
- Hyb_4 : Here, we choose $x^* \in \{0,1\}^{\lambda - \dim(S)}$ at random, except not equal to 0. Since x^* comes from a set of size $2^{\lambda - \dim(S)} \geq 2^{\lambda^\delta + 1}$ which by assumption is at least of sub-exponential size, the two distributions are statistically close. Finally, the set S' is a random $(\dim(S) + 1)$-dimensional superspace of S, so Hyb_4 is the case that corresponds to the obfuscation of T above.

□

Instantiations. Indistinguishability Obfuscation for classical circuits that has security against quantum polynomial-time attacks follows from the recent line of works on lattice-inspired iO candidates [BDGM20a, GP21, BDGM20b, DQV+21].

3.2 Leveled Hybrid Quantum Fully Homomorphic Encryption

We rely on quantum fully homomorphic encryption of a specific structure. The formal definition follows.

Definition 3.2 (Leveled Hybrid Quantum Fully-Homomorphic Encryption)**.** *A hybrid leveled quantum fully homomorphic encryption scheme is given by six algorithms* ($\mathsf{QHE.Gen}$, $\mathsf{QHE.Enc}$, $\mathsf{QHE.OTP}$, $\mathsf{QHE.Dec}$, $\mathsf{QHE.QOTP}$, $\mathsf{QHE.Eval}$) *with the following syntax:*

- $\mathsf{fhek} \leftarrow \mathsf{QHE.Gen}(1^\lambda, 1^\ell)$: *A PPT algorithm that given a security parameter $\lambda \in \mathbb{N}$ and target circuit bound $\ell \in \mathbb{N}$, samples a classical secret key* fhek.
- $m \oplus x \leftarrow \mathsf{QHE.OTP}_x(m)$: *A deterministic algorithm that takes as input a classical pad $x \in \{0,1\}^*$ and message m such that $|m| = |x|$, and outputs $m \oplus x$.*
- $\mathsf{ct} \leftarrow \mathsf{QHE.Enc}_{\mathsf{fhek}}(x)$: *A PPT algorithm that takes as input a classical string $x \in \{0,1\}^*$ and the secret key* fhek *and outputs a classical ciphertext* ct.
- $x = \mathsf{QHE.Dec}_{\mathsf{fhek}}(\mathsf{ct})$: *A deterministic algorithm that takes as input a classical ciphertext* ct *and the secret key* fhek *and outputs a string x.*
- $|\psi\rangle^{(x,z)} = \mathsf{QHE.QOTP}_{(x,z)}(|\psi\rangle)$: *A QPT algorithm that takes as input an n-qubit quantum state $|\psi\rangle$ and classical strings as quantum OTPs $x, z \in \{0,1\}^n$ and outputs its QOTP transformation $|\psi\rangle^{(x,z)} := \left(\otimes_{i\in[n]} X^{x_i}\right) \cdot \left(\otimes_{i\in[n]} Z^{z_i}\right) \cdot |\psi\rangle$.*
- $|\phi\rangle^{(x',z')}, \mathsf{ct}_{(x',z')} \leftarrow \mathsf{QHE.Eval}\left((|\psi\rangle^{(x,z)}, \mathsf{ct}_{(x,z)}), C\right)$: *A QPT algorithm that takes as input a general quantum circuit C, a quantum one-time-pad encrypted state $|\psi\rangle^{(x,z)}$ and a classical ciphertext $\mathsf{ct}_{(x,z)}$ of the pads. The evaluation outputs a QOTP encryption of some quantum state $|\phi\rangle$ encrypted under new keys (x', z') and a classical ciphertext $\mathsf{ct}_{(x',z')}$.*

The scheme satisfies the following.

- **Encryption Security:** *For every polynomials $m(\cdot)$, $\ell(\cdot)$, and quantum polynomial-time algorithm $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\})_{\lambda\in\mathbb{N}}$ there exists a negligible function $\mathrm{negl}_{\mathcal{A}}(\cdot)$ such that*

$$\left\{(m_0 \oplus x, \mathsf{ct}_x) \;\middle|\; \begin{array}{l} x \leftarrow \{0,1\}^{m(\lambda)}, \mathsf{fhek} \leftarrow \mathsf{QHE.Gen}(1^\lambda, 1^{\ell(\lambda)}), \\ \mathsf{ct}_x \leftarrow \mathsf{QHE.Enc}_{\mathsf{fhek}}(x), \end{array}\right\}_{\lambda, m_0, m_1} \approx_{\mathcal{A}_\lambda, \rho_\lambda, \mathrm{negl}_{\mathcal{A}}(\lambda)}$$
$$\left\{(m_1 \oplus x, \mathsf{ct}_x) \;\middle|\; \begin{array}{l} x \leftarrow \{0,1\}^{m(\lambda)}, \mathsf{fhek} \leftarrow \mathsf{QHE.Gen}(1^\lambda, 1^{\ell(\lambda)}), \\ \mathsf{ct}_x \leftarrow \mathsf{QHE.Enc}_{\mathsf{fhek}}(x), \end{array}\right\}_{\lambda, m_0, m_1},$$

  where $\lambda \in \mathbb{N}$, $m_0, m_1 \in \{0,1\}^{m(\lambda)}$.

  - *If there exists a constant $\delta \in (0,1]$ such that, for every adversary $\mathcal{A}$, $\forall\lambda \in \mathbb{N}$, $\mathrm{negl}_{\mathcal{A}}(\lambda) \leq 2^{-\lambda^\delta}$, we say that the QFHE scheme has sub-exponential advantage security.*

- **Homomorphism:** *There is a negligible function $\mathrm{negl}(\cdot)$ such that the following holds. Let $\mathsf{fhek} \in \mathsf{QHE.Gen}(1^\lambda, 1^\ell)$, let $\mathsf{ct}_{(x,z)} \in \mathsf{QHE.Enc}_{\mathsf{fhek}}(x, z)$, let C a quantum circuit of size $\leq \ell$, let $|\psi\rangle$ a quantum input for C. Then, $\mathrm{TD}(D_0, D_1) \leq \mathrm{negl}(\lambda)$, where D_0, D_1 are defined as follows.*

  - *D_0 : The output state $|\psi'\rangle \leftarrow C(|\psi\rangle)$.*
  - *D_1 : The state generated by first evaluating $\left(|\phi\rangle^{(x',z')}, \mathsf{ct}_{(x',z')}\right) \leftarrow \mathsf{QHE.Eval}\left((|\psi\rangle^{(x,z)}, \mathsf{ct}_{(x,z)}), C\right)$, and then decrypting $\mathsf{QHE.QOTP}_{(\tilde{x},\tilde{z})}(|\phi\rangle^{(x',z')})$ with $(\tilde{x}, \tilde{z}) = \mathsf{QHE.Dec}_{\mathsf{fhek}}(\mathsf{ct}_{(x',z')})$.*

Instantiations. Quantum Leveled Fully-Homomorphic encryption with the hybrid structure follows from the work of Mahadev [Mah20], and can be based on the hardness of Learning with Errors. Brakerski [Bra18] shows how to increase the security of QFHE using a weaker LWE assumption. Consequently, constructing QFHE that has hybrid structure, leveled, and has sub-exponential advantage security can be based on assuming Decisional LWE for quantum computers, with sub-exponential indistinguishability.

3.3 Public-key Semi-Quantum Money

In this work we construct a public-key semi-quantum money scheme based on cryptographic assumptions. Before describing our construction in Section 4, we give a definition of public-key semi-quantum money, which was formally introduced in [Rad19]. Our version of the definition is written below.

Definition 3.3 (Public-key semi-quantum money)**.** *A public-key semi-quantum money scheme consists of algorithms* (Bank, Rec, QV, GenCert, CV) *with the following syntax.*

- $(\mathsf{pk}, |\$\rangle_{\mathsf{pk}}) \leftarrow \langle \mathsf{Bank}(1^\lambda), \mathsf{Rec}(1^\lambda)\rangle_{(\mathsf{OUT}_{\mathsf{Bank}}, \mathsf{OUT}_{\mathsf{Rec}})}$ *: a classical-communication protocol between a PPT algorithm* Bank *and a QPT algorithm* Rec*. At the end of interaction the bank outputs a classical public key* pk *and the receiver outputs a quantum state $|\$\rangle_{\mathsf{pk}}$.*

- $(b, |\$'\rangle) \leftarrow \mathsf{QV}(\mathsf{pk}, |\$\rangle)$ *: A QPT algorithm that gets as input the public key and a candidate banknote $|\$\rangle$ and outputs a banknote $|\$'\rangle$ along with a bit $b \in \{0,1\}$.*

- $\mathsf{crt} \leftarrow \mathsf{GenCert}(\mathsf{pk}, |\$\rangle)$ *: A QPT algorithm that gets as input the public key and a candidate banknote and outputs a classical string* crt*.*

- $\mathsf{CV}(\mathsf{pk}, \mathsf{crt}) \in \{0,1\}$ *: A deterministic polynomial-time algorithm that takes as input the public key* pk *and a classical string* crt*, and outputs a bit.*

The scheme satisfies the following guarantees.

- **Statistical Correctness:** *There exists a negligible function* $\mathrm{negl}(\cdot)$ *such that for every* $\lambda \in \mathbb{N}$*,*

$$\Pr\left[(1,|\$'\rangle) \leftarrow \mathsf{QV}(\mathsf{pk},|\$\rangle_{\mathsf{pk}}) \;\middle|\; (\mathsf{pk},|\$\rangle_{\mathsf{pk}}) \leftarrow \langle \mathsf{Bank}(1^\lambda), \mathsf{Rec}(1^\lambda)\rangle\right] \geq 1-\mathrm{negl}(\lambda) \ .$$

- **Security:** *Let* $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ *a quantum polynomial-time algorithm, and consider the following game:*

  - *$\mathcal{A}$ interacts with the bank* $\mathsf{Bank}(1^\lambda)$ *in the minting protocol. At the end, $\mathcal{A}$ outputs a quantum register* BN^**, and in formal notations,* $(\mathsf{pk},\mathsf{BN}^*) \leftarrow \langle \mathsf{Bank}(1^\lambda), \mathcal{A}_\lambda(\rho_\lambda)\rangle_{(\mathsf{OUT}_{\mathsf{Bank}},\mathsf{OUT}_{\mathcal{A}})}$.

  Then, there exists a negligible function $\mathrm{negl}(\cdot)$ *such that for all* $\lambda \in \mathbb{N}$*, for each of the below events, the probability for it to occur is* $\leq \mathrm{negl}(\lambda)$*:*

  - **Counterfeiting:** $\mathsf{BN}^* = (\mathsf{crt},|\$\rangle)$*, such that* $\mathsf{CV}(\mathsf{pk},\mathsf{crt}) = 1$ *and* $(1,|\$'\rangle) \leftarrow \mathsf{QV}(\mathsf{pk},|\$\rangle)$.
  - **Quantum Sabotage:** $\mathsf{BN}^* = |\$\rangle$ *such that* $(1,|\$'\rangle) \leftarrow \mathsf{QV}(\mathsf{pk},|\$\rangle)$ *on first execution of* QV*, and then* $(0,|\$''\rangle) \leftarrow \mathsf{QV}(\mathsf{pk},|\$'\rangle)$.
  - **Classical Sabotage:** $\mathsf{BN}^* = |\$\rangle$ *such that* $(1,|\$'\rangle) \leftarrow \mathsf{QV}(\mathsf{pk},|\$\rangle)$ *on first execution of* QV*, and then* $\mathsf{crt} \leftarrow \mathsf{GenCert}(\mathsf{pk},|\$'\rangle)$, $\mathsf{CV}(\mathsf{pk},\mathsf{crt}) = 0$.

The above definition is relatively succinct compared to the number of protections it guarantees. We go over these derived guarantees here.

Security against sabotage. Security against quantum and classical sabotage protects wallets in the system i.e. banknote holders. It basically says that when a wallet is given a quantum banknote and it passed the public quantum verification $\mathsf{QV}(\mathsf{pk},\cdot)$ once, it will pass all further quantum verifications with overwhelming probability, and at the end of this process we can destroy the banknote with $\mathsf{GenCert}(\mathsf{pk},\cdot)$, to successfully generate a valid classical certificate of destruction crt that will be verified by $\mathsf{CV}(\mathsf{pk},\cdot)$.

Security against counterfeiting is intended to protect the bank. The guarantee says that an adversary cannot output both a quantum banknote and a corresponding classical certificate of destruction for it. This guarantee is stronger then no cloning: This follows as if we had an extra copy of a quantum state that passes quantum verification, we keep one copy on the side and process the second like this: due to the security against classical sabotage, this state yields a valid classical certificate with overwhelming probability. In that case we have one quantum banknote on the side and now a classical certificate.

Correctness. The formal correctness guarantee says that when the protocol is executed honestly, then the generated banknote $|\$\rangle$ passes quantum verification with overwhelming probability. When combined with security against classical sabotage, this means that the banknote which passed the a quantum verification will successfully generate a classical certificate of destruction crt that passes the classical verification CV. So, when the protocols are executed honestly the banknote both passes quantum verification and classical certificate generation and verification.

Multi-session Security. The above definition considers security over a single session of the minting protocol between the bank and the adversary $\mathcal{A}$. However, under standard cryptographic assumptions, without the loss of generality the definition captures multi-session security, where arbitrarily many pairs $(\mathsf{pk}_1,|\$\rangle_{\mathsf{pk}_1})$, $(\mathsf{pk}_2,|\$\rangle_{\mathsf{pk}_2})$, $\cdots$, can be generated in many different sessions: Using quantum-secure classical digital signatures, the bank can sign on each of the classical public keys it samples. In multi-session security the adversary can perform the minting protocol with the bank arbitrarily many times to generate many different banknotes, and still can't counterfeit or sabotage any of the banknotes. Like in the definition from [Rad19], multi-session security requires the bank to be stateful and keep a database of banknotes that have been previously destroyed by the classical certificate of destruction mechanism, this in order to prevent re-satisfying the classical verification of the bank, using the same classical certificate of the same banknote.

4 Public-Key Semi-Quantum Money Construction

In this section we present our construction of a Public-key Semi-Quantum Money scheme (Definition 3.3), and proof of correctness.

Ingredients and notation:

- A quantum hybrid fully homomorphic encryption scheme (QHE.Gen, QHE.Enc, QHE.OTP, QHE.Dec, QHE.QOTP, QHE.Eval), with sub-exponential advantage security (Definition 3.2).
- An indistinguishability obfuscation scheme iO (Definition 3.1).

We describe the scheme in Figure 1, it includes the minting protocol $\langle \mathsf{Bank}, \mathsf{Rec} \rangle$, the quantum public verification QV of a banknote and the classical verification CV of a classical certificate of destruction. The certificate generation GenCert is simply a measurement of the banknote in the standard basis.

4.1 Correctness and Sabotage Resistance

We prove that our scheme is correct, and that it is secure against sabotage. On the side of correctness, we first show that if the scheme's algorithms are ran honestly, then the protocol ends successfully with probability $1 - \mathrm{negl}(\lambda)$. Also, if the protocol ends successfully, then a quantum banknote generated in the minting protocol passes quantum verification with probability $1 - \mathrm{negl}(\lambda)$, which will finish the correctness argument. Finally, on the side of sabotage protection, a quantum banknote which passed the quantum verification, passes another quantum verification with probability 1, and also generates a classical certificate of destruction that passes the classical verification with probability 1.

Claim 4.1. *At the end of a successful honest execution of the minting protocol, the state $|\psi\rangle^{(x,z)}$ has negligible trace distance to,*

$$|S\rangle^{(x,z)} := \sum_{u \in S} (-1)^{\langle z,u \rangle} |x+u\rangle \ ,$$

where x, z are the strings obtained by decryption in step 3 of the protocol.

Proof. In case the protocol ends successfully, it follows readily from the statistical correctness of the QFHE that the output state is a quantum one-time pad (QOTP) encryption of a state that has a negligible trace distance to the state $C(\mathbf{M}_S) = |S\rangle$, which is exactly $|S\rangle^{(x,z)}$. □

We also use a known fact from the literature on public-key quantum money [AC12, BDS16]: That the quantum verification procedure that we use on coset states (which is known in the literature) is projective, and a successful verification projects the state to be exactly $|S\rangle^{(x,z)}$.

Lemma 4.1 (Verification of Coset States is Projective, [AC12, BDS16])**.** *Consider the quantum verification procedure* QV *described in Figure 1, for a subspace $S \subseteq \{0,1\}^\lambda$ and strings $x, z \in \{0,1\}^\lambda$. Then, a successful verification procedure acts as a projection of the state in* BN *on the state,*

$$|S\rangle^{(x,z)} := \sum_{u \in S} (-1)^{\langle z,u \rangle} |x+u\rangle \ .$$

At this point we can state the correctness of the scheme.

Proposition 4.1. *The scheme presented in Protocol 1 has statistical correctness (Definition 3.3).*

Proof. First, we explain why when executed honestly, the minting protocol ends successfully with probability $1 - \mathrm{negl}(\lambda)$. The only scenario where there is an abort in the honest execution of the protocol is in step 3, $x \in S$. By Claim 4.1, after the successful honest execution of the protocol, the quantum part

Protocol 1

Minting Protocol: The joint input is the security parameter $\lambda \in \mathbb{N}$.

1. Bank samples a random $\frac{\lambda}{2}$-dimensional subspace $S \subseteq \{0,1\}^\lambda$, described by a matrix $\mathbf{M}_S \in \{0,1\}^{\frac{\lambda}{2} \times \lambda}$. Samples OTP key $r \leftarrow \{0,1\}^{\frac{\lambda^2}{2}}$ to encrypt $\mathbf{M}_S^{(r)} = \mathsf{QHE.OTP}_r(\mathbf{M}_S)$, and then $\mathsf{fhek} \leftarrow \mathsf{QHE.Gen}(1^\lambda)$, $\mathsf{ct}_r \leftarrow \mathsf{QHE.Enc}_{\mathsf{fhek}}(r)$. Bank sends the encryption $(\mathbf{M}_S^{(r)}, \mathsf{ct}_r)$ to Rec.

2. Let C the quantum circuit that for an input matrix $\mathbf{M} \in \{0,1\}^{\frac{\lambda}{2} \times \lambda}$, outputs a uniform superposition of its row span. Rec homomorphically evaluates C: $\left(|\psi\rangle^{(x,z)}, \mathsf{ct}_{x,z}\right) \leftarrow \mathsf{QHE.Eval}\left((\mathbf{M}_S^{(r)}, \mathsf{ct}_r), C\right)$. Rec saves the quantum part $|\psi\rangle^{(x,z)}$ and sends the classical part $\mathsf{ct}_{x,z}$ to Bank.

3. Bank decrypts $(x,z) = \mathsf{QHE.Dec}_{\mathsf{fhek}}(\mathsf{ct}_{x,z})$. If $x \in S$, the interaction is terminated. Recall $\mathbf{M}_S$ is a matrix with rows that are a basis for S, and let $\mathbf{M}_{S^\perp} \in \{0,1\}^{\frac{\lambda}{2} \times \lambda}$ a matrix with rows that are a basis for $S^\perp$. Bank computes indistinguishability obfuscations $\mathsf{O}_{S+x} \leftarrow \mathsf{iO}(\mathbf{M}_S, x)$, $\mathsf{O}_{S^\perp+z} \leftarrow \mathsf{iO}(\mathbf{M}_{S^\perp}, z)$.

   The classical output of Bank is $\mathsf{pk} := \left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}\right)$, the quantum output of Rec is $|\$\rangle_{\mathsf{pk}} := |\psi\rangle^{(x,z)}$.

Quantum Verification:

- $\mathsf{QV}\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \mathsf{BN}\right)$: The verifier executes:
  - Checks that the output qubit of the computation $\mathsf{O}_{S+x}(\mathsf{BN})$[a] is 1.
  - Executes Hadamard transform $H^{\otimes\lambda}$ on BN and then checks that the output qubit of the computation $\mathsf{O}_{S^\perp+z}(\mathsf{BN})$ is 1.

  If both checks passed, the verifier executes $H^{\otimes\lambda}$ again on BN and accepts the banknote.

Classical Certificate Generation:

- $\mathsf{GenCert}\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \mathsf{BN}\right)$: In order to generate a classical certificate, the note holder measures BN in the standard basis $\mathsf{crt} \leftarrow Measure(\mathsf{BN})$.

Classical Certificate Verification:

- $\mathsf{CV}\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \mathsf{crt}\right)$: The bank accepts the certificate iff $\mathsf{O}_{S+x}(\mathsf{crt}) = 1$.

[a] We are running a classical function on a quantum input, which can be interpreted as running a classical function in superposition.

Figure 1: A public-key semi-quantum money scheme.

$|\psi\rangle^{(x,z)}$ of the QFHE encryption is negligibly close to $|S\rangle^{(x,z)}$ in trace distance. This means that if $x \in S$, then the state $|S\rangle^{x,z}$ is $|S\rangle^{(0^\lambda,z)}$ and thus the state $|\psi\rangle^{(x,z)}$ has a negligible trace distance from $|S\rangle^{(0^\lambda,z)}$, and measuring $|\psi\rangle^{(x,z)}$ in the computational basis yields, with a noticeable probability, $s \in (S \setminus \{0^\lambda\})$.

Finding a non-trivial vector $s \in (S \setminus \{0^\lambda\})$, given a QFHE encryption (M_S^r, ct_r) of a basis M_S for

S a random $\frac{\lambda}{2}$-dimensional subspace is impossible to do with a noticeable probability for any quantum polynomial-time algorithm, by the security of the QFHE. It follows that with at most probability $\mathrm{negl}(\lambda)$ that $x \in S$.

Again by Claim 4.1, if the protocol ends successfully then the state $|\psi\rangle^{(x,z)}$ is negligibly close to $|S\rangle^{(x,z)}$, and thus the projection on $|S\rangle^{(x,z)}$ succeeds with overwhelming probability $1 - \mathrm{negl}(\lambda)$. Due to Lemma 4.1, the verification $\mathsf{QV}\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \cdot\right)$ succeeds with the same probability $1 - \mathrm{negl}(\lambda)$.

To conclude, when the minting protocol is executed honestly then with probability $1 - \mathrm{negl}_1(\lambda)$ the protocol ends successfully. If the protocol ends successfully, then the state $|\psi\rangle^{(x,z)}$ passes the quantum verification $\mathsf{QV}\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \cdot\right)$ with probability $1-\mathrm{negl}_2(\lambda)$. Overall, the probability that both events happen is $1 - \mathrm{negl}_3(\lambda)$. □

Security Against Quantum and Classical Sabotage. Security against quantum and classical sabotage follows readily from Lemma 4.1: After one successful quantum verification, the state is exactly $|S\rangle^{(x,z)}$. This means that another quantum verification $\mathsf{QV}\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \cdot\right)$ will succeed with probability 1, and furthermore, measuring the state will yield with probability 1 a string in $S + x$, which is verified with probability 1 by the classical verification algorithm $\mathsf{CV}\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \cdot\right)$.

5 Security Against Counterfeiting

In this section we prove that the scheme is secure against counterfeiting attacks (Definition 3.3).

Proposition 5.1 (Security against Counterfeiting)**.** *The public-key semi-quantum money scheme described in Protocol 1 has security against counterfeiting, according to Definition 3.3.*

Proof. Let $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ a quantum polynomial time adversary that succeeds in counterfeiting with some non-negligible probability $\varepsilon = \{\varepsilon_\lambda\}_{\lambda\in\mathbb{N}}$. Whenever $\mathcal{A}$ counterfeits successfully it sends a quantum banknote BN and a classical certificate crt, such that the quantum register BN passes quantum verification $\mathsf{QV}(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \cdot)$ and crt passes classical certificate verification $\mathsf{CV}(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}, \cdot)$. We next describe a sequence of hybrid experiments.

- Hyb_0 : The original experiment.

We define Hyb_0 as the original attack, where the adversary $\mathcal{A}$ interacts with the bank and successfully counterfeits by sending BN and crt. Let $|\psi\rangle_{\mathcal{A}}$ the (pure) quantum state in BN, at the end of the execution of $\mathcal{A}$, before executing the quantum verification QV, and after tracing out all other quantum registers in the system (note that $|\psi\rangle_{\mathcal{A}}$ is a random variable over quantum states). We define Hyb_0 to be successful if $|\psi\rangle_{\mathcal{A}}$ passes the quantum verification and crt passes the classical verification. Note that the quantum verification QV executes only on the register BN and thus commutes with any other quantum operation on a register entangled with BN at the point where $\mathcal{A}$ finishes executing, thus, the probability that $|\psi\rangle_{\mathcal{A}}$ passes the verification QV is identical to that of BN, without tracing out other registers in the system. The experiment Hyb_0 is thus successful with probability ε.

- Hyb_1 : Using un-obfuscated circuits for verification.

This hybrid is identical to Hyb_0 with the following change: At the end of the experiment, when applying the quantum and classical verifications, instead of using the obfuscated circuits O_{S+x}, $\mathsf{O}_{S^\perp+z}$, we use the original circuits C_{S+x}, $C_{S^\perp+z}$.

From the correctness of the obfuscation schemes, using either a circuit or its obfuscated version has the exact same functionality, which implies that the success probability of the two experiments is the same, ε.

- Hyb_2 : Removing subspace information from obfuscated circuits.

Let $\delta' \in (0,1]$ the sub-exponential security level of the QFHE (that is, any quantum polynomial-time algorithm cannot break the security of the QFHE with advantage bigger than $2^{-\lambda^{\delta'}}$), and denote $\delta := \frac{\delta'}{2}$. This hybrid is identical to Hyb_1, with the only difference is that when the bank returns the obfuscations O_{S+x}, $\mathsf{O}_{S^\perp+z}$ at step 3 of the minting protocol, the obfuscations are changed: We sample two random $\lambda-\lambda^\delta$-dimensional subspaces $T_0, T_1 \subseteq \{0,1\}^\lambda$ subjected to $T_1^\perp \subseteq S \subseteq T_0$ (or in other words, $S^\perp \subseteq T_1$, $S \subseteq T_0$). We send $\mathsf{O}_{T_0+x} \leftarrow \mathsf{iO}(\mathbf{M}_{T_0}, x)$ instead of $\mathsf{O}_{S+x} \leftarrow \mathsf{iO}(\mathbf{M}_S, x)$, and $\mathsf{O}_{T_1+z} \leftarrow \mathsf{iO}(\mathbf{M}_{T_1}, z)$ instead of $\mathsf{O}_{S^\perp+z} \leftarrow \mathsf{iO}(\mathbf{M}_{S^\perp}, z)$.

Note that because in both cases, i.e. in Hyb_1 and in Hyb_2, we know what is S, $S^\perp$, x and z before we check if the experiment was successful (which means trying to verify $|\psi\rangle_{\mathcal{A}}$ and crt, quantumly and classically respectively, using the verification circuits $S+x$, $S^\perp+z$), we can obtain the obfuscated circuits (which we send to the adversary in step 3) from an outside source. This means that we can use a distinguisher between Hyb_1 and Hyb_2 as a distinguisher between obfuscations of the original functions $S+x$, $S^\perp+z$ and obfuscations of T_0+x, T_1+z. The success probabilities of Hyb_1 and Hyb_2 are thus indistinguishable by the subspace hiding property of indistinguishability obfuscators 3.1. This means in particular that the success probability of Hyb_2 is negligibly close to the of Hyb_1 and is thus $\geq \varepsilon - \mathrm{negl}(\lambda)$.

- Hyb_3 : Lowering the need to fully know x, z in order to compute the obfuscations.

The difference between this process and the previous is that when we send the obfuscations O_{T_0+x}, O_{T_1+z} at step 3 of the generation protocol, the way in which we check membership in each of the cosets is this: Let B_0 a basis for $T_0^\perp$, let B_1 a basis for $T_1^\perp$ and let $y_x, y_z \in \{0,1\}^{\lambda^\delta}$ defined as $y_x := B_0 \cdot x$, $y_z := B_1 \cdot z$. O_{T_0+x} is changed to be an obfuscation of a circuit that for input $u \in \{0,1\}^\lambda$ checks whether $B_0 \cdot u = y_x$. O_{T_1+z} is changed to be an obfuscation of a circuit that for input $u \in \{0,1\}^\lambda$ checks whether $B_1 \cdot u = y_z$.

One can verify that the functionality of the obfuscated circuits O_{T_0+x}, O_{T_1+z} did not change, and thus by the security of the indistinguishability obfuscation schemes, the distributions are indistinguishable and the success probability of Hyb_3 is $\varepsilon - \mathrm{negl}(\lambda)$.

- Hyb_4 : Changing the order of sampling the subspaces S, T_0, T_1.

This process is the same as the previous hybrid, only that we change the order of sampling the subspaces: In the previous hybrid, we sample S first as a random $\frac{\lambda}{2}$-dimensional subspace of $\{0,1\}^\lambda$. Then we sample the subspaces T_0, T_1 as random $(\lambda-\lambda^\delta)$-dimensional subspaces conditioned on $T_1^\perp \subseteq S \subseteq T_0$. In the current hybrid, we sample T_0, T_1 first as random $(\lambda-\lambda^\delta)$-dimensional subspaces of $\{0,1\}^\lambda$, conditioned on $T_1^\perp \subseteq T_0$. Then we sample S as a random $\frac{\lambda}{2}$-dimensional subspace conditioned on $T_1^\perp \subseteq S \subseteq T_0$.

Note that while the algorithm for sampling the subspaces changed, the distributions (S, T_0, T_1) that both algorithms output are identical. This in particular means that the success probability of the experiments is the same, and the success probability of the current hybrid is $\varepsilon - \mathrm{negl}(\lambda)$.

- Hyb_5 : Fixing the subspaces T_0, T_1.

We can take the sampling procedure of the subspaces described in the previous hybrid and perform an averaging argument on the sampling of T_0, T_1, to take the samples that maximize the success probability of the previous hybrid. It is straightforward to make this averaging argument at this point, because T_0, T_1 are sampled before everything else. This means that there exist *fixed* T_0, T_1 for which the experiment is successful with probability $\geq \varepsilon - \mathrm{negl}(\lambda)$. This process where these intermediate samples are fixed is defined to be Hyb_5.

- Hyb_6 : Changing the success definition of the experiment.

The change from this hybrid and previous is the success probability of the two hybrid experiments: In the previous hybrid, the success of the experiment is defined to be when the state $|\psi\rangle_{\mathcal{A}}$ passes the quantum verification and crt passes the classical verification, both using the classical membership circuits C_{S+x}, $C_{S^\perp+z}$. In the current hybrid, the experiment is defined to be successful for the following operation: Let $C_{+\mathsf{crt}}$ the unitary circuit that maps $\forall y \in \{0,1\}^\lambda : |y\rangle \to |y+\mathsf{crt}\rangle$. Instead of verifications, we execute $|\psi\rangle'_{\mathcal{A}} := C_{+\mathsf{crt}}(|\psi\rangle_{\mathcal{A}})$, then measure $s^* \leftarrow \text{Measure}(|\psi\rangle'_{\mathcal{A}})$, and the experiment is successful if $s^* \in (S \setminus T_1^\perp)$.

Writing $|\psi\rangle_{\mathcal{A}}$ in a basis with a non-negligible amplitude on $|S\rangle^{x,z}$: To see why the success probability of the current hybrid is still non-negligible, let $\mathcal{B}$ an orthonormal basis for the space of λ-qubit pure quantum states such that one of its basis vectors is $|S\rangle^{x,z}$. Now write $|\psi\rangle_{\mathcal{A}}$ as a linear combination of the vectors in $\mathcal{B}$:

$$|\psi\rangle_{\mathcal{A}} = \sum_{w\in(\mathcal{B}\setminus\{|S\rangle^{x,z}\})} \alpha_w \cdot w + \alpha_S \cdot |S\rangle^{x,z} \ .$$

The success of the previous hybrid experiment is defined to be when (1) $|\psi\rangle_{\mathcal{A}}$ is successfully projected on $|S\rangle^{x,z}$ and (2) $\mathsf{crt} \in (S+x)$, and the success probability is non-negligible $\varepsilon - \mathrm{negl}(\lambda)$. From the fact that the current and previous hybrids execute identically until the end of the execution of $\mathcal{A}$, it follows that if we denote by ε_S the probability for which $\mathsf{crt} \in (S+x)$, we obtain that $|\alpha_S|^2 \cdot \varepsilon_S$ is a non-negligible probability.

Applying $C_{+\mathsf{crt}}$ to $|S\rangle^{x,z}$ for $\mathsf{crt} \in (S+x)$: Notice the following property of the state $|S\rangle^{x,z}$: when $\mathsf{crt} \in (S+x)$ we have $\mathsf{crt} = v + x$ for some $v \in S$, and when we execute $C_{+\mathsf{crt}}$ on $|S\rangle^{x,z}$ we get:

$$C_{+\mathsf{crt}}\left(|S\rangle^{x,z}\right) = C_{+\mathsf{crt}}\left(\sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u\rangle\right) = \sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u+\mathsf{crt}\rangle$$

$$= \sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u+x+v\rangle = \sum_{u\in S}(-1)^{\langle z,u\rangle}|u+v\rangle \ .$$

By denoting $u' := u + v$ the above state is:

$$\sum_{u'\in S}(-1)^{\langle z,u'+v\rangle}|u'\rangle = (-1)^{\langle z,v\rangle}\cdot\sum_{u'\in S}(-1)^{\langle z,u'\rangle}|u'\rangle = (-1)^{\langle z,v\rangle}\cdot|S\rangle^{0^\lambda,z} \equiv |S\rangle^{0^\lambda,z} \ .$$

In other words, the state $C_{+\mathsf{crt}}(|S\rangle^{x,z})$ gives $|S\rangle^{0^\lambda,z}$, which is an x-part decryption of the quantum one-time-pad-encrypted state $|S\rangle$. Intuitively this follows because the state $|S\rangle$ is unaffected by shifts v that are in the subspace S.

Measuring $s^* \leftarrow C_{+\mathsf{crt}}(|\psi\rangle_{\mathcal{A}})$: $C_{+\mathsf{crt}}(\cdot)$ is a unitary circuit and thus applying it to $|\psi\rangle_{\mathcal{A}}$ gives us,

$$C_{+\mathsf{crt}}(|\psi\rangle_{\mathcal{A}}) = C_{+\mathsf{crt}}\left(\sum_{w\in(\mathcal{B}\setminus\{|S\rangle^{x,z}\})}\alpha_w\cdot w\right) + \alpha_S\cdot C_{+\mathsf{crt}}\left(|S\rangle^{x,z}\right)$$

$$= \sum_{w\in(\mathcal{B}\setminus\{|S\rangle^{x,z}\})}\alpha_w\cdot C_{+\mathsf{crt}}\left(w\right) + \alpha_S\cdot|S\rangle^{0^\lambda,z} \ ,$$

where all vectors in the above sum are orthogonal to each other (they were an orthonormal set before applying C_{crt}, which is a unitary transformation, which preserves inner product). It follows that whenever $\mathsf{crt} \in (S+x)$, which happens with probability ε_S, measuring $C_{+\mathsf{crt}}(|\psi\rangle_{\mathcal{A}})$ yields a *uniformly random* sample s^* from S with probability $\geq |\alpha_S|^2$. This means that we get a uniform sample from S with

overall probability $\geq \varepsilon_S \cdot |\alpha_S|^2$, which is non-negligible as we explained. Recall that a uniform sample from S gives a vector in $(S \setminus T_1^{\perp})$ with probability $\frac{2^{\lambda^\delta}}{2^{\frac{\lambda}{2}}}$ exponentially close to 1. Overall, there is a non-negligible probability ε' such that $s^* \leftarrow C_{+\mathsf{crt}}(|\psi\rangle_{\mathcal{A}})$ satisfies $s^* \in (S \setminus T_1^{\perp})$.

- Hyb_7 : Losing the QFHE secret key.

This experiment is identical to the previous hybrid with one change: In step 3, when the bank usually decrypts the QFHE classical part to get the QOTP keys x, z, the current hybrid process does not decrypt to get x, z and instead it samples uniformly random $y'_x, y'_z \in \{0,1\}^{\lambda^\delta}$, and inserts these strings as y_x, y_z in the obfuscations O_{T_0+x}, O_{T_1+z}, respectively.

Observe that conditioned on the probabilistic event $y'_x = y_x$, $y'_z = y_z$ (for which to happen, the probability is exactly $2^{-2\cdot\lambda^\delta}$), the current and previous hybrids distribute identically. It follows that the success probability in Hyb_7 is at least $2^{-2\cdot\lambda^\delta} \cdot \varepsilon' > 2^{-3\cdot\lambda^\delta}$.

- Hyb_8 : Clearing all given knowledge on S.

This hybrid is identical to the previous, with the exception that instead of the honest bank sending the QFHE encryption $(\mathbf{M}_S^{(r)}, \mathsf{ct}_r)$ in step 1 of the minting protocol, it sends an encryption of (a matrix of) zeros $(\mathbf{M}_0^{(r)}, \mathsf{ct}_r)$, $M_0 = 0^{\frac{\lambda}{2}\times\lambda}$.

Note that in order to execute Hyb_7 there is no need to know the secret key of the QFHE scheme, so it follows that we can invoke the security of the QFHE to argue the indistinguishability of the current and previous hybrids, and in particular the indistinguishability between their success probabilities. Specifically, we use the sub-exponential-advantage security of the QFHE, so the success probability of Hyb_8 is $> 2^{-3\cdot\lambda^\delta} - 2^{-\lambda^{\delta'}} > 2^{-3\cdot\lambda^\delta - 1}$.

- Getting a contradiction by using Hyb_8.

At this point in the proof we can use the process Hyb_8 in order to perform a task that is information-theoretically impossible. Specifically, observe that Hyb_8 has the fixed subspaces T_0, T_1 and has no input (the process generates by itself the encryption of zeros $(\mathbf{M}_0^{(r)}, \mathsf{ct}_r)$), and outputs a vector $s^* \in (S \setminus T_1^{\perp})$ for a random $\frac{\lambda}{2}$-dimensional subspace S subjected to $T_1^{\perp} \subseteq S \subseteq T_0$. This is exactly in contradiction to Claim 5.1. □

We prove a supporting claim to the proof of the main Proposition 5.1.

Claim 5.1. *For any natural number λ, two subspaces $T_0, T_1 \subseteq \{0,1\}^\lambda$ and a constant $\delta \in (0,1)$ such that $T_1^{\perp} \subseteq T_0$, $\dim(T_1^{\perp}) = \lambda^\delta$, $\dim(T_0) = \lambda - \lambda^\delta$, assume we sample a random subspace S subject to $T_1^{\perp} \subseteq S \subseteq T_0$, $\dim(S) = \frac{\lambda}{2}$. Then for any (possibly unbounded) algorithm the probability to output $s \in (S \setminus T_1^{\perp})$ is bounded by $2^{-\frac{\lambda}{2}+\lambda^\delta+1}$.*

Proof. Let $\mathcal{A}$ any unbounded algorithm. As $\mathcal{A}$ got no input, we can make an averaging argument on the output s of $\mathcal{A}$ that maximizes the probability to guess s that hits the set $(S \setminus T_1^{\perp})$. So, $\mathcal{A}$ always outputs some $s^* \in \{0,1\}^\lambda$, independently of the sampled S.

If $s^* \notin (T_0 \setminus T_1^{\perp})$ then $s^* \notin (S \setminus T_1^{\perp})$ and the proof ends as the probability that $\mathcal{A}$ guesses correctly is 0. Since S is a uniformly random $\frac{\lambda}{2}$-dimensional subspace subject to $T_1^{\perp} \subseteq S \subseteq T_0$, it follows that for *any* string $s^* \in (T_0 \setminus T_1^{\perp})$, the probability that $s^* \in \left(S \setminus T_1^{\perp}\right)$ is the same, which is,

$$\frac{|S \setminus T_1^{\perp}|}{|T_0 \setminus T_1^{\perp}|} = \frac{2^{\frac{\lambda}{2}} - 2^{\lambda^\delta}}{2^{\lambda-\lambda^\delta} - 2^{\lambda^\delta}} < \frac{2^{\frac{\lambda}{2}}}{2^{\lambda-\lambda^\delta-1}} = 2^{-\frac{\lambda}{2}+\lambda^\delta+1} \quad .$$

□

Acknowledgments

We are grateful to Nir Bitansky and Zvika Brakerski for valuable discussions.

References

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

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

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

[BBBW83] Charles H Bennett, Gilles Brassard, Seth Breidbart, and Stephen Wiesner. Quantum cryptography, or unforgeable subway tokens. In *Advances in Cryptology*, pages 267–275. Springer, 1983.

[BCM+18] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 320–331. IEEE, 2018.

[BCP14] Elette Boyle, Kai-Min Chung, and Rafael Pass. On extractability obfuscation. pages 52–73, 2014.

[BDG19] Mathieu Bozzio, Eleni Diamanti, and Frédéric Grosshans. Semi-device-independent quantum money with coherent states. *Physical Review A*, 99(2):022336, 2019.

[BDGM20a] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Candidate io from homomorphic encryption schemes. 2020.

[BDGM20b] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Factoring and pairings are not necessary for io: Circular-secure lwe suffices. *IACR Cryptol. ePrint Arch.*, 2020:1024, 2020.

[BDS16] Shalev Ben-David and Or Sattath. Quantum tokens for digital signatures. *arXiv preprint arXiv:1609.09047*, 2016.

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

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

[BS20] Amit Behera and Or Sattath. Almost public quantum coins. *arXiv preprint arXiv:2002.12438*, 2020.

[BSS21] Amit Behera, Or Sattath, and Uriel Shinar. Noise-tolerant quantum tokens for mac. *arXiv preprint arXiv:2105.05016*, 2021.

[CLLZ21] Andrea Coladangelo, Jiahui Liu, Qipeng Liu, and Mark Zhandry. Hidden cosets and applications to unclonable cryptography. In *Annual International Cryptology Conference*, pages 556–584. Springer, 2021.

[DQV+21] Lalita Devadas, Willy Quach, Vinod Vaikuntanathan, Hoeteck Wee, and Daniel Wichs. Succinct lwe sampling, random polynomials, and obfuscation. *Cryptology ePrint Archive*, 2021.

[DSS16] Yfke Dulek, Christian Schaffner, and Florian Speelman. Quantum homomorphic encryption for polynomial-sized circuits. In *Annual International Cryptology Conference*, pages 3–32. Springer, 2016.

[FGH+12] 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*, pages 276–289, 2012.

[Gav12] Dmitry Gavinsky. Quantum money with classical verification. In *2012 IEEE 27th Conference on Computational Complexity*, pages 42–52. IEEE, 2012.

[GP21] Romain Gay and Rafael Pass. Indistinguishability obfuscation from circular security. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 736–749, 2021.

[HS20] Karol Horodecki and Maciej Stankiewicz. Semi-device-independent quantum money. *New Journal of Physics*, 22(2):023007, 2020.

[JLS18] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In *Annual International Cryptology Conference*, pages 126–152. Springer, 2018.

[LAF+09] Andrew Lutomirski, Scott Aaronson, Edward Farhi, David Gosset, Avinatan Hassidim, Jonathan Kelner, and Peter Shor. Breaking and making quantum money: toward a new quantum cryptographic protocol. *arXiv preprint arXiv:0912.3825*, 2009.

[Mah20] Urmila Mahadev. Classical homomorphic encryption for quantum circuits. *SIAM Journal on Computing*, (0):FOCS18–189, 2020.

[MS06] Michele Mosca and Douglas Stebila. Uncloneable quantum money. In *Canadian Quantum Information Students' Conference (CQISC)*, 2006.

[MS10] Michele Mosca and Douglas Stebila. Quantum coins. *Error-correcting codes, finite geometries and cryptography*, 523:35–47, 2010.

[PYJ+12] Fernando Pastawski, Norman Y Yao, Liang Jiang, Mikhail D Lukin, and J Ignacio Cirac. Unforgeable noise-tolerant quantum tokens. *Proceedings of the National Academy of Sciences*, 109(40):16079–16082, 2012.

[Rad19] Roy Radian. Semi-quantum money. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, pages 132–146, 2019.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009.

[Rob21] Bhaskar Roberts. Security analysis of quantum lightning. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 562–567. Springer, 2021.

[RZ20] Bhaskar Roberts and Mark Zhandry. Franchised quantum money. *URL: https://www. cs*, 2020.

[Unr16a] Dominique Unruh. Collapse-binding quantum commitments without random oracles. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 166–195. Springer, 2016.

[Unr16b] Dominique Unruh. Computationally binding quantum commitments. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 497–527. Springer, 2016.

[VZ21] Thomas Vidick and Tina Zhang. Classical proofs of quantum knowledge. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 630–660. Springer, 2021.

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

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

[Zha19] Mark Zhandry. Quantum lightning never strikes the same state twice. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 408–438. Springer, 2019.