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 1
1.1 Results 3

2 Technical Overview 3
2.1 The Lightning Strike Paradigm and Bolt Verifiability 3
2.2 Our Technique - An Alternative Lightning Bolt 6
2.3 Security in the Standard Model 8

3 Preliminaries 10
3.1 Indistinguishability Obfuscation 11
3.2 Leveled Hybrid Quantum Fully Homomorphic Encryption 13
3.3 Signature Schemes 14
3.4 Public-key Semi-Quantum Money 15

4 Public-Key Semi-Quantum Money Construction 16
4.1 Correctness 16

5 Security Proof 21
5.1 Security against Counterfeiting and Sabotage 23
5.2 Supplementary Claims 26

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.*

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

Let us recall the known methods for classical delegation of unclonable state generation. Specifically, we consider a scenario where a classical delegator D interacts with a quantum receiver Q and at the end of the interaction Q has a single copy of a quantum state $|\psi\rangle$, and D will know what the state is i.e. D will have some classical string s that uniquely identifies $|\psi\rangle$. The unclonability guarantee will say that for any quantum polynomial-time Q^* interacting with D, Q^* cannot generate *two* copies of $|\psi\rangle$.

A known template to classically delegate unclonable state sampling is that D samples a quantum circuit $G \leftarrow \mathcal{G}$ from some large distribution of possible circuits. While $\mathcal{G}$ is a distribution on circuits,

[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.

each circuit G defines a distribution Ψ_G on quantum states in the following way: G outputs a 2λ-qubit state in two λ-qubit registers (A, B). When wanting to sample from Ψ_G, Q executes G and measures B at the end of computation, to get some measurement outcome $\beta \in \{0,1\}^\lambda$. After the measurement, A collapses to some marginal quantum state $|\psi_\beta\rangle$ - this state is the sample from Ψ_G. The hope is that G is such a circuit that given both β and a single copy of $|\psi_\beta\rangle$, it is hard to generate an additional copy of the state $|\psi_\beta\rangle$ in register A.

We think of this process of Q computing G as the formation of a lightning, and on the act of measuring register B as a lightning strike - a natural probabilistic event with outcome that is unique with high probability[5]. The remaining quantum state $|\psi_\beta\rangle$ is called the lightning bolt, and β is called the identifier.

There are natural examples for quantum computations G that generate lightning strikes. A known cryptographic example is that $\mathcal{G}$ is a family of collision resistant functions, and G is computing a collision-resistant function H in superposition. The register A is the input register to H and B is the output register of H. Measuring B we get a uniform image $y \in \{0,1\}^\lambda$ of H and register A collapses to a uniform superposition $|H_y\rangle := \sum_{x:H(x)=y} |x\rangle$ of the y-preimages in H. The sampled state $|H_y\rangle$ is unclonable, because if we managed to generate two copies of the state in A 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. So to conclude this part, the delegator D can sample H and send it to Q, which can then compute it in superposition and generate the identifier $\beta := y$ and bolt $|\psi_\beta\rangle := |H_y\rangle$.

The Problem of Lightning Bolt Verification. A classical delegator D that lets the quantum Q generate lightning strikes is not a problem, but enabling 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 the following protocol template:

1. D samples $G \leftarrow \mathcal{G}$ and sends G to Q.
2. Q computes $(A, B) \leftarrow G$, measures register B to get $(|\psi_\beta\rangle, \beta)$, and sends β to D.
3. D sends a classical description of V, a verification circuit that accepts $|\psi_\beta\rangle$.

For now, we will think of V as an ideal obfuscation of some classical circuit, that is, the delegator enables quantum oracle access to some classical efficient function. We will later move to public verification in the standard model, without oracles. The unclonability of the protocol will guarantee that no quantum polynomial-time Q* can end up with two quantum states that pass the verification of V. That is, Q samples a state that is unclonable and publicly verifiable.

Keeping in mind the previous example of hash functions, given H and the image y, we don't know what D can classically send to Q in order to allow the classical verification of the quantum state $|H_y\rangle$. Given H, y, one can check that the state in register A is *some* superposition of preimages of y, by computing H in superposition with input register A and watching the output y. The challenge is to check the quantumness of register A 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",

[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].

for example, if it is a subspace state. Specifically, the circuit G is such that when register B is measured to get β, register A collapses to 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 $|S_\beta\rangle$ in register A can be verified with quantum oracle access to the classical membership functions for the subspace S_β and its dual $S_\beta^\perp$:

1. Execute on register A membership check for the subspace S_β in superposition.
2. Execute Quantum Fourier Transform (QFT) on register A.
3. Execute on register A 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 delegator D knows how to efficiently compute the classical circuits for membership checks for the two subspaces S_β, $S_\beta^\perp$ given β, verification (with respect to oracles) is possible. Consequently, a second "trapdoor property" of G will solve the problem. Specifically, a satisfying property is that when we sample $G \leftarrow \mathcal{G}$ we can sample G along with a classical trapdoor td that allows, given any β in the support of G, to efficiently compute a succinct representation of the subspaces S_β, $S_\beta^\perp$.

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 quantum circuit and a trapdoor $(\mathsf{td}, G) \leftarrow \mathcal{G}$ such that when computed, and register B is measured to get β, register A collapses to a quantum state exponentially close in trace distance to $|x_0^\beta\rangle + |x_1^\beta\rangle$. The strings $x_0^\beta, x_1^\beta \in \{0,1\}^\lambda$ are called the claw of β, and the security of the NTCF asserts that for every β in the support of the circuit G, 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 $|x_0^\beta\rangle + |x_1^\beta\rangle$, 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 trapdoor property of NTCFs guarantee that given the trapdoor td, 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 delegator can enable verification of claw states using the trapdoor td.

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 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 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 C involves changing the pad from the initial x, z to some other x', z':

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

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 C. For example, for any C 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 C such that when given along with a QFHE encryption $(|s\rangle^{(x,z)}, \mathsf{ct}_{x,z})$ of any string $s \in \{0,1\}^\lambda$, the following process generates a lightning strike:

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

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

The quantum circuit C 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 $|s\rangle \otimes \left(|0, 0^\lambda\rangle + |1, s\rangle\right)$. C 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 $(|s\rangle^{(x,z)}, \mathsf{ct}_{x,z})$ 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,

$$C\left(|s\rangle\right)^{(x',z')} \otimes C\left(|s\rangle\right)^{(x',z')} \otimes \mathsf{ct}_{x',z'}$$
$$= Z^{\otimes z'} \cdot \Big(|x' + 0^{1+\lambda}\rangle + |x' + (1,s)\rangle\Big) \otimes Z^{\otimes z'} \cdot \Big(|x' + 0^{1+\lambda}\rangle + |x' + (1,s)\rangle\Big) \otimes \mathsf{ct}_{x',z'} \quad .$$

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:

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

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 delegator D, 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 let the adversary sample copies of the state with noticeable probability. Indeed, we have reason to believe that increasing the dimension of the subspace can aid public verification. In their seminal work, Aaronson and Christiano [AC12] suggest a uniform superposition over a hidden random subspace S as the money state, and show that when the dimensions of S and $S^\perp$ are both $\lambda/2$, any adversary that is given $|S\rangle := \sum_{u \in S} |u\rangle$ and quantum oracle access to the classical membership functions S, $S^\perp$ cannot clone the state. In particular, the subspace S stays hidden.

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 C we homomorphically evaluated earlier in order to create a bolt. Under the encryption, given input $s \in \{0,1\}^\lambda$ what the circuit C 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 when the specific quantum circuit homomorphically evaluated is C, than the state is unclonable. This proof only uses the fact that C 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 C 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

[6] The circuit C 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 $S = \{0^\lambda, s\}$.

$s \in \{0,1\}^\lambda$, 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 $C(|\mathbf{M}_S\rangle)^{(x',z')} = \sum_{u\in S}(-1)^{\langle z',u\rangle}|x'+u\rangle$ 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 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 delegator D samples a random $\frac{\lambda}{2}$-dimensional subspace $S \subseteq \{0,1\}^\lambda$ and sends the QFHE encryption $(|\mathbf{M}_S\rangle^{(x,z)}, \mathsf{ct}_{x,z})$. The quantum receiver computes,

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

to generate a lightning bolt, and sends $\mathsf{ct}_{x',z'}$ to D. By decrypting $(x',z') = \mathsf{QHE.Dec}(\mathsf{ct}_{x',z'})$, D 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 D enables public verification of the bolt in the standard model (without ideal obfuscation), and second, how given this public verification in the standard model the state is still unclonable[7].

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

Subspace hiding 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 delegator 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 a means for public quantum verification. We examine this possibility next.

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

[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.

2. There is no delegator i.e. there is a 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 reduction we can 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 original pad x, z but does not have control over the actual 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 $\mathsf{O}_{S+x'}$, $\mathsf{O}_{S^{\perp}+z'}$ without knowing the QFHE secret key that is used in the original protocol to decrypt $(x', z') = \mathsf{QHE.Dec}(\mathsf{fhek}, \mathsf{ct}_{(x',z')})$. At this point we get stuck: The subspace hiding guarantee lets us swap $\mathsf{O}_{S+x'}$, $\mathsf{O}_{S^{\perp}+z'}$ with $\mathsf{O}_{T_0+x'}$, $\mathsf{O}_{T_1+z'}$ for random superspaces T_0, T_1, and still we do not know how to send something indistinguishable from any of the above obfuscations without knowing the actual pads x', z'.

Knowing the Pads Versus Containing the Pads. We suggest a tweak to the original scheme, and a stronger subspace hiding guarantee. We will explain how the stronger subspace hiding follows from the same assumptions as in [Zha19]. We change the verification slightly: the first check $C_{S+x'}$ is swapped to $C_{(S,x')}$ i.e. membership check not in the coset $S + x'$ but in the subspace spanned by vectors in S and $\{x'\}$ (this subspace is $(S + x') \cup S$), and the dual check is also swapped analogously from $C_{S^{\perp}+z'}$ to $C_{(S^{\perp},z')}$.

How does this helps exactly? Recall that subspace hiding of indistinguishability obfuscators lets you go to a significantly larger superspace, as long as the initial subspace is not too large. So, $\mathsf{O}_{(S,x')}$ is indistinguishable from O_{T_0} where T_0 is a random high-dimensional superspace of (S, x') and $\mathsf{O}_{(S^{\perp},z')}$ is indistinguishable from from O_{T_1} where T_1 is a random high-dimensional superspace of $(S^{\perp}, z')$. It follows that if we can simulate O_{T_0}, O_{T_1} we can perform the reduction. The helping part here is that in order to simulate these obfuscations, we don't need to *know* the strings x', z' but only let the corresponding subspaces T_0, T_1 to "catch" them. So, to simulate the obfuscation the reduction can guess T_0, T_1 subject only to $S \subseteq T_0$, $S^{\perp} \subseteq T_1$ (rather than to $(S, x') \subseteq T_0$, $(S^{\perp}, z') \subseteq T_1$ as in the original protocol) and hope that $(x' \in T_0) \wedge (z' \in T_1)$. If this is the case then the obfuscations distribute the same and our reduction should check.

Recalling the subspace hiding guarantee from earlier, this isn't helpful as it is. T_0 can only be of dimension $(1 - \epsilon') \cdot \lambda$ for a constant $\epsilon' \in (0, 1)$, which means its fraction in the space of all strings $\{0,1\}^{\lambda}$ is exponentially small $\frac{2^{(1-\epsilon')\cdot\lambda}}{2^{\lambda}} = 2^{-\epsilon'\cdot\lambda} = 2^{-O(\lambda)}$. The same is true for T_1. However, by looking at the actual proof of the subspace hiding property of indistinguishability obfuscators in [Zha19] (proof of Theorem 6.3), 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 T_0, T_1 contain the corresponding x', z' ($(x' \in T_0) \wedge (z' \in T_1)$ 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 $C_{(S,x')}, C_{(S^{\perp},z')}$.

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 $\mathbf{B} \in \{0,1\}^{\lambda-\dim(S)}$ 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

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

  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 $y \in \{0,1\}^{\lambda-\dim(S)}$,

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

  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 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') that are encrypted in the classical encryption* $\mathsf{ct}_{(x',z')}$.

The scheme satisfies the following.

- **Quantum Semantic 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:** *If the homomorphic evaluation is executed honestly $|\phi\rangle^{(x',z')}, \mathsf{ct}_{(x',z')} \leftarrow \mathsf{QHE.Eval}\left((|\psi\rangle^{(x,z)}, \mathsf{ct}_{(x,z)}), C\right)$ and the size of C is bounded by ℓ the parameter used for the generation of the secret key fhek, then the state $|\phi\rangle$ has exponentially small trace distance from $C(|\psi\rangle)$.*

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 can be based on assuming Decisional LWE for quantum computers, with sub-exponential indistinguishability.

3.3 Signature Schemes

We use signature schemes that are secure against quantum polynomial-time attacks.

Definition 3.3 (Signature Scheme)**.** *A signature scheme consists of 3 classical algorithms* ($\mathsf{Sig.Gen}$, $\mathsf{Sig.Sign}$, $\mathsf{Sig.Ver}$) *with the following syntax.*

- $(\mathsf{pk}_{\mathsf{Sig}}, \mathsf{sk}_{\mathsf{Sig}}) \leftarrow \mathsf{Sig.Gen}(1^\lambda)$: *The key generation algorithm is a PPT that takes as input a security parameter and outputs a pair of public verification key $\mathsf{pk}_{\mathsf{Sig}}$ and secret signing key $\mathsf{sk}_{\mathsf{Sig}}$.*

- $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_{\mathsf{Sig}}, m)$: *The signature algorithm is a PPT that takes as input a secret signing key $\mathsf{sk}_{\mathsf{Sig}}$ and a message $m \in \{0,1\}^*$ and outputs a signature σ.*

- $\mathsf{Sig.Ver}(\mathsf{pk}_{\mathsf{Sig}}, m, \sigma) \in \{0,1\}$: *The verification algorithm is a deterministic algorithm such that for the public verification key, a message $m \in \{0,1\}^*$ and a candidate signature σ for m outputs a bit signalling whether or not the signature was successful.*

The algorithms have the following properties.

- **Correctness:** *for any message $m \in \{0,1\}^*$,*

$$\Pr\left[\mathsf{Sig.Ver}(\mathsf{pk}_{\mathsf{Sig}}, m, \sigma) = 1 \;\middle|\; (\mathsf{pk}_{\mathsf{Sig}}, \mathsf{sk}_{\mathsf{Sig}}) \leftarrow \mathsf{Sig.Gen}(1^\lambda), \sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_{\mathsf{Sig}}, m)\right] = 1 \ .$$

- **Unforgeability against Chosen Plaintext Attack:** *For any oracle aided quantum polynomial-time adversary $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ there exists a negligible function $\mathrm{negl}(\cdot)$ such that*

$$\Pr\left[\mathsf{Sig.Ver}(\mathsf{pk}_{\mathsf{Sig}}, m^*, \sigma^*) = 1 \;\middle|\; \begin{array}{c} (m^*, \sigma^*) \leftarrow \mathcal{A}^{\mathsf{Sig.Sign}(\mathsf{sk}_{\mathsf{Sig}}, \cdot)}(\mathsf{pk}_{\mathsf{Sig}}), \\ m^* \notin Q \end{array}\right] \leq \mathrm{negl}(\lambda) \ ,$$

  where $(\mathsf{sk}_{\mathsf{Sig}}, \mathsf{pk}_{\mathsf{Sig}}) \leftarrow \mathsf{Sig.Gen}(1^\lambda)$ and Q is the set of queries that $\mathcal{A}$ makes to $\mathsf{Sig.Sign}(\mathsf{sk}_{\mathsf{Sig}}, \cdot)$.

Instantiations. Signature schemes with quantum security are known based on assuming the Learning with Errors Assumption [BZ13].

3.4 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 3.4 (Public-key semi-quantum money)**.** *A public-key semi-quantum money scheme consists of algorithms* $(\mathsf{Gen}, \mathsf{BankMint}, \mathsf{RecMint}, \mathsf{QV}, \mathsf{GenCert}, \mathsf{CV})$ *with the following syntax.*

- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$: *A PPT algorithm that gets as input the security parameter and samples a pair of classical keys, a public verification key and a secret generation key.*
- $|\$\rangle \leftarrow \langle \mathsf{BankMint}(\mathsf{sk}), \mathsf{RecMint}(\mathsf{pk})\rangle$: *a classical-communication protocol between a classical algorithm* $\mathsf{BankMint}(\mathsf{sk})$ *and a quantum algorithm* $\mathsf{RecMint}(\mathsf{pk})$*. At the end of interaction the receiver has a quantum banknote* $|\$\rangle$*.*
- $(b, |\$'\rangle) \leftarrow \mathsf{QV}(\mathsf{pk}, |\$\rangle)$: *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 classical 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* $\lambda \in \mathbb{N}$*,*

$$\Pr\left[(1, |\$'\rangle) \leftarrow \mathsf{QV}(\mathsf{pk}, |\$\rangle) \;\middle|\; \begin{array}{c} (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda), \\ |\$\rangle \leftarrow \langle \mathsf{BankMint}(1^\lambda, \mathsf{sk}), \mathsf{RecMint}(1^\lambda)\rangle \end{array}\right] \geq 1 - \mathrm{negl}(\lambda) \enspace .$$

- **Security:** *Let* $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ *a quantum polynomial-time algorithm, and consider the following game:*
  - *A key pair* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$ *is sampled,* $\mathcal{A}$ *gets* pk *and interacts the bank minting protocol* $\mathsf{BankMint}(\mathsf{sk})$*. At the end,* $\mathcal{A}$ *outputs a quantum state* BN^**.*

  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, the definition captures multi-session security without the loss of generality by a generic and trivial transformation. 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 (nor quantum or classical sabotage). Like in the definition from [Rad19], multi-session security requires the bank to keep a database of banknotes that have been previously destroyed by the classical certificate of destruction mechanism.

A scheme with multi-session security can be obtained by the following use of the single-session-secure definition above. The initial key generation is generating keys for a signature scheme $(\mathsf{pk}_{\mathsf{Sig}}, \mathsf{sk}_{\mathsf{Sig}}) \leftarrow \mathsf{Sig.Gen}(1^{\lambda})$. At the beginning of every session of classical minting, the bank uses the single-session definition above, sends the public key (of the single session definition) along with the first message, and signs on the public key of the one-session bank, using its many-session secret key $\mathsf{sk}_{\mathsf{Sig}}$, at the end of each minting protocol. Breaking the security of the original, single-session-secure scheme can be reduced to breaking the security of the multi-session scheme, by the fact that minting sessions are independent.

4 Public-Key Semi-Quantum Money Construction

In this section we present our construction of a Public-key Semi-Quantum Money scheme (Definition 3.4), and proof of correctness.

Ingredients and notation:

- A quantum hybrid fully homomorphic encryption scheme $(\mathsf{QHE.Gen}, \mathsf{QHE.Enc}, \mathsf{QHE.OTP}, \mathsf{QHE.Dec}, \mathsf{QHE.QOTP}, \mathsf{QHE.Eval})$, with sub-exponential advantage security (Definition 3.2).
- An indistinguishability obfuscation scheme iO (Definition 3.1).
- A signature scheme $(\mathsf{Sig.Gen}, \mathsf{Sig.Sign}, \mathsf{Sig.Ver})$ for classical messages (Definition 3.3).

We describe the scheme in Figure 1, this includes the initial public key generation, the minting protocol, the quantum public verification of a bank note and the classical verification of a classical certificate of destruction.

4.1 Correctness

We prove that our scheme is correct, that is, 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 $1-\mathrm{negl}(\lambda)$ and generates a valid classical certificate of destruction with probability $1-\mathrm{negl}(\lambda)$.

Claim 4.1. *At the end of a successful execution of the minting protocol, the bank note has negligible trace distance from the state,*

$$\left(\sigma, (\mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S), |S\rangle^{(x,z)}\right) \ ,$$

where,

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

Proof. In case the protocol ends successfully, it follows readily from the perfect correctness of the Indistinguishability obfuscation and the statistical correctness of the QFHE that the output state has a negligible trace distance to the state $C(\mathbf{M}) = |S\rangle^{(x,z)}$. □

Claim 4.2. *A successful quantum verification procedure* $\mathsf{QV}\left(\mathsf{pk}_{\mathsf{Sig}}, (\sigma, \mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S, \mathsf{BN})\right)$ *acts as a projection of the state in* BN *on the space spanned by the* 4 *following vectors,*

$$|S_{00}\rangle := \sum_{u\in S}|u\rangle \ , \ |S_{01}\rangle := \sum_{u\in S}|x+u\rangle \ ,$$

$$|S_{10}\rangle := \sum_{u\in S}(-1)^{\langle z,u\rangle}|u\rangle \ , \ |S_{11}\rangle := \sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u\rangle \ .$$

Proof. It is easy to verify that the quantum verification procedure acts as the identity on each of the vectors $|S_{00}\rangle$, $|S_{01}\rangle$, $|S_{10}\rangle$, $|S_{11}\rangle$. We'll show that any vector that passes verification is projected to the span of these vectors.

Assume that the signature check passes, and we perform the first part of quantum verification, i.e. executing $\mathsf{O}_{S,x}(\mathsf{BN})$. After passing this check successfully, the state in BN is some superposition of vectors in (S,x), with some phases (possibly entangled with an external system). More precisely, if we isolate the state (and trace other registers) it can be written as,

$$\sum_{u\in S}\alpha_u|u\rangle + \sum_{u\in S}\beta_u|x+u\rangle + \sum_{u\in S}\gamma_u(-1)^{\langle z,u\rangle}|u\rangle + \sum_{u\in S}\delta_u(-1)^{\langle z,u\rangle}|x+u\rangle \ ,$$

for amplitudes $\sum_{u\in S}|\alpha_u|^2+|\beta_u|^2+|\gamma_u|^2+|\delta_u|^2=1$.

After executing $H^{\otimes\lambda}$, the state is,

$$\begin{aligned}
&\sum_{u\in S}\alpha_u\left(\sqrt{2^{-\lambda}}\cdot\sum_{v\in\{0,1\}^\lambda}(-1)^{\langle u,v\rangle}|v\rangle\right)\\
&+\sum_{u\in S}\beta_u\left(\sqrt{2^{-\lambda}}\cdot\sum_{v\in\{0,1\}^\lambda}(-1)^{\langle x+u,v\rangle}|v\rangle\right)\\
&+\sum_{u\in S}\gamma_u(-1)^{\langle z,u\rangle}\left(\sqrt{2^{-\lambda}}\cdot\sum_{v\in\{0,1\}^\lambda}(-1)^{\langle u,v\rangle}|v\rangle\right)\\
&+\sum_{u\in S}\delta_u(-1)^{\langle z,u\rangle}\left(\sqrt{2^{-\lambda}}\cdot\sum_{v\in\{0,1\}^\lambda}(-1)^{\langle x+u,v\rangle}|v\rangle\right)
\end{aligned}$$

$$
\begin{aligned}
= &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle u,v\rangle}\cdot\alpha_u\right)\cdot|v\rangle\\
+ &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle x+u,v\rangle}\cdot\beta_u\right)\cdot|v\rangle\\
+ &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle u,v\rangle}(-1)^{\langle z,u\rangle}\cdot\gamma_u\right)\cdot|v\rangle\\
+ &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle x+u,v\rangle}(-1)^{\langle z,u\rangle}\cdot\delta_u\right)\cdot|v\rangle
\end{aligned}
$$

$$
\begin{aligned}
= &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle u,v\rangle}\cdot\alpha_u\right)\cdot|v\rangle\\
+ &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle u,v\rangle}\cdot(-1)^{\langle x,v\rangle}\cdot\beta_u\right)\cdot|v\rangle\\
+ &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle u,v\rangle}\cdot(-1)^{\langle z,u\rangle}\cdot\gamma_u\right)\cdot|v\rangle\\
+ &\sum_{v\in\{0,1\}^\lambda}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle u,v\rangle}\cdot(-1)^{\langle x,v\rangle}\cdot(-1)^{\langle z,u\rangle}\cdot\delta_u\right)\cdot|v\rangle\ .
\end{aligned}
$$

Now, the state passes the second verification circuit $\mathsf{O}_{S^\perp,z}(\mathsf{BN})$. This means that our state is the same as above, only that the sum is only over $v\in(S^\perp,z)$ rather than over $v\in\{0,1\}^\lambda$. We will fist calculate what happens with the part of the sum that is over $S^\perp$. We will later consider the part of the sum that is for $v\in S^\perp+z$.

$$
\begin{aligned}
= &\sum_{v\in S^\perp}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}\alpha_u\right)\cdot|v\rangle\\
+ &\sum_{v\in S^\perp}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle x,v\rangle}\cdot\beta_u\right)\cdot|v\rangle\\
+ &\sum_{v\in S^\perp}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle z,u\rangle}\cdot\gamma_u\right)\cdot|v\rangle\\
+ &\sum_{v\in S^\perp}\left(\sqrt{2^{-\lambda}}\cdot\sum_{u\in S}(-1)^{\langle x,v\rangle}\cdot(-1)^{\langle z,u\rangle}\cdot\delta_u\right)\cdot|v\rangle
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} \alpha_u \right) \cdot |v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} \beta_u \right) \cdot (-1)^{\langle x,v\rangle} \cdot |v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle z,u\rangle} \cdot \gamma_u \right) \cdot |v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle z,u\rangle} \cdot \delta_u \right) \cdot (-1)^{\langle x,v\rangle} \cdot |v\rangle
\end{aligned}
$$

$$
\begin{aligned}
:= &\sum_{v \in S^\perp} A \cdot |v\rangle \\
+ &\sum_{v \in S^\perp} B \cdot (-1)^{\langle x,v\rangle} \cdot |v\rangle \\
+ &\sum_{v \in S^\perp} C \cdot |v\rangle \\
+ &\sum_{v \in S^\perp} D \cdot (-1)^{\langle x,v\rangle} \cdot |v\rangle \ ,
\end{aligned}
$$

$$
:= (A + C) \cdot H^{\otimes\lambda} \cdot |S_{00}\rangle + (B + D) \cdot H^{\otimes\lambda} \cdot |S_{01}\rangle \ ,
$$

for some complex numbers A, B, C, D, that are independent of v (some of them might be zero).

Similarly to the above computation, we look at the sum over $S^\perp$ with a z shift:

$$
\begin{aligned}
&= \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle u,z\rangle} \cdot \alpha_u \right) \cdot |z + v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle u,z\rangle} \cdot (-1)^{\langle x,v\rangle} \cdot \beta_u \right) \cdot |z + v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle u,z\rangle} \cdot (-1)^{\langle z,u\rangle} \cdot \gamma_u \right) \cdot |z + v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle u,z\rangle} \cdot (-1)^{\langle x,v\rangle} \cdot (-1)^{\langle z,u\rangle} \cdot \delta_u \right) \cdot |z + v\rangle
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle u,z\rangle} \cdot \alpha_u \right) \cdot |z + v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} (-1)^{\langle u,z\rangle} \cdot \beta_u \right) \cdot (-1)^{\langle x,v\rangle} \cdot |z + v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} \gamma_u \right) \cdot |z + v\rangle \\
&+ \sum_{v \in S^\perp} \left(\sqrt{2^{-\lambda}} \cdot \sum_{u \in S} \delta_u \right) \cdot (-1)^{\langle x,v\rangle} |z + v\rangle
\end{aligned}
$$

$$
\begin{aligned}
:= &\sum_{v \in S^\perp} A' \cdot |z+v\rangle \\
+ &\sum_{v \in S^\perp} B' \cdot (-1)^{\langle x,v \rangle} \cdot |z+v\rangle \\
+ &\sum_{v \in S^\perp} C' \cdot |z+v\rangle \\
+ &\sum_{v \in S^\perp} D' \cdot (-1)^{\langle x,v \rangle} \cdot |z+v\rangle \ ,
\end{aligned}
$$

$$
:= (A'+C') \cdot H^{\otimes \lambda} \cdot |S_{10}\rangle + (B'+D') \cdot H^{\otimes \lambda} \cdot |S_{11}\rangle \ ,
$$

where as before, A', B', C', D' are complex numbers independent of v.

Considering both the analysis for summing $v \in S^\perp$ and $v \in S^\perp + z$, we get that after successful verification of the second circuit $\mathsf{O}_{S^\perp,z}$, the state in BN (after tracing out other registers) is,

$$
H^{\otimes \lambda} \cdot \big((A+C) \cdot |S_{00}\rangle + (B+D) \cdot |S_{01}\rangle + (A'+C') \cdot |S_{10}\rangle + (B'+D') \cdot |S_{11}\rangle \big) \ .
$$

At the end of quantum verification an additional $H^{\otimes \lambda}$ is executed, which makes the state exactly be in the span of $\{|S_{00}\rangle, |S_{01}\rangle, |S_{10}\rangle, |S_{11}\rangle\}$, and our proof is finished. □

Proposition 4.1. *The scheme presented in Protocol 1 has statistical correctness (Definition 3.4).*

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 of the QFHE encryption is $|S\rangle^{(x,z)}$. This means that if $x \in S$, then the state $|S\rangle^{x,z}$ is $|S\rangle^{(0^\lambda,z)}$. It follows that the state in BN has a negligible trace distance from such state, and measuring BN in the computational basis yields, with a noticeable probability, $s \in (S \setminus T_1^\perp)$. This contradicts Claim 5.2, so, with probability $1 - \mathrm{negl}(\lambda)$ we have $x \notin S$.

Now, assume that the protocol ended successfully. By the correctness of the signature scheme, the signature check passes successfully. Let $x, z \in \{0,1\}^\lambda$ the strings that $\mathsf{BankMint}$ obtains by decryption, at step 3 of the minting protocol.

- In Claim 4.2 we saw that the quantum verification procedure acts as a projector on the space spanned by $\{|S_{00}\rangle, |S_{01}\rangle, |S_{10}\rangle, |S_{11}\rangle\}$, which means that the state in BN passes quantum verification of QV with probability $1 - \mathrm{negl}(\lambda)$, as it has negligible trace distance from $|S_{11}\rangle := \sum_{u \in S} (-1)^{\langle z,u \rangle} |x+u\rangle$.

- Since a measurement to $|S\rangle^{(x,z)}$ yields an element $\mathsf{crt} \in S + x$ with probability 1, then with probability $1 - \mathrm{negl}(\lambda)$ a measurement $\mathsf{crt} \leftarrow \mathsf{BN}$ satisfies $\mathsf{crt} \in S + x$. Consequently, with probability $1 - \mathrm{negl}(\lambda)$, measuring BN yields crt such that $\mathsf{O}_{S,x}(\mathsf{crt}) = 1$.

  Additionally, the protocol is defined such that it ends successfully only if $x \notin S$. This means that measuring $|S\rangle^{(x,z)}$ yields crt such that $\mathsf{crt} \notin S$ with probability 1. It follows that measuring BN yields crt such that with probability $1 - \mathrm{negl}(\lambda)$ we have $\mathsf{crt} \notin S$. Consequently, with probability $1 - \mathrm{negl}(\lambda)$, measuring BN satisfies $\mathsf{O}_S(\mathsf{crt}) = 0$.

  It follows that with probability $1 - \mathrm{negl}(\lambda)$, a measurement $\mathsf{crt} \leftarrow \mathsf{BN}$ passes the classical verification of CV.

Overall, with probability $1 - \mathrm{negl}(\lambda)$ the protocol ends successfully, and with probability $1 - \mathrm{negl}(\lambda)$ both quantum and classical certificate verification pass successfully. □

5 Security Proof

In this section we will argue that the scheme is secure, that is, under the security of our ingredient primitives, there is no quantum polynomial time adversary that can counterfeit, sabotage quantum verification or sabotage classical verification.

Lemma 5.1. *(Main Security Lemma) For every quantum polynomial time adversary $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$, the probability of the following event $E = \{E_\lambda\}_{\lambda \in \mathbb{N}}$ is negligible:*

- *The adversary $\mathcal{A}$ interacts with the bank during the minting protocol, let S the subspace picked by the bank and let x, z be the decrypted QOTP keys obtained by the bank at step 3 of the minting protocol.*
- *At the end of the protocol $\mathcal{A}$ outputs an λ-qubit register* BN.
- *E is the event where the projection of* BN *on the space spanned by $\{|S_{00}\rangle, |S_{01}\rangle\} := \{\sum_{u \in S} |u\rangle,\ \sum_{u \in S} (-1)^{\langle z,u \rangle} |u\rangle\}$ is successful.*

Proof. Let $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ a quantum polynomial time adversary that succeeds in event E with some noticeable probability $\varepsilon = \{\varepsilon_\lambda\}_{\lambda \in \mathbb{N}}$. We will show how to use $\mathcal{A}$ in order to break the sub-exponential security of the QFHE. We define the hybrid experiment Hyb_1 to be exactly the experiment described above where $\mathcal{A}$ succeeds in E. We next describe a sequence of hybrid experiments, consequently arriving to a hybrid experiment that is directly useful for breaking the security of the QFHE.

Hyb_2 : 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 $\mathsf{O}_{S,x}$, $\mathsf{O}_{S^\perp,z}$, O_S at step 3 of the minting protocol, the obfuscations $\mathsf{O}_{S,x}$, O_S are changed: We sample a random $\frac{3\lambda}{4}$-dimensional superspace $\tilde{T}_0 \subseteq \{0,1\}^\lambda$ of S, and an $(\lambda - \lambda^\delta)$-dimensional superspace $T_0 \subseteq \{0,1\}^\lambda$ of $(\tilde{T}_0, x)$. We send $\mathsf{O}_{T_0} \leftarrow \mathsf{iO}(\mathbf{M}_{T_0})$ instead of $\mathsf{O}_{S,x} \leftarrow \mathsf{iO}(\mathbf{M}_{S,x})$, and $\mathsf{O}_{\tilde{T}_0} \leftarrow \mathsf{iO}(\mathbf{M}_{\tilde{T}_0})$ instead of $\mathsf{O}_S \leftarrow \mathsf{iO}(\mathbf{M}_S)$

Note that for the rest of Hyb_1, after computing (S, x) from the receiver's message in the minting protocol (step 2), we can get the obfuscated circuits (which is either $(\mathsf{O}_{S,x}, \mathsf{O}_S)$ or $(\mathsf{O}_{T_0}, \mathsf{O}_{\tilde{T}_0})$) from an outside source, as we do not use the knowledge of T_0, $\tilde{T}_0$, only the subspace S and the decrypted pad z are both used to verify the success of the experiment at the end of step 3 of Hyb_1. The outputs of the experiments are indistinguishable due to Claim 5.1. This means in particular that the success probabilities are negligibly close $\geq \varepsilon - \mathrm{negl}(\lambda)$.

Hyb_3 : This hybrid is the same as Hyb_2 only that we now swap the obfuscation of the dual subspace $(S^\perp, z)$. Identical to Hyb_2, with the only difference is that when the bank returns the obfuscations O_{T_0}, $\mathsf{O}_{S^\perp,z}$, $\mathsf{O}_{\tilde{T}_0}$, the obfuscation $\mathsf{O}_{S^\perp,z}$ is changed: Instead of obfuscating the row span of $\mathbf{M}_{S^\perp,z}$, we choose a random superspace $(S^\perp, z) \subseteq T_1 \subseteq \{0,1\}^\lambda$ of dimension $\lambda - \lambda^\delta$ with a generating matrix $\mathbf{M}_{T_1}$. We send $\mathsf{O}_{T_1} \leftarrow \mathsf{iO}(\mathbf{M}_{T_1})$ instead of $\mathsf{O}_S \leftarrow \mathsf{iO}(\mathbf{M}_{S^\perp,z})$.

Similarly to the above explanation of why the success probabilities of Hyb_1 and Hyb_2 are negligibly close, the success probabilities of Hyb_2 and Hyb_3 are negligibly close. The success probability of Hyb_3 is thus $\geq \varepsilon - \mathrm{negl}(\lambda)$.

Hyb_4 : The following hybrid is going to be identical to Hyb_3, but executed in a different way. First, the process samples an intermediate random subspace $\tilde{T}_0 \subseteq \{0,1\}^\lambda$ of dimension $\frac{3 \cdot \lambda}{4}$, and an additional intermediate random subspace $\tilde{T}_1 \subseteq \{0,1\}^\lambda$ of dimension $\frac{3 \cdot \lambda}{4}$, subject to $\tilde{T}_1^\perp \subseteq \tilde{T}_0$ (this sampling can be done by sampling $\tilde{T}_1^\perp$ directly, by sampling a random $\frac{\lambda}{4}$-dimensional subspace of $\tilde{T}_0$).

Now, we sample a random subspace $S \subseteq \{0,1\}^\lambda$ subject to $\tilde{T}_1^\perp \subseteq S \subseteq \tilde{T}_0$, and carry on the interaction process between the bank and $\mathcal{A}$ regularly. Consider the step when the bank gets the message of $\mathcal{A}$ after step 2 of the minting protocol. The bank decrypts the classical FHE ciphertext to get the QOTP keys x, z. Now, we sample a random $(\lambda - \lambda^\delta)$-dimensional subspace $T_0 \subseteq \{0,1\}^\lambda$ subject to $(\tilde{T}_0, x) \subseteq T_0$ and a random $(\lambda - \lambda^\delta)$-dimensional subspace $T_1 \subseteq \{0,1\}^\lambda$ subject to $(\tilde{T}_1, z) \subseteq T_1$. We send the obfuscations $\mathsf{O}_{T_0} \leftarrow \mathsf{iO}(\mathbf{M}_{T_0})$, $\mathsf{O}_{T_1} \leftarrow \mathsf{iO}(\mathbf{M}_{T_1})$, $\mathsf{O}_{\tilde{T}_0} \leftarrow \mathsf{iO}(\mathbf{M}_{\tilde{T}_0})$ and continue regularly as in Hyb_3.

Note that the only change between the two experiments Hyb_3 and Hyb_4 is the *algorithm* to sample the subspaces $S, T_0, T_1, \tilde{T}_0, \tilde{T}_1$. However, all subspaces distribute identically as to how they distribute in Hyb_3. Since this is the only difference, the experiments are identical in their output and in particular in the success probability, which is $\geq \varepsilon - \text{negl}(\lambda)$.

Hyb_5 : We can take the sampling procedure of the subspaces described in Hyb_4 and perform an averaging argument on the sampling of the intermediate subspaces $\tilde{T}_0, \tilde{T}_1$, to take the two subspaces that maximize the success probability of Hyb_4. This means that there exist *fixed* subspaces $\tilde{T}_0, \tilde{T}_1$ for which that experiment is successful with probability $\geq \varepsilon - \text{negl}(\lambda)$. This process where the intermediate subspaces $\tilde{T}_0, \tilde{T}_1$ are fixed is defined to be Hyb_5.

Hyb_6 : In this hybrid experiment we perform the exact same experiment Hyb_5, but with the following changes to the operation of the experiment and definition of success:

- At the end of Hyb_5 we project the quantum register BN onto the span of $\{|S_{00}\rangle, |S_{01}\rangle\}$. In Hyb_6, instead of executing the projection, we simply measure BN in the standard basis. Denote by $s \in \{0,1\}^\lambda$ the measurement outcome.
- The experiment Hyb_6 is defined to be successful if $s \in \left(S \setminus \tilde{T}_1^\perp\right)$.

Let us understand the success probability of Hyb_6. We know that when executing Hyb_5, the probability that BN is successfully projected onto the span of $\{|S_{00}\rangle, |S_{01}\rangle\}$ at the end is $\geq \varepsilon - \text{negl}(\lambda)$. It is necessarily the case that the average amplitude of this span in BN just before the projection, is $\geq \sqrt{\varepsilon - \text{negl}(\lambda)}$. Recall that measuring either of the quantum states $|S_{00}\rangle$, $|S_{01}\rangle$ in the standard basis yields a *uniform* sample from the subspace S. It follows that with probability $\geq \varepsilon - \text{negl}(\lambda)$, the measurement outcome s distributes uniformly in the subspace S. It follows that the probability that $s \in \left(S \setminus \tilde{T}_1^\perp\right)$ is $\frac{|S \setminus \tilde{T}_1^\perp|}{|S|} = \frac{2^{\frac{\lambda}{2}} - 2^{\frac{\lambda}{4}}}{2^{\frac{\lambda}{2}}} > \frac{1}{2}$, and overall the probability that Hyb_6 is successful is $\frac{\varepsilon}{2} - \text{negl}(\lambda)$.

Hyb_7 : This experiment is identical to Hyb_6 with one change: in step 3 of the minting protocol, when the bank usually decrypts the QFHE classical part to get the QOTP keys x, z (and also checks that $x \notin S$), the process Hyb_7 does not decrypt to get x, z and instead it samples the subspaces T_0, T_1 *independently* of x, z. More precisely, T_0 is a random $(\lambda - \lambda^\delta)$-dimensional subspace subject only to $\tilde{T}_0 \subseteq T_0$ (rather than subject to $(\tilde{T}_0, x) \subseteq T_0$) and T_1 is a random $(\lambda - \lambda^\delta)$-dimensional subspace subject only to $\tilde{T}_1 \subseteq T_1$ (rather than subject to $(\tilde{T}_1, z) \subseteq T_1$).

First, notice that the first message of $\mathsf{BankMint}$ in the protocol distributes the same between Hyb_6 and Hyb_7, so the probability that $x \notin S$ is the same and is $\geq \varepsilon$. Observe that conditioned on the probabilistic event that

$$(x \in T_0) \wedge (z \in T_1) \ ,$$

the experiments Hyb_6 and Hyb_7 distribute exactly the same. Due to the fact that the dimension of each of T_0, T_1 is $\lambda - \lambda^\delta$ and that these subspaces are random, the probability that $(x \in T_0) \wedge (z \in T_1)$, for any x, z, is at least

$$\frac{|T_0|}{|\{0,1\}^\lambda|} \cdot \frac{|T_1|}{|\{0,1\}^\lambda|} = \left(\frac{2^{\lambda-\lambda^\delta}}{2^\lambda}\right)^2 = 2^{-2\cdot\lambda^\delta} \ .$$

Overall, the success probability of Hyb_7 is $\geq \left(\frac{\varepsilon}{2} - \text{negl}(\lambda)\right) \cdot 2^{-2\cdot\lambda^\delta} \geq 2^{-3\cdot\lambda^\delta}$.

At this point we use the success probability and structure of Hyb_7 to show that we can efficiently solve some hard problem with probability that should not be possible, under our computational assumptions. Recall that the QFHE has security against advantage $2^{-\lambda^{\delta'}}$ for quantum polynomial-time adversaries. By Claim 5.2, given any fixed subspaces $\tilde{T}_0, \tilde{T}_1^\perp \subseteq \{0,1\}^\lambda$ such that $\tilde{T}_0$ is of dimension $\frac{3\cdot\lambda}{4}$ and $\tilde{T}_1^\perp$ is a $\frac{\lambda}{4}$-dimensional subspace of $\tilde{T}_0$, there is no quantum polynomial-time algorithm that can get as input a QFHE encryption of a classical description of a random $\frac{\lambda}{2}$-dimensional subspace $\tilde{T}_1^\perp \subseteq S \subseteq \tilde{T}_0$ and find a string $s \in \left(S \setminus \tilde{T}_1^\perp\right)$ with probability $\geq 2^{-\lambda^{\delta'}+1} = 2^{-\lambda^{2\cdot\delta}+1}$.

Executing Hyb_7 does not require any knowledge on S nor (x, z), but only $\tilde{T}_0, \tilde{T}_1$ and the fact that S is subject to $\tilde{T}_0^\perp \subseteq S \subseteq \tilde{T}_0$. It follows that using $\mathcal{A}$ in the experiment Hyb_7 lets us find such vector $s \in \left(S \setminus \tilde{T}_1^\perp\right)$ with probability $\geq 2^{-3\cdot\lambda^\delta} > 2^{-\lambda^{2\cdot\delta}+1}$, in contradiction. □

5.1 Security against Counterfeiting and Sabotage

Security against counterfeiting. We next use our main Lemma 5.1 in order to prove that the scheme is secure against counterfeiting attacks.

Proposition 5.1 (Security against Counterfeiting)**.** *The public-key semi-quantum money scheme described in Protocol 1 has security against counterfeiting, according to Definition 3.4.*

Proof. Let $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ a quantum polynomial time adversary that succeeds in counterfeiting with some noticeable probability $\varepsilon = \{\varepsilon_\lambda\}_{\lambda\in\mathbb{N}}$. We will show how to use $\mathcal{A}$ and Lemma 5.1 to get a contradiction, that is, we will describe an adversary $\mathcal{A}' = \{\mathcal{A}'_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ that violates Lemma 5.1.

Whenever $\mathcal{A}$ counterfeits successfully it sends a quantum bank note $\left(\sigma^{(1)}, \mathsf{O}_{S,x}^{(1)}, \mathsf{O}_{S^\perp,z}^{(1)}, \mathsf{O}_S^{(1)}, \mathsf{BN}\right)$, and a classical certificate $\left(\sigma^{(2)}, \mathsf{O}_{S,x}^{(2)}, \mathsf{O}_{S^\perp,z}^{(2)}, \mathsf{O}_S^{(2)}, \mathsf{crt} \in \{0,1\}^\lambda\right)$ such that:

- $\sigma^{(1)}$ is a valid signature for $(\mathsf{O}_{S,x}^{(1)}, \mathsf{O}_{S^\perp,z}^{(1)}, \mathsf{O}_S^{(1)})$.
- $\sigma^{(2)}$ is a valid signature for $(\mathsf{O}_{S,x}^{(2)}, \mathsf{O}_{S^\perp,z}^{(2)}, \mathsf{O}_S^{(2)})$.
- The quantum register BN passes quantum verification with the circuits $(\mathsf{O}_{S,x}^{(1)}, \mathsf{O}_{S^\perp,z}^{(1)})$, and crt passes classical certificate verification with $(\mathsf{O}_{S,x}^{(1)}, \mathsf{O}_S^{(1)})$ respectively.

Recall that the bank signs on exactly one message, which is the original obfuscations $\mathsf{O}_{S,x}$, $\mathsf{O}_{S^\perp,z}$, O_S generated in the minting protocol. From the unforgeability property of the signature scheme it follows that with probability negligibly close to ε we have

$$(\mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S) = (\mathsf{O}_{S,x}^{(1)}, \mathsf{O}_{S^\perp,z}^{(1)}, \mathsf{O}_S^{(1)}) = (\mathsf{O}_{S,x}^{(2)}, \mathsf{O}_{S^\perp,z}^{(2)}, \mathsf{O}_S^{(2)})$$

and BN, crt pass quantum (classical, resp.) verification *using the original verification circuits*. Let us update the output of $\mathcal{A}$ such that it is only BN, crt, after performing the above verification. We next describe an experiment Exp.

Exp : The experiment starts with its first step, which is $\mathcal{A}$ interacting with the bank during the minting protocol procedure. At the end of step 1 of Exp, $\mathcal{A}$ outputs BN, crt as described above, after running on BN quantum verification with $\mathsf{O}_{S,x}$, $\mathsf{O}_{S^{\perp},z}$ and running on crt classical verification with $\mathsf{O}_{S,x}$, O_{S}. We would like that the next step of the experiment sets BN to hold $|S_{b,0}\rangle := \sum_{u\in S}(-1)^{\langle b\cdot z,u\rangle}|u\rangle$ (for some $b\in\{0,1\}$) with a noticeable amplitude, and we will later want to show that this happens with a noticeable probability. The second step of Exp is to sample $\alpha \leftarrow \{0,1\}$.

- If $\alpha = 0$ then leave BN unchanged.
- If $\alpha = 1$ then add crt to register BN.

The third and final step of the experiment Exp is to project the state in BN onto the space spanned by only $\{\sum_{u\in S}|u\rangle, \sum_{u\in S}(-1)^{\langle z,u\rangle}|u\rangle\}$. This is done by first measuring whether BN is in the row span of $\mathbf{M}_S$, then executing $H^{\otimes\lambda}$ on BN, and then measuring whether BN is in the row span of $\mathbf{M}_{S^{\perp},z}$. The experiment is successful iff the projective measurement succeeds.

The success probability of Exp**.** We will see what is the success probability of each of the three steps of Exp, conditioned on previous steps succeeding.

1. As explained in the beginning of the proof, the probability that $\mathcal{A}$ outputs BN, crt that pass quantum and classical (respectively) verifications with the original circuits is at least $\frac{\varepsilon}{2} > \varepsilon - \mathrm{negl}(\lambda)$, which means that the probability that the first step of Exp succeeds is also at least $\frac{\varepsilon}{2}$.

2. Assume that the first step of Exp was successful. As we saw in Claim 4.2, a successful quantum verification projects the state in BN to the space spanned by four vectors,

$$|S_{00}\rangle := \sum_{u\in S}|u\rangle \ , \ |S_{01}\rangle := \sum_{u\in S}|x+u\rangle \ ,$$

$$|S_{10}\rangle := \sum_{u\in S}(-1)^{\langle z,u\rangle}|u\rangle \ , \ |S_{11}\rangle := \sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u\rangle \ ,$$

   for the x, z derived from the decryption in step 3 of the minting protocol. This means that if we isolate and focus from hereon only on register BN (e.g. by tracing out other registers), the quantum state in BN is spanned by $\{|S_{b_1,b_2}\rangle\}_{b_1,b_2\in\{0,1\}}$. For at least one of these 4 basis states there is an amplitude $\geq \sqrt{\frac{1}{4}}$ for the state in BN, denote such state by $|S_{a_1,a_2}\rangle$.

   We define the second part of Exp as successful if the guess for the x coordinate is correct i.e. $\alpha = a_1$. This happens with probability $\geq \frac{1}{2}$, and overall the probability that both steps 1 and 2 of Exp are successful is at least $\frac{\varepsilon}{4}$.

3. Assume both steps 1 and 2 of Exp were successful. This means that there is $a_2 \in \{0,1\}$ such that after the guess for α made at step 2 of Exp, in register BN we have an amplitude $\geq \sqrt{\frac{1}{4}}$ on $|S_{\alpha,a_2}\rangle$.

   Consider the follow-up step done, where we add crt to BN conditioned on $\alpha = 1$. It is easy to verify that if $\alpha = 0$ (and indeed we guessed correctly) then at the end of the follow-up step, the state in BN has amplitude $\geq \sqrt{\frac{1}{4}}$ on the semi-decrypted subspace state $\sum_{u\in S}(-1)^{\langle b\cdot z,u\rangle}|u\rangle = |S_{0,b}\rangle$ (for some $b \in \{0,1\}$). In the case $\alpha = 1$ it follows that before the follow-up step, the quantum state in $\mathsf{BN}^{(1)}$ has amplitude $\geq \sqrt{\frac{1}{4}}$ on,

$$|S_{1,a_2}\rangle = \sum_{u\in S}(-1)^{\langle a_2\cdot z,u\rangle}\cdot|u+x\rangle \ .$$

Now, because $\mathsf{crt} \in \{0,1\}^\lambda$ passed classical certificate verification, it means that $\mathsf{O}_{S,x}(\mathsf{crt}) = 1$ and $\mathsf{O}_S(\mathsf{crt}) = 0$, which implies that there is some $u' \in S$ such that $\mathsf{crt} = u' + x$. Observe that after adding crt to the state $|S_{1,a_2}\rangle$ we get,

$$\sum_{u \in S} (-1)^{\langle a_2 \cdot z, u \rangle} \cdot |u + u'\rangle \ ,$$

and now we can use the fact that we are dealing with subspaces: because $u' \in S$ and the sum of u runs over the entire subspace S, the above state can be re-written as,

$$\sum_{u \in S} (-1)^{\langle a_2 \cdot z, u + u' \rangle} \cdot |u\rangle = (-1)^{\langle a_2 \cdot z, u + u' \rangle} \cdot \sum_{u \in S} (-1)^{\langle a_2 \cdot z, u \rangle} \cdot |u\rangle \ ,$$

where the above state has a global phase and is thus equivalent to $|S_{0,a_2}\rangle$.

4. It follows that conditioned on the success of steps 1 and 2 in Exp, after the follow-up procedure, if we isolate register BN, the quantum state in it has an amplitude $\geq \sqrt{\frac{1}{4}}$ for a semi-decrypted subspace state, that is, $|S_{0,b}\rangle$ for some $b \in \{0,1\}$. Now, $|S_{0,b}\rangle$ is in the span of the two vectors,

$$\{\sum_{u \in S} |u\rangle, \sum_{u \in S} (-1)^{\langle z, u \rangle} |u\rangle\} = \{|S_{00}\rangle, |S_{01}\rangle\} \ ,$$

and thus passes the projection test at step 3 of Exp with probability 1. Due to the fact that the amplitude of $|S_{0,b}\rangle$ is at least $\sqrt{\frac{1}{4}}$, the probability that the projection on BN succeeds, is at least $\frac{1}{4}$. Overall, the probability that the experiment Exp is successful is at least $\frac{\varepsilon}{16}$.

We can define the actions of the experiment Exp, except the final projection on $\{|S_{00}\rangle, |S_{01}\rangle\}$, as a quantum polynomial-time adversary $\mathcal{A}' = \{\mathcal{A}'_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ that does not need any extra information than what $\mathcal{A}$ receives. By the success probability of Exp, we know that $\mathcal{A}'$ contradicts Lemma 5.1. □

Security against sabotage. We next prove that the scheme has security against both, quantum verification sabotage and classical verification sabotage. That is, if a quantum banknote passes the public quantum verification of QV once, we are guaranteed two things:

- **Security against quantum sabotage:** The banknote will pass the next public quantum verification by QV with probability 1.
- **Security against classical sabotage:** A classical certificate of of destruction crt can be generated from the banknote by simply measuring it, such that crt passes the classical certificate verification CV with probability $1 - \mathrm{negl}(\lambda)$.

We start with proving security against quantum sabotage, and then show security against classical sabotage.

Proposition 5.2. *The scheme described in Protocol 1 is secure against quantum sabotage (as in Definition 3.4).*

Proof. We need to show that if a bank note BN passes the quantum verification procedure, then the probability that it passes another quantum verification procedure is negligibly close to 1, which in our case is going to be exactly 1. In Claim 4.2 we show that the quantum verification procedure acts as a projector on the space spanned by $B_S := \{|S_{00}\rangle, |S_{01}\rangle, |S_{10}\rangle, |S_{11}\rangle\}$. It is also straightforward to see that any state in B_S passes quantum verification with probability 1. This exactly guarantees security against quantum sabotage. □

Proposition 5.3. *The scheme described in Protocol 1 is secure against classical sabotage (as in Definition 3.4).*

Proof. We need to show that if a bank note BN passes quantum verification, then if we measure it in the computational standard basis, it generates a valid classical certificate crt with probability negligibly lose to 1. Consider an isolation of the quantum register BN after we performed quantum verification, and let $\varepsilon = \{\varepsilon_\lambda\}_{\lambda\in\mathbb{N}}$ be the average amplitude of the state in BN on the subspace $\{|S_{00}\rangle, |S_{01}\rangle\}$. Recall that the state in BN after successful quantum verification is in the space of $B_S := \{|S_{00}\rangle, |S_{01}\rangle, |S_{10}\rangle, |S_{11}\rangle\}$.

- If ε is negligible, this means the amplitude of $\{|S_{10}\rangle, |S_{11}\rangle\}$ in BN is negligibly close to 1. Now, due to the fact that a standard basis measurement on any of $|S_{10}\rangle$, $|S_{11}\rangle$ yields a vector in $S + x$ with probability 1, this means that with probability negligibly close to 1 the generated classical receipt $\mathsf{crt} \leftarrow Measure(\mathsf{BN})$ passes classical verification.
- If ε is noticeable, it means that projecting BN on the subspace $\{|S_{00}\rangle, |S_{01}\rangle\}$ succeeds with noticeable probability, in contradiction to Lemma 5.1.

□

5.2 Supplementary Claims

We prove supporting claims for the main Lemma 5.1.

Claim 5.1 (Synchronized Subspace Indistinguishability)**.** *Let iO an indistinguishability obfuscator for classical circuits, as in Definition 3.1. Denote the following,*

- $\mathbf{M} = \{\mathbf{M}_\lambda\}_{\lambda\in\mathbb{N}}$ *a binary matrix in $\{0,1\}^{k\times\lambda}$, for k such that there is some constant $\delta' \in (0,1)$, such that $k < \lambda - \lambda^{\delta'}$.*
- $x = \{x_\lambda\}_{\lambda\in\mathbb{N}}$ *a binary string of length λ, and denote by $(\mathbf{M}, x)$ the matrix in $\{0,1\}^{(k+1)\times\lambda}$ with its first row x and the rest of the matrix is $\mathbf{M}$.*
- $S = \{S_\lambda\}_{\lambda\in\mathbb{N}}$ *the row span of $\mathbf{M}$ and (S, x) the row span of $(\mathbf{M}, x)$.*
- *For any matrix M, let $C_M : \{0,1\}^\lambda \to \{0,1\}$ a classical circuit that checks membership in the row span of M, by Gaussian elimination.*

Assume injective one-way functions exist, then for any constant $\delta \in (0,1)$ such that $\delta < \delta'$ for all $d_0, d_1 \in \mathbb{N}$ such that $k < d_0 < d_1 < \lambda - \lambda^\delta$ we have the following indistinguishability.

$$\{\mathsf{O}_S, \mathsf{O}_{S,x} | \mathsf{O}_S \leftarrow \mathsf{iO}(C_\mathbf{M}), \mathsf{O}_{S,x} \leftarrow \mathsf{iO}(C_{\mathbf{M},x})\} \approx_c \{\mathsf{O}_{\tilde{T}}, \mathsf{O}_T | \mathsf{O}_{\tilde{T}} \leftarrow \mathsf{iO}(C_{\mathbf{M}_{\tilde{T}}}), \mathsf{O}_T \leftarrow \mathsf{iO}(C_{\mathbf{M}_T})\} \ ,$$

where $\tilde{T} \subseteq \{0,1\}^\lambda$ is a random d_0-dimensional superspace of S, and $T \subseteq \{0,1\}^\lambda$ is a random d_1-dimensional superspace of $(\tilde{T}, x)$.

Proof. We describe a sequence of indistinguishable hybrids.

Hyb_1 : The initial experiment as it is, i.e. we obfuscate the membership circuits $C_\mathbf{M}$, $C_{\mathbf{M},x}$.

Hyb_2 : Same as Hyb_1, only that instead of obfuscating the circuit $C_{\mathbf{M},x}$, we first obfuscate $\mathsf{O}_S \leftarrow \mathsf{iO}(C_\mathbf{M})$ as usual, and define the circuit $C'_{\mathbf{M},x}$ that for input u outputs 1 iff $(\mathsf{O}_S(u) = 1) \vee (\mathsf{O}_S(u + x) = 1)$. So, in Hyb_2 we send an obfuscation $\mathsf{O}'_{S,x} \leftarrow \mathsf{iO}(C'_{\mathbf{M},x})$ rather than an obfuscation of $C_{\mathbf{M},x}$.

Note that checking whether $(u \in S) \vee (u \in S + x)$ is the same as checking whether $u \in (S, x)$. So, by the correctness of the obfuscation $\mathsf{O}_S \leftarrow \mathsf{iO}(C_\mathbf{M})$, the functionality of $C'_{\mathbf{M},x}$ is the same as that of $C_{\mathbf{M},x}$. It follows that by the security of the indistinguishability obfuscation $\mathsf{O}'_{S,x} \leftarrow \mathsf{iO}(C'_{\mathbf{M},x})$, the distributions Hyb_1 and Hyb_2 are indistinguishable.

Hyb_3 : Same as Hyb_2, except that instead of obfuscating $\mathsf{O}_S \leftarrow \mathsf{iO}(C_{\mathbf{M}})$ (and use it in two places, once, send it out in the open, and second, we wire it inside the second obfuscated circuit $C'_{\mathbf{M}_{\tilde{T}},x}$) we obfuscate $\mathsf{O}_{\tilde{T}} \leftarrow \mathsf{iO}(C_{\mathbf{M}_{\tilde{T}}})$, where $\tilde{T} \subseteq \{0,1\}^\lambda$ is a random d_0-dimensional superspace of S (and $\mathbf{M}_{\tilde{T}}$ is some matrix that its row span is $\tilde{T}$). By the subspace hiding property of indistinguishability obfuscators (Lemma 3.1) and the fact that injective OWFs exist, the distributions Hyb_2 and Hyb_3 are indistinguishable.

Hyb_4 : In this hybrid we sample $\tilde{T} \subseteq \{0,1\}^\lambda$ a random d_0-dimensional superspace of S, and let $\mathbf{M}_{\tilde{T}} \in \{0,1\}^{d_0 \times \lambda}$ a matrix such that its row space is $\tilde{T}$. Let $\mathbf{M}_{\tilde{T},x} \in \{0,1\}^{(d_0+1)\times\lambda}$ the matrix with its first row x and the rest is $\mathbf{M}_{\tilde{T}}$. We send the obfuscations $\mathsf{O}_{\tilde{T}} \leftarrow \mathsf{iO}(C_{\mathbf{M}_{\tilde{T}}})$ and $\mathsf{O}_{\tilde{T},x} \leftarrow \mathsf{iO}(C_{\mathbf{M}_{\tilde{T},x}})$.

In both hybrids Hyb_3 and Hyb_4, the first obfuscation is $\mathsf{O}_{\tilde{T}} \leftarrow \mathsf{iO}(C_{\mathbf{M}_{\tilde{T}}})$. Regarding the second obfuscation, in Hyb_3 we send $\mathsf{O}'_{\tilde{T},x} \leftarrow \mathsf{iO}(C'_{\mathbf{M}_{\tilde{T}},x})$, and in Hyb_4 we send $\mathsf{O}_{\tilde{T},x} \leftarrow \mathsf{iO}(C_{\mathbf{M}_{\tilde{T},x}})$. Given $u \in \{0,1\}^\lambda$, checking whether $(\mathsf{O}_{\tilde{T}}(u) = 1) \vee (\mathsf{O}_{\tilde{T}}(u + x))$ and checking that $u \in (\tilde{T}, x)$ is equivalent. Thus, by the security of the indistinguishability obfuscation, the second obfuscations are indistinguishable between the two hybrids.

Hyb_5 : Same as Hyb_4, with the change that instead of obfuscating $\mathsf{O}_{\tilde{T},x} \leftarrow \mathsf{iO}(C_{\mathbf{M}_{\tilde{T}},x})$, we sample $T \subseteq \{0,1\}^\lambda$ a random d_1-dimensional superspace of $(\tilde{T}, x)$, and obfuscate $\mathsf{O}_T \leftarrow \mathsf{iO}(C_{\mathbf{M}_T})$. We send $\mathsf{O}_{\tilde{T}}, \mathsf{O}_T$.

The distributions are indistinguishable by the subspace hiding property of the indistinguishability obfuscation (Lemma 3.1) and that injective OWFs exist. □

Claim 5.2. *[Subspace-Hiding Encryption] Let* $(\mathsf{Enc}, \mathsf{Dec})$ *any encryption scheme such that no quantum polynomial-time adversary can distinguish encryptions of two different messages with advantage larger than* p_λ*, where* $\lambda \in \mathbb{N}$ *is the security parameter.*

Then, for any two subspaces $T_0, T_1 \subseteq \{0,1\}^\lambda$ *of dimension* $\frac{3\cdot\lambda}{4}$ *each, such that* $T_1^\perp \subseteq T_0$*, any quantum polynomial time algorithm* $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ *can win the following game with probability at most* $p_\lambda + 2^{-\frac{\lambda}{4}+1}$*:*

- *A random* $\frac{\lambda}{2}$*-dimensional subspace* S *subject to* $T_1^\perp \subseteq S \subseteq T_0$ *is sampled, described by a matrix* $\mathbf{M} \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$.
- *The matrix is encrypted* $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathbf{M})$*, and* ct *is sent to* $\mathcal{A}$.
- $\mathcal{A}$ *wins iff it finds a vector* $s \in \left(S \setminus T_1^\perp\right)$.

Proof. We prove the claim by a hybrid argument, and based on the security of the encryption scheme. Assume there is some quantum polynomial time adversary $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ that wins the game with probability $> p_\lambda + 2^{-\frac{\lambda}{4}+1}$.

Hyb_1 : The initial experiment as it is. The output of the experiment is defined to be the success bit of the adversary in the game.

Hyb_2 : The game as it is, but an encryption to the zero matrix $\mathbf{M}_0 \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$, $\mathsf{ct}_0 \leftarrow \mathsf{Enc}(\mathbf{M}_0)$ is sent instead of an encryption of the original matrix $\mathbf{M}$.

Consider the following two distributions.

- Distribution X: Sampling $\mathbf{M} \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$ randomly, and an encryption of it $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathbf{M})$. The output of the distribution is $(\mathbf{M}, \mathsf{ct})$.

- Distribution Y: Sampling $\mathbf{M} \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$ randomly, and an encryption of the zero matrix $\mathsf{ct}_0 \leftarrow \mathsf{Enc}(\mathbf{M}_0)$. The output of the distribution is $(\mathbf{M}, \mathsf{ct}_0)$.

In both X and Y the first step is to sample the matrix $\mathbf{M}$, but in X we send an encryption of it along with the matrix, and in Y we send an encryption of the zero matrix.

By the fact that for every two different messages $m \neq m'$, their encryptions cannot be distinguished with advantage better than p_λ, it follows in particular (by an averaging argument) that the above distributions cannot be distinguished with advantage better than p_λ.

A successful distinguisher between Hyb_1 and Hyb_2 (i.e. a quantum polynomial-time algorithm $\mathcal{A}'$ that succeeds in the two experiments with probabilities differing in advantage a) can be used to distinguish between X and Y with the same advantage a. This follows because when we get a sample from X or Y, we get to see the matrix $\mathbf{M}$, which means that when $\mathcal{A}'$ outputs a candidate s' to be in $(S \setminus T_1^\perp)$, we can check it against $\mathbf{M}$.

Now, if the success probability in Hyb_1 and Hyb_2 differ in more than p_λ, it means that distributions X and Y can be distinguished with advantage $> p_\lambda$, in contradiction. So, the success probability in Hyb_2 has to be $> p_\lambda + 2^{-\frac{\lambda}{4}+1} - p_\lambda = 2^{-\frac{\lambda}{4}+1}$.

Now, consider a different game where $\mathbf{M}$ is sampled but $\mathcal{A}$ gets no input at all, and just needs to guess some $s \in \left(S \setminus T_1^\perp\right)$. As $\mathcal{A}$ got no input, we can make an averaging argument on the output s of $\mathcal{A}$ that maximizes the probability to win the game. So, $\mathcal{A}$ always outputs some $s' \in \{0,1\}^\lambda$, independently of the sampled $\mathbf{M}$. Since $\mathbf{M}$ is a random $\frac{\lambda}{2}$-size basis for $T_1^\perp \subseteq S \subseteq T_0$, then for *any* string $s^* \in T_0$, the probability that $s^* \in \left(S \setminus T_1^\perp\right)$ is the same, which is,

$$\frac{|S \setminus T_1^\perp|}{|T_0 \setminus T_1^\perp|} = \frac{2^{\frac{\lambda}{2}} - 2^{\frac{\lambda}{4}}}{2^{\frac{3\cdot\lambda}{4}} - 2^{\frac{\lambda}{4}}} < \frac{2^{\frac{\lambda}{2}}}{2^{\frac{3\cdot\lambda}{4}-1}} = 2^{-\frac{\lambda}{4}+1} \ .$$

Finally, if $\mathcal{A}$ wins in the original game Hyb_1 with probability $> p_\lambda + 2^{-\frac{\lambda}{4}+1}$, then it wins in Hyb_2 w.p. $> 2^{-\frac{\lambda}{4}+1}$, and if this is the case, it can win in the above information theoretic game with the same probability $> 2^{-\frac{\lambda}{4}+1}$, in contradiction to the above bound $2^{-\frac{\lambda}{4}+1}$. □

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.

[BZ13] Dan Boneh and Mark Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. In *Annual cryptology conference*, pages 361–379. Springer, 2013.

[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.

Protocol 1

Key Generation:

- $\mathsf{Gen}(1^\lambda)$: The key generation algorithm is that of the signature scheme, that is, compute $(\mathsf{pk}_{\mathsf{Sig}}, \mathsf{sk}_{\mathsf{Sig}}) \leftarrow \mathsf{Sig.Gen}(1^\lambda)$, and set $\mathsf{pk} = \mathsf{pk}_{\mathsf{Sig}}$, $\mathsf{sk} = \mathsf{sk}_{\mathsf{Sig}}$.

Minting Protocol: The bank has $\mathsf{sk} = \mathsf{sk}_{\mathsf{Sig}}$ as private input, the joint input is the security parameter $\lambda \in \mathbb{N}$.

1. $\mathsf{BankMint}$ 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 $p_x \leftarrow \{0,1\}^{\frac{\lambda^2}{2}}$ to encrypt $\mathbf{M}_S^{(p_x)} = \mathsf{QHE.OTP}_{p_x}(\mathbf{M}_S)$, and then $\mathsf{fhek} \leftarrow \mathsf{QHE.Gen}(1^\lambda)$, $\mathsf{ct}_{p_x} \leftarrow \mathsf{QHE.Enc}_{\mathsf{fhek}}(p_x)$. $\mathsf{BankMint}$ sends the encryption $(\mathbf{M}_S^{(p_x)}, \mathsf{ct}_{p_x})$ to $\mathsf{RecMint}$.
2. Let C the quantum circuit that for an input matrix $\mathbf{M}$, outputs a uniform superposition of its row span. $\mathsf{RecMint}$ homomorphically evaluates C: $\left(|S\rangle^{(x,z)}, \mathsf{ct}_{x,z}\right) \leftarrow \mathsf{QHE.Eval}\left((\mathbf{M}_S^{(p_x)}, \mathsf{ct}_{p_x}), C\right)$. $\mathsf{RecMint}$ saves the quantum part $|S\rangle^{(x,z)}$ and sends the classical part $\mathsf{ct}_{x,z}$ to $\mathsf{BankMint}$.
3. $\mathsf{BankMint}$ decrypts $(x,z) = \mathsf{QHE.Dec}_{\mathsf{fhek}}(\mathsf{ct}_{x,z})$. If $x \in S$, abort the interaction. Let $\mathbf{M}_{S,x} \in \{0,1\}^{\left(\frac{\lambda}{2}+1\right) \times \lambda}$ the matrix generated by adding x as an initial row to $\mathbf{M}_S$. Let $\mathbf{M}_{S^\perp} \in \{0,1\}^{\frac{\lambda}{2} \times \lambda}$ a matrix with rows that are a basis for $S^\perp$, and let $\mathbf{M}_{S^\perp,z} \in \{0,1\}^{\left(\frac{\lambda}{2}+1\right) \times \lambda}$ the matrix generated by adding z as an initial row to $\mathbf{M}_{S^\perp}$. $\mathsf{BankMint}$ computes indistinguishability obfuscations $\mathsf{O}_{S,x} \leftarrow \mathsf{iO}(\mathbf{M}_{S,x})$, $\mathsf{O}_{S^\perp,z} \leftarrow \mathsf{iO}(\mathbf{M}_{S^\perp,z})$, $\mathsf{O}_S \leftarrow \mathsf{iO}(\mathbf{M}_S)$ and signs $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{sk}_{\mathsf{Sig}}}(\mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S)$ and sends $(\sigma, \mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S)$.

   The quantum part of the bank note is $|S\rangle^{(x,z)}$ which is stored in register BN, the classical part of the bank note is $(\sigma, \mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S)$.

Quantum Verification:

- $\mathsf{QV}\left(\mathsf{pk}_{\mathsf{Sig}}, (\sigma, \mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S, \mathsf{BN})\right)$: The verifier checks three things:
  - Checks the signature $\mathsf{Sig.Ver}_{\mathsf{pk}_{\mathsf{Sig}}}\left(\sigma, (\mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S)\right) = 1$.
  - Checks that $\mathsf{O}_{S,x}(\mathsf{BN}) = 1$ in superposition.
  - Executes Hadamard transform $H^{\otimes \lambda}$ on BN and then checks that $\mathsf{O}_{S^\perp,z}(\mathsf{BN}) = 1$ in superposition.

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

Classical Certificate Verification:

- In order to generate a classical certificate, the note holder measures BN in the standard basis $\mathsf{crt} \leftarrow Measure(\mathsf{BN})$.
- $\mathsf{CV}\left(\mathsf{pk}_{\mathsf{Sig}}, (\sigma, \mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S, \mathsf{crt})\right)$: The bank accepts the certificate iff $\mathsf{Sig.Ver}_{\mathsf{pk}_{\mathsf{Sig}}}\left(\sigma, (\mathsf{O}_{S,x}, \mathsf{O}_{S^\perp,z}, \mathsf{O}_S)\right) = 1$ and $(\mathsf{O}_{S,x}(\mathsf{crt}) = 1) \land (\mathsf{O}_S(\mathsf{crt}) = 0)$.

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