Semi-Quantum Tokenized Signatures*

Omri Shmueli†

Tel Aviv University

Abstract. Quantum tokenized signature schemes (Ben-David and Sattath, QCrypt 2017) allow a sender to generate and distribute quantum unclonable states which grant their holder a one-time permission to sign in the name of the sender. Such schemes are a strengthening of public-key quantum money schemes, as they imply public-key quantum money where some channels of communication in the system can be made classical.

An even stronger primitive is semi-quantum tokenized signatures, where the sender is classical and can delegate the generation of the token to a (possibly malicious) quantum receiver. Semi-quantum tokenized signature schemes imply a powerful version of public-key quantum money satisfying two key features:

- The bank is classical and the scheme can execute on a completely classical communication network. In addition, the bank is *stateless* and after the creation of a banknote, does not hold any information nor trapdoors except the balance of accounts in the system. Such quantum money scheme solves the main open problem presented by Radian and Sattath (AFT 2019).
- Furthermore, the classical-communication transactions between users in the system are *direct* and do not need to go through the bank. This enables the transactions to be both classical and private.

While fully-quantum tokenized signatures (where the sender is quantum and generates the token by itself) are known based on quantum-secure indistinguishability obfuscation and injective one-way functions, the semi-quantum version is not known under any computational assumption. In this work we construct a semi-quantum tokenized signature scheme based on quantum-secure indistinguishability obfuscation and the sub-exponential hardness of the Learning with Errors problem. In the process, we show new properties of quantum coset states and a new hardness result on indistinguishability obfuscation of classical subspace membership circuits.

1 Introduction

Quantum money schemes are one of the basis pillars in quantum cryptography, allowing a bank to distribute quantum unclonable states in a system of users, who can trade the states as currency. The gold standard of quantum money requires the scheme to be *public-key* [2], including two quantum algorithms, Bank and QV, with the following syntax: Bank samples a quantum token $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}) \leftarrow \mathsf{Bank}$,

*The full version of this work can be found at https://eprint.iacr.org/2022/228.

†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.

where $|\mathsf{qt}\rangle_{\mathsf{pk}}$ is a quantum state and pk is a classical public verification key. pk can be distributed in the user network and the quantum part $|\mathsf{qt}\rangle_{\mathsf{pk}}$ can be sent to some specific user. The copy of $|\mathsf{qt}\rangle_{\mathsf{pk}}$ can then be passed around between users in the system, and be publicly verified with QV using the key pk. The core security guarantee assures that tokens are unclonable by anyone but the bank, or even more tightly, no user can generate two states that both pass the quantum verification $\mathsf{QV}(\cdot, \mathsf{pk})$.

By combining intrinsic properties of quantum information with cryptographic techniques, public-key quantum money holds great promise for the future of information technology. Such quantum cryptographic schemes implement functionalities that are *known to be impossible* in a world where only classical computation exists and also create a basis of techniques towards even more advanced primitives, like quantum lightning [8] and quantum copy-protection of programs [1]. Notably, public-key quantum money gives a solution to the problem of privacy in a currency system, where we want a system that is both, secure (a banknote keeps its value and cannot be counterfeited) and private (transaction's information can be kept only to the two parties involved, in particular, the bank does not have to know).

Unfortunately, by the standard definition, to execute a quantum money scheme we need quantum computation to generate and verify tokens, and quantum communication to transfer tokens between devices[1]. Ideally, however, we would like to minimize the required model, and use quantum computation and only *classical* communication - more precisely, making the communication classical while keeping the key advantages of quantum money (e.g. privacy of transactions) is a central open problem in quantum cryptography. Besides the intriguing theoretical question and the fact that there is a fundamental difference between classical and quantum communication[2], practical differences include (1) the fact that a classical communication network can be based on *information broadcasting* (which uses information cloning to execute), which in particular enables communication between mobile devices, and (2) that transactions based on classical communication has the potential to provide *proof of payment*, as the clonable classical transcript can serve as a proof.

Looking more closely on the classical communication problem, there are three directions of communication in a token system: (1) from the bank to a user,

[1]Note that quantum teleportation is a known technique to transfer quantum information using classical communication channels. However, assuming no available quantum channel, physical contact is required to distribute the entangled EPR pairs that are used for teleporting the quantum data.

[2]e.g. classical information is more stable and classical communication is likely to be more efficient, as a consequence of the better algorithmic efficiency and lower rate of classical error correcting codes, compared to their quantum counterparts.

(2) from a user to another user, and (3) from a user to the bank. It is a known fact that the classical communication problem can be partially solved, by getting stronger no-cloning guarantees. Specifically, there are three known levels of no-cloning security for the quantum tokens. These levels enable increased classical communication, as we will later see.

1. **No Cloning:** The most basic security level of a quantum token is unclonability. No cloning says that a quantum polynomial-time malicious receiver Rec^* that obtains a single token $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ cannot output two quantum states $|\mathsf{qt}_1\rangle$, $|\mathsf{qt}_2\rangle$, such that both pass the public quantum verification $\mathsf{QV}(\cdot, \mathsf{pk})$.
2. **Classically Certifiable Destruction:** The next, stronger guarantee is classically certifiable destruction (CCD). In this version, along with Bank, QV, there are two additional algorithms; a quantum algorithm $\mathsf{GenCert}$ and a classical algorithm CV. While QV allows to publicly verify quantum tokens as before, $\mathsf{GenCert}$ allows to destroy the quantum token and output crt, a classical certificate of destruction for it. This certificate can later be verified by the classical verification algorithm CV using the public key pk.
   CCD security says that no adversary Rec^* can get a single token $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ and output both, a quantum token $|\mathsf{qt}\rangle'$ that passes the verification of $\mathsf{QV}(\cdot, \mathsf{pk})$ and crt a classical certificate for its destruction that passes the classical verification of $\mathsf{CV}(\cdot, \mathsf{pk})$. Note that this guarantee is at least as strong as the previous no-cloning, because as part of the correctness of schemes with CCD, for any quantum token $|\mathsf{qt}\rangle'$ that passes the verification $\mathsf{QV}(\cdot, \mathsf{pk})$, a valid classical certificate of destruction crt that passes $\mathsf{CV}(\cdot, \mathsf{pk})$ can be generated (thus two copies of the quantum token imply one quantum token and one classical certificate of destruction for it).
3. **Tokenized Signing:** The third and strongest known level of no-cloning security is tokenized signing. In such scheme like before we have Bank, QV, $\mathsf{GenCert}$, CV, except that now $\mathsf{GenCert}$ gets not only the quantum token $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$, but also a bit $b \in \{0, 1\}$. The bit b acts as a target for the destruction process. Specifically, given $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ and $b \in \{0, 1\}$, the algorithm generates $\mathsf{crt}_b \leftarrow \mathsf{GenCert}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}, b)$, a "certificate of destruction with respect to the bit b". The classical verification algorithm then gets, additionally to the classical certificate crt and the public key pk, a bit b, and verifies that indeed crt is a valid certificate for the bit b.
   The tokenized signatures security guarantee says that no Rec^* can get a single token $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ and generate two classical certificates crt_0, crt_1 that pass the classical verification with the two different bits, that is, crt_0 passes for $b = 0$ and crt_1 passes for $b = 1$. This guarantee is at least as strong as the previous CCD. To see this, assume there is an adversary Rec^* that outputs a quantum token $|\mathsf{qt}\rangle'$ that passes quantum verification and a

classical receipt crt that passes classical verification. crt passes classical verification which means it passes it for some bit $b \in \{0,1\}$ - we can find out what the bit b is by executing classical verification on crt with input target 0 and input target 1, and then use $|\mathsf{qt}\rangle'$ to generate a targeted classical certificate of destruction for $\neg b$. In this process we obtain crt_b, $\mathsf{crt}_{\neg b}$. The targeted destruction mechanism allows us to think of $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ as a one-time signature token to sign in the name of the bank on a single bit, and in particular, we can think of the certificate generation algorithm as a quantum signing algorithm $\mathsf{crt}_b \leftarrow \mathsf{Sign}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}, b)$, hence the name signature tokens.

User-to-bank classical communication from CCD tokens. When we move from standard unclonable tokens to CCD tokens, any user can effectively "send" tokens to the bank, using only classical communication: by destroying the token $\mathsf{crt} \leftarrow \mathsf{GenCert}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ and sending the classical crt to the bank, the user proves to the bank that it cannot spend the money of that token anymore in the network, and the bank can reimburse the balance of that user. Still, CCD tokens do not solve any of the other two directions of communication: from the bank to a user, and from one user to another user.

1.1 The Advantages of Quantum Signature Tokens

Having the strongest no-cloning guarantee, the power behind signature tokens emerges when the tokens are used in a sequence: We can take λ i.i.d. signature tokens $(\mathsf{pk}_1, |\mathsf{qt}\rangle_{\mathsf{pk}_1}), (\mathsf{pk}_2, |\mathsf{qt}\rangle_{\mathsf{pk}_2}), \cdots, (\mathsf{pk}_\lambda, |\mathsf{qt}\rangle_{\mathsf{pk}_\lambda})$ as a single "string signature token" unit that can sign on any length-λ string. Along with the sequence of tokens, the bank decides on a token value $x \in \mathbb{N} \cup \{0\}$ (in the context of quantum money, this is how much money the bank assigns to that token), samples a unique (with high probability) identifier which is a random serial number $s \leftarrow \{0,1\}^\lambda$, and a classical signature $\sigma := \sigma_{(\mathsf{pk}_1, \cdots, \mathsf{pk}_\lambda, x, s)}$ for the entire classical part of the token. The signature token is then

$$\mathsf{pk} = (\mathsf{pk}_1, \cdots, \mathsf{pk}_\lambda, x, s, \sigma)\,,\ |\mathsf{qt}\rangle_{\mathsf{pk}} = \left(|\mathsf{qt}\rangle_{\mathsf{pk}_1}, |\mathsf{qt}\rangle_{\mathsf{pk}_2}, \cdots, |\mathsf{qt}\rangle_{\mathsf{pk}_\lambda}\right)\ .$$

Note that σ is a signature for the entire sequence together, thus one cannot mix and match signatures of two different strings s_1, s_2 produced from two different tokens, in order to get a signature for a third string s_3. Tokens of value $x = 0$ can be regarded as "dummy tokens" - we next show how they can be used.

User-to-user classical communication from signature tokens. Like CCD tokens, string signature tokens enable the previous classical communication from user to bank (as they are only a strengthening of CCD tokens), but moreover,

they enable an additional direction of classical communication, from one user to another. More elaborately, one user Rec_1 holding a token $(\mathsf{pk}_1, |\mathsf{qt}\rangle_{\mathsf{pk}_1})$ of value x_1, can transfer the value x_1 to another user Rec_2 holding a token $(\mathsf{pk}_2, |\mathsf{qt}\rangle_{\mathsf{pk}_2})$ of value of 0, by using $|\mathsf{qt}\rangle_{\mathsf{pk}_1}$ to sign on s_2, the serial number of the token $(\mathsf{pk}_2, |\mathsf{qt}\rangle_{\mathsf{pk}_2})$. After the produced signature is sent to Rec_2, the token $(\mathsf{pk}_2, |\mathsf{qt}\rangle_{\mathsf{pk}_2})$ can be considered to have the value x_1.

Additionally to enabling user-to-user classical communication, two derived abilities of string signature tokens are as follows:

- **Online token destruction:** When the bank wants a certificate of destruction for any token, it samples a random string $d \leftarrow \{0,1\}^\lambda$ and asks the user to sign on d with the signature token.
- **Token value split:** To split the value x of the token $(\mathsf{pk}_1, |\mathsf{qt}\rangle_{\mathsf{pk}_1})$ between two tokens $(\mathsf{pk}_2, |\mathsf{qt}\rangle_{\mathsf{pk}_2})$, $(\mathsf{pk}_3, |\mathsf{qt}\rangle_{\mathsf{pk}_3})$ into $u_2, u_3 \in \mathbb{N} \cup \{0\}$ such that $u_2 + u_3 = x$ (i.e. the value of $(\mathsf{pk}_2, |\mathsf{qt}\rangle_{\mathsf{pk}_2})$ is added u_2 and the value of $(\mathsf{pk}_3, |\mathsf{qt}\rangle_{\mathsf{pk}_3})$ is added u_3), we can hash the serial numbers s_2, s_3 of the two target tokens along with the partition u_2, u_3 of x to a length-λ string, $H(s_2, s_3, u_2, u_3) = y$ for a collision resistant hash function $H : \{0,1\}^* \to \{0,1\}^\lambda$, and then use $(\mathsf{pk}_1, |\mathsf{qt}\rangle_{\mathsf{pk}_1})$ to sign on y. This effectively gives a classical proof for the new values of the tokens $(\mathsf{pk}_2, |\mathsf{qt}\rangle_{\mathsf{pk}_2})$, $(\mathsf{pk}_3, |\mathsf{qt}\rangle_{\mathsf{pk}_3})$.

More advantages of signature tokens for quantum money. Aside from direct classical transactions, we get additional unique characteristics to a public-key quantum money system that is based on string signature tokens: **(1) No token database:** When a user wants to return a token to the bank and get its bank account balance reimbursed (using only classical communication), the user and bank can execute the online destruction mechanism. In contrast, in a quantum money system based on CCD tokens, where the token return mechanism is the user simply generating a classical certificate of destruction by itself and sending it to the bank, the bank needs to maintain a database of all previously-destroyed tokens, so malicious users cannot illegally re-use the mechanism and send the same classical certificate of destruction multiple times, for the same token. **(2) Dynamic payment amounts:** The value split mechanism gives one the ability for granular payment amounts, where a user can dynamically choose the amount it wants to pay (unlike in the CCD-based scheme where the value x of a token is fixed during its creation by the bank). **(3) Provable payments:** When one user sends a direct payment to a second user, by signing on the serial number of a dummy token which the second users holds, this signature on the serial number is also a proof of payment, which we do not have in the CCD tokens setting (without going through the bank). **(4) Private classical payments:** While in a scheme based on tokenized signatures, classical user-to-user transactions

are direct and thus private, the bank can still obtain information when the user returns a banknote. The online destruction mechanism enables that when the user returns the signature for d using a token that was worth x, if it wishes to hide the token's information (i.e. all information of that token except its worth) and maintain privacy, it can encrypt the classical signature for d and send the encryption together with a zero-knowledge proof that the content of the encryption is a signature for d, and the token that signed on it has a value of x. This mechanism is still secure for the bank, as with high probability, it will never sample a repeating test string d.

1.2 Semi-Quantum Tokenized Signatures

We know how to construct public-key quantum money with signature tokens based on quantum-secure indistinguishability obfuscation and injective one-way functions, from a combination of the work of Ben-David and Sattath [3] with the work of Coladangelo, Liu, Liu, and Zhandry [4]. While such quantum money scheme can cover two out of three directions of communication classically (i.e. from users to the bank and from users to other users), the direction from the bank to users still needs to be quantum.

A strengthening of public-key quantum money is public-key *semi-quantum* money, where everything is the same as before (i.e. same syntax and hierarchy of no-cloning levels of the tokens), but the bank is a classical algorithm, which in particular makes the interaction from bank to users classical. More precisely, the generation of a token is by an interactive protocol between the classical bank Bank and a possibly malicious, quantum receiver Rec: $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}) \leftarrow \langle \mathsf{Bank}, \mathsf{Rec}\rangle_{(\mathsf{OUT}_{\mathsf{Bank}}, \mathsf{OUT}_{\mathsf{Rec}})}$, i.e. the output of the bank is pk (this is the public key which the bank can now distribute), and the output of the receiver is the quantum state $|\mathsf{qt}\rangle_{\mathsf{pk}}$. Similarly to before, no-cloning guarantees (i.e. standard no-cloning, CCD or tokenized signing) apply for the state $|\mathsf{qt}\rangle_{\mathsf{pk}}$, but crucially, these guarantees now need to hold even given the fact the actual generator of the state is a possibly malicious receiver Rec^*. Radian and Sattath [5] introduced the notion of semi-quantum money, and showed a construction of public-key semi-quantum money with CCD tokens, based on quantum lightning [8] - a primitive which to this day we do not know how to construct.

Shmueli [7] later constructs a public-key semi-quantum money scheme with CCD tokens, based on quantum-secure indistinguishability obfuscation and the sub-exponential quantum hardness of the Learning With Errors problem. This means that based on these computational assumptions, we know how to construct a public-key quantum money scheme that covers two directions of communication classically: from the bank to users (because the scheme is semi-quantum and a user can execute the receiver in the token generation protocol) and from

a user to the bank (because the tokens are CCD tokens, and as we have seen earlier, such tokens enable returning tokens to the bank by destroying them and sending the receipt to the bank)[3]. So, looking on what we saw until now,

- Public-key fully-quantum money with signature tokens is missing the classical direction from the bank to users, and,
- Public-key semi-quantum money with CCD tokens is missing the classical direction from one user to another.

It remains an open question to classically cover *all three directions of communication at once*. We don't know how to construct such primitive under any computational assumption.

A construction of public-key semi-quantum money with *signature tokens*, or in short, a semi-quantum tokenized signature scheme, solves the above problem, and more. Such scheme has a classical bank like the scheme from [7], but unlike the previous scheme, it also has the 4 fundamental advantages of signature tokens for quantum money (mentioned in Section 1.1). In particular, Radian and Sattath [5] leave two open problems in their work: One open problem of constructing a *memory-dependent* public-key semi-quantum money, and a stronger and the main open problem of constructing a *memoryless* public-key semi-quantum money (both notions are defined in their work). The public-key semi-quantum money with CCD tokens of Shmueli [7] solves the construction of a memory-dependent scheme, while constructing a semi-quantum tokenized signature scheme will resolve the main question of constructing a memoryless scheme.

Our focus in this work is to construct a semi-quantum tokenized signature scheme. On the technical side of things, such scheme will show for the first time that it is possible for a classical computer to securely delegate the generation of quantum states that maintain the tokenized signing property.

1.3 Results

We resolve the open question and construct a semi-quantum tokenized signature scheme, based on the existence of indistinguishability obfuscation (iO) for classical circuits secure against quantum polynomial-time attacks, and on that the Learning With Errors [6] problem has sub-exponential indistinguishability against quantum computers, that is, there exists some constant $\delta \in (0, 1)$ such

[3] A nice property of a semi-quantum CCD tokens scheme is *in-direct* classical-communication transactions from user to user: A user can return a token to the bank, and then the bank can classically send a newly-generated token with the same value to the recipient user of that transaction. Observe, however, that such in-direct transactions are always known by the bank and thus are not private, which is one of the fundamental problems that quantum money is intended to solve.

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[4].

Formally, we have the following main Theorem.

Theorem 1. *Assume that Decisional LWE has sub-exponential quantum indistinguishability and that indistinguishability obfuscation for classical circuits exists with security against quantum polynomial time distinguishers. Then, there is a semi-quantum tokenized signature scheme.*

The remaining of the paper is as follows. In Section 2 we explain the main ideas in our construction. The Preliminaries are omitted from this proceedings version and are given in the full version of this work. In Section 3 we present our construction of semi-quantum tokenized signatures with correctness proof and proof for security against sabotage. The full version of the paper also contains the security proof against signature counterfeiting.

2 Technical Overview

In this section we explain the main technical ideas in our construction and the structure of the overview is as follows. In Section 2.1 we review the previous works related to our goal of constructing semi-quantum tokenized signatures, and explain why a straightforward extension of these works does not work to obtain our goal. In Section 2.2 we describe our construction and the reasoning behind it, with no security proof. In Section 2.3 we explain how the security of the entire scheme is reduced to proving a new hardness property of indistinguishability obfuscation, which is captured by our main technical Lemma in the full version of this work. Finally, in Section 2.4 we explain the main steps of proving this Lemma.

2.1 Semi-quantum CCD Tokens and Fully-quantum Signature Tokens

Starting off based on previous work, there is a single protocol [7] where a classical Bank can delegate to a quantum Rec the generation of quantum unclonable tokens - this scheme lets the bank and receiver sample together by interaction $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}) \leftarrow \langle \mathsf{Bank}, \mathsf{Rec}\rangle_{(\mathsf{OUT}_{\mathsf{Bank}}, \mathsf{OUT}_{\mathsf{Rec}})}$ a token for the receiver (the public key is the output of the bank, which the bank can then share with anyone, in particular the receiver). More precisely, the tokens in the scheme are CCD tokens. As

[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.

mentioned in the introduction, the scheme also includes public quantum verification $\mathsf{QV}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}) \in \{0,1\}$, certificate generation $\mathsf{crt} \leftarrow \mathsf{GenCert}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$, and public classical verification $\mathsf{CV}(\mathsf{pk}, \mathsf{crt}) \in \{0,1\}$.

Our direction in this overview will be to upgrade the construction to be able to generate not only CCD, but signature tokens. This means to have a signing procedure $\sigma_b \leftarrow \mathsf{Sign}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}, b)$ instead of the certificate generation $\mathsf{crt} \leftarrow \mathsf{GenCert}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$, and the classical verification will become a classical signature verification $\mathsf{CV}(\mathsf{pk}, \sigma_b, b) \in \{0,1\}$. Looking at another previous work [3, 4] which uses a quantum bank but manages to build the stronger signature tokens, it makes sense to try and combine the techniques of the two works. These two works are even more so inviting to be fused, as it is the case that in both works, the tokens are *coset states* - states of the form $|S\rangle^{x,z} := \sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u\rangle$ for a subspace $S \subseteq \{0,1\}^\lambda$ and two strings $x, z \in \{0,1\}^\lambda$. Let us recall the high-order bits in the two works, and then examine their possible joining.

Recap: Coset states as fully-quantum signature tokens. The fully-quantum tokenized signature scheme of [3, 4] is as follows: The bank samples a random $\frac{\lambda}{2}$-dimensional subspace $S \subseteq \{0,1\}^\lambda$, random strings $x, z \in \{0,1\}^\lambda$ and generates $|\mathsf{qt}\rangle_{\mathsf{pk}} := |S\rangle^{x,z}$ i.e. $\sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u\rangle$. The public verification key of the state is $\mathsf{pk} = \left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}\right)$, for $\mathsf{O}_{S+x} \leftarrow \mathsf{iO}(C_{S+x}), \mathsf{O}_{S^\perp+z} \leftarrow \mathsf{iO}(C_{S^\perp+z})$, where iO is a quantum-secure indistinguishability obfuscator for classical circuits and $C_{S+x}, C_{S^\perp+z}$ are circuits that check membership in the corresponding cosets $S+x$, $S^\perp+z$. The entire token $(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ is sent to the receiver.

Public quantum verification QV of the scheme is the standard procedure to verify a coset state [2]: Given input a quantum λ-qubit register QT, (1) Check that the output qubit of $\mathsf{O}_{S+x}(\mathsf{QT})$ is 1, then (2) perform Quantum Fourier Transform (QFT) in base 2 i.e. $H^{\otimes\lambda}$ on QT, then (3) Check that the output qubit of $\mathsf{O}_{S^\perp+z}(\mathsf{QT})$ is 1. It is a known fact in the literature that a successful verification in such procedure projects the state to be exactly $|\mathsf{qt}\rangle_{\mathsf{pk}} = |S\rangle^{x,z}$. Finally, regarding the signing algorithm $\mathsf{Sign}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}}, b)$, to sign on $b=0$ just measure $|\mathsf{qt}\rangle_{\mathsf{pk}}$, and to sign on $b=1$ measure in the Hadamard basis i.e. perform $H^{\otimes\lambda}$ and then measure. Accordingly, a valid signature for $b=0$ is any string in $S+x$, which can be publicly verified using O_{S+x}, and a valid signature for $b=1$ is any string in $S^\perp+z$, which can be publicly verified using $\mathsf{O}_{S^\perp+z}$.

The main technical part of the works [3, 4] is to show that it is computationally impossible, given $\left(\left(\mathsf{O}_{S+x}, \mathsf{O}_{S^\perp+z}\right), |S\rangle^{x,z}\right)$, to output both $s \in (S+x)$ and $s^\perp \in \left(S^\perp+z\right)$.

Recap: Coset states as semi-quantum CCD tokens. Moving to the semi-quantum setting, the scheme of [7] includes a 3-message coset state generation protocol, as follows:

1. The classical Bank samples a random $\frac{\lambda}{2}$-dimensional subspace $S \subseteq \{0,1\}^\lambda$ (represented by a matrix $\mathbf{M}_S \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$), and sends to the receiver $(\mathbf{M}_S^x, \mathsf{ct}_x)$, an encryption of the matrix $\mathbf{M}_S$ under hybrid quantum fully-homomorphic encryption (QFHE)[5].
2. The quantum receiver Rec homomorphically evaluates the circuit C_{ssg}, which is a quantum circuit that gets as input the classical description of a subspace $S \subseteq \{0,1\}^\lambda$ e.g. by a matrix, and generates a uniform superposition over S. Thus, the receiver obtains a quantum, homomorphically evaluated ciphertext,
$$\left(|S\rangle^{x',z'}, \mathsf{ct}_{(x',z')}\right) \leftarrow \mathsf{QHE.Eval}\left((\mathbf{M}_S^x, \mathsf{ct}_x), C_{\mathsf{ssg}}\right) \quad ,$$
and sends to Bank the classical part $\mathsf{ct}_{(x',z')}$.
3. Bank decrypts $(x',z') = \mathsf{QHE.Dec}(\mathsf{ct}_{(x',z')})$ and sends obfuscations $\mathsf{O}_{S+x'} \leftarrow \mathsf{iO}(C_{S+x'})$, $\mathsf{O}_{S^\perp+z'} \leftarrow \mathsf{iO}(C_{S^\perp+z'})$ as the public verification key pk.

The coset state $|S\rangle^{x',z'}$ which the receiver holds is the quantum part $|\mathsf{qt}\rangle_{\mathsf{pk}}$ of the token. Accordingly, public quantum verification QV is identical to that of [3, 4], the certificate generation $\mathsf{crt} \leftarrow \mathsf{GenCert}(\mathsf{pk}, |\mathsf{qt}\rangle_{\mathsf{pk}})$ is simply a standard basis measurement and the classical certificate verification is just verifying $\mathsf{CV}(\mathsf{pk}, \mathsf{crt}) := \mathsf{O}_{S+x'}(\mathsf{crt})$.

In the security argument it is shown that it is computationally impossible to output both, the quantum state $|\mathsf{qt}\rangle'$ that passes the verification $\mathsf{QV}(\mathsf{pk}, \cdot)$ and a certificate of destruction for it i.e. any string $s \in (S + x')$. The work does not claim that the generated coset state maintains the tokenized signing property, in fact, it is not even defined what it means that a tokens signs on 0 or 1.

Attacking the combined scheme. As we said in the beginning of the overview, we should first try to combine the schemes. Since both schemes have the same token structure (a coset state) and public key (obfuscations of the membership functions for the primal and dual cosets), to combine the schemes, all we need to do is to take the token generation protocol of [7] and define a signature for $b = 0$ to be any $s \in (S + x')$ and a signature for $b = 1$ to be any $s^\perp \in (S^\perp + z')$. To argue that the combined scheme maintains the tokenized signing property, it is required to prove that for any quantum polynomial-time receiver Rec^* that interacts with the classical Bank during the token generation protocol, it is impossible to output $(s, s^\perp)$.

As it turns out, there is a simple way for an adversary to break the tokenized signing security of the combined protocol. More elaborately, consider the fol-

[5] A hybrid QFHE scheme is one where every encryption of a quantum state $|\psi\rangle$ is of the form $\left(|\psi\rangle^{x,z}, \mathsf{ct}_{(x,z)}\right)$, where $|\psi\rangle^{x,z}$ is a quantum OTP encryption of $|\psi\rangle$ with keys $x, z \in \{0,1\}^\lambda$, and $\mathsf{ct}_{(x,z)}$ is a classical FHE encryption of the keys.

lowing attacker Rec^* that interacts with Bank in the protocol of [7] (described in the previous paragraph):

1. Rec^* obtains $(\mathbf{M}_S^x, \mathsf{ct}_x)$, the first message from Bank.
2. Rec^* samples a random $r \in \{0,1\}^{\frac{\lambda}{2}}$ and homomorphically evaluates the following *classical* circuit $C_{r,1}$: The circuit $C_{r,1}$ takes as input the matrix $\mathbf{M}_S \in \{0,1\}^{\frac{\lambda}{2} \times \lambda}$ and outputs $s := r^T \cdot \mathbf{M}_S$, a vector in the row span. The receiver gets the ciphertext $(\mathsf{ct}_{x'}, s \oplus x')$.
3. Rec^* samples a random $r^\perp \in \{0,1\}^{\frac{\lambda}{2}}$ and homomorphically evaluates the following *classical* circuit $C_{r^\perp,2}$: The circuit $C_{r^\perp,2}$ takes as input the matrix $\mathbf{M}_S \in \{0,1\}^{\frac{\lambda}{2} \times \lambda}$, computes a basis for $S^\perp$ in the form of a matrix $\mathbf{M}_{S^\perp} \in \{0,1\}^{\frac{\lambda}{2} \times \lambda}$ and outputs $s^\perp := \left(r^\perp\right)^T \cdot \mathbf{M}_{S^\perp}$, a vector in the row span. The receiver gets the ciphertext $\left(\mathsf{ct}_{x''}, s^\perp \oplus x''\right)$.

Assume without the loss of generality that in the QFHE, the classical FHE scheme that encrypts the classical QOTP keys x, z, is a bit encryption scheme (this assumption is w.l.o.g. as we do have such QFHE schemes where the classical FHE is a bit-encryption scheme. In fact, this is true for most known constructions). This means in particular that the ciphertext $\mathsf{ct}_{x',z'}$ which the receiver sends in the second message of the protocol is comprised of two ciphertexts, $\mathsf{ct}_{x'}$, $\mathsf{ct}_{z'}$.

Going back to our attack, the malicious receiver Rec^* can send $(\mathsf{ct}_{x'}, \mathsf{ct}_{x''})$ as the second message in the protocol (which was originally $\mathsf{ct}_{x',z'}$) to Bank, which decrypts to get x', x'', and sends the obfuscations accordingly: $\mathsf{O}_{S+x'}$, $\mathsf{O}_{S^\perp+x''}$ in the third message of the protocol. Finally, note that the receiver still holds $(s \oplus x') \in (S + x')$ and thus a signature for $b = 0$, and also holds $(s^\perp \oplus x'') \in (S^\perp + x'')$ and thus a signature for $b = 1$.

2.2 Signing Coset States by Splitting

With accordance to the above attack, if we wish to stay with the classical generation protocol of [7], we need to move to a different signing procedure - this will be our first new technique. Formally, we would like to reduce the task of breaking the security of QFHE, to the task of breaking the security of the tokenized signature scheme. Note that S is a random subspace of dimension $\frac{\lambda}{2}$ and thus takes a tiny fraction of $\frac{2^{\frac{\lambda}{2}}}{2^\lambda} = 2^{-\frac{\lambda}{2}}$ inside the set of all length-λ strings $\{0,1\}^\lambda$. This means that by the security of the QFHE, it should be computationally hard to get $(\mathbf{M}_S^x, \mathsf{ct}_x)$ the classical QFHE encryption of a basis for S, and find a non-zero vector in S. Thus, what we aim for as a very first step is a *definition* of valid signatures for $b = 0$ and $b = 1$ such that given σ_0, σ_1, two signatures for 0 and 1, it is possible to efficiently derive a vector $s \in (S \setminus \{0\})$.

We suggest the following signature definitions for a bit $b \in \{0,1\}$: At the beginning of the protocol, additionally to choosing S at random, the bank randomly splits S (which has $\frac{\lambda}{2}$ dimensions) into S_0, a $\left(\frac{\lambda}{2}-1\right)$-dimensional subspace of S, and the coset $S_0 + w$, for $w \in (S \setminus S_0)$. Note that these two parts are exactly two disjoint halves of S. If we define a signature for b to be any string in $S_0 + b \cdot w + x'$, then one can verify that the sum of any pair of signatures $\sigma_0 \in (S + x')\,, \sigma_1 \in (S + w + x')$ is a non-zero vector inside S. The above only opens the way for the solution, as we did not yet solve the two main technical parts:

- **Signing:** Given the generated coset state $|S\rangle^{x',z'}$, how can the honest Rec always succeed in signing on b? Simply measuring $|S\rangle^{x',z'}$ will yield the wanted signature only with probability $1/2$.
- **Security:** Given our mechanism for signing (which we did not describe yet), how can we prove security for the new scheme? This part is presented in Sections 2.3 and 2.4 of the overview.

Projecting on half the coset with overwhelming probability. We put the security of the scheme aside for the rest of Section 2.2 and focus on proving correctness, that is, explaining how to sign. We show how to transform $|S\rangle^{x',z'}$ into $|S_0 + b \cdot w\rangle^{x',z'}$ given $b \in \{0,1\}$, which will suffice, as a signature can be obtained at that point with probability 1, by measurement. To enable the transformation, the first change in the protocol is that in the third and last message of the protocol, where the bank usually sends the public key $\mathsf{pk} := \left(\mathsf{O}_{S+x'}, \mathsf{O}_{S^\perp+z'}\right)$, it now sends an expanded key: $\mathsf{pk}' := \left(\mathsf{O}_{S_0+x'}, \mathsf{O}_{S_0+w+x'}, \mathsf{O}_{S^\perp+z'}\right)$.

Given the state $|S\rangle^{x',z'}$ and $\mathsf{pk}' := \left(\mathsf{O}_{S_0+x'}, \mathsf{O}_{S_0+w+x'}, \mathsf{O}_{S^\perp+z'}\right)$, we explain how to sign on $b = 0$ (the procedure for $b = 1$ is symmetric) by getting the state $|S_0\rangle^{x',z'}$. By measuring the output bit of $\mathsf{O}_{S_0+x'}(|S^{x',z'}\rangle)$, if we succeed (which happens with probability $1/2$) we are done, and if we fail we have $|S_0 + w\rangle^{x',z'}$. It will be enough for the procedure to make the correction and go from the faulty state $|S_0 + w\rangle^{x',z'}$ back to the original state $|S\rangle^{x',z'}$ - since the original state re-enables the experiment of obtaining the correct state $|S_0\rangle^{x',z'}$ with probability $1/2$, we can make λ consecutive iterations of trying to project $|S\rangle^{x',z'}$ to $|S_0\rangle^{x',z'}$ (and correct otherwise), and thus fail with an overall probability of $1 - 2^{-\lambda}$.

Correction of a faulty coset state. The correction procedure from $|S_0 + w\rangle^{x',z'}$ to $|S\rangle^{x',z'}$ is as follows: We start with performing QFT (i.e. $H^{\otimes\lambda}$) on $|S_0 + w\rangle^{x',z'}$ which gives us

$$\sum_{u \in S_0^\perp} (-1)^{\langle x'+w,u\rangle} |z' + u\rangle \ .$$

We can write the above state as

$$\sum_{\left(u\in S_0^{\perp}\right)\wedge(\langle u,w\rangle=0)}(-1)^{\langle x'+w,u\rangle}|z'+u\rangle+\sum_{\left(u\in S_0^{\perp}\right)\wedge(\langle u,w\rangle=1)}(-1)^{\langle x'+w,u\rangle}|z'+u\rangle$$

$$=\sum_{\left(u\in S_0^{\perp}\right)\wedge(\langle u,w\rangle=0)}(-1)^{\langle x',u\rangle}|z'+u\rangle-\sum_{\left(u\in S_0^{\perp}\right)\wedge(\langle u,w\rangle=1)}(-1)^{\langle x',u\rangle}|z'+u\rangle.$$

Notice that $u\in S^{\perp}$ if and only if $\left(u\in S_0^{\perp}\right)\wedge(\langle u,w\rangle=0)$, also, the set of vectors u' such that $\left(u'\in S_0^{\perp}\right)\wedge(\langle u',w\rangle=1)$ is exactly $S^{\perp}+v$, for any v such that $\left(v\in S_0^{\perp}\right)\wedge(\langle v,w\rangle=1)$. We thus write the above sum as

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

The left sum in the above state is exactly $|S^{\perp}\rangle^{z',x'}$, which means that if we project the above state with measuring the output bit of $\mathsf{O}_{S^{\perp}+z'}(\cdot)$ and get 1, we have $|S^{\perp}\rangle^{z',x'}$ and by executing QFT we go back to $|S\rangle^{x',z'}$, as required.

In case we get 0 then we have $\sum_{u\in S^{\perp}}(-1)^{\langle x',u+v\rangle}|z'+u+v\rangle$ and we go for the last part of the correction: We can clear the global phase,

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

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

and execute QFT to get

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

We can write the above state by splitting the sum to S_0 and S_0+w,

$$\sum_{u\in S_0}(-1)^{\langle z'+v,u\rangle}|x'+u\rangle+\sum_{u\in S_0}(-1)^{\langle z'+v,u+w\rangle}|x'+u+w\rangle\ ,$$

and the advantage in that is, because $\left(v\in S_0^{\perp}\right)\wedge(\langle v,w\rangle=1)$, the above state can be written as

$$\sum_{u\in S_0}(-1)^{\langle z',u\rangle}|x'+u\rangle-\sum_{u\in S_0}(-1)^{\langle z',u+w\rangle}|x'+u+w\rangle$$

$$=|S_0\rangle^{x',z'}-|S_0+w\rangle^{x',z'}\ .$$

Finally, although we can correct the above state to be $|S\rangle^{x',z'} := |S_0\rangle^{x',z'} + |S_0 + w\rangle^{x',z'}$ (by a phase flip conditioned on the acceptance bit of the circuit $\mathsf{O}_{S_0+w+x'}$), there is no need. This follows because the above state is again a state that enables projecting it on $|S_0\rangle^{x',z'}$ with success probability of $1/2$, and if we fail we get $-|S_0 + w\rangle^{x',z'} \equiv |S_0 + w\rangle^{x',z'}$, which were exactly the properties we needed from $|S\rangle^{x',z'}$.

2.3 Proving CCD Security Versus Proving Tokenized Signing Security

To quickly touch base on where we currently stand, our new generation protocol for signature tokens is the same as the CCD token generation from [7] (which is described in Section 2.1), with two differences:

- The last message from Bank to Rec in the new protocol is $\mathsf{pk}' := \left(\mathsf{O}_{S_0+x'}, \mathsf{O}_{S_0+w+x'}, \mathsf{O}_{S^\perp+z'}\right)$ rather than $\mathsf{pk} = \left(\mathsf{O}_{S+x'}, \mathsf{O}_{S^\perp+z'}\right)$ from the previous.
- Instead of the certificate generation $\mathsf{crt} \leftarrow \mathsf{GenCert}(\mathsf{pk}, |S\rangle^{x',z'})$ of the previous work which just makes a measurement to the coset state (and does not really use pk), we now have a bit-signing procedure $\sigma_b \leftarrow \mathsf{Sign}(\mathsf{pk}', |S\rangle^{x',z'}, b)$, described in Section 2.2.

Until now we did not cover any of the security aspects of our construction, only the correctness. This following part of the overview, which explains the security argument in high-level, is constructed as follows: We recall the security arguments from previous work [7] that are still relevant for our new construction, until we arrive at the key point of difference between the current work and the previous. Next, we explain why the previous techniques do not cover this difference. Finally, we explain how our main technical Lemma covers this gap and enables us to prove that the new scheme produces signature tokens. The overview for the proof of Lemma is presented in Section 2.4.

Previous techniques and our security argument outline. In our reduction setting, given a malicious Rec^* that breaks the security of the semi-quantum tokenized signature scheme, we construct an adversary $\mathcal{A}_{\mathbf{QHE}}$ against the QFHE scheme, in the following manner:

1. $\mathcal{A}_{\mathbf{QHE}}$ gets the ciphertext $(\mathbf{M}_S^x, \mathsf{ct}_x)$ as input (for a random S with dimension $\frac{\lambda}{2}$) and passes it directly to Rec^* as the first message of the bank in the protocol.
2. Rec^* returns ct^* as the second message in the protocol.
3. $\mathcal{A}_{\mathbf{QHE}}$ computes $(\mathsf{O}_1, \mathsf{O}_2, \mathsf{O}_3)$ as the third message in the protocol and sends to Rec^*.
4. Rec^* outputs two signatures σ_0, σ_1. These signatures are used by $\mathcal{A}_{\mathbf{QHE}}$, which outputs the sum $\sigma_0 + \sigma_1$ as an attempt for a non-zero vector in S.

The reason why this sum is indeed a non-zero vector in S, at least when the messages of the bank are honestly generated, was explained earlier, in the beginning of Section 2.2.

Note that the third message $(\mathsf{O}_1, \mathsf{O}_2, \mathsf{O}_3)$ of $\mathcal{A}_{\mathbf{QHE}}$ needs to be computationally indistinguishable from $\left(\mathsf{O}_{S_0+x'}, \mathsf{O}_{S_0+w+x'}, \mathsf{O}_{S^\perp+z'}\right)$, the third message in the original protocol. Crucially, in the original protocol, the secret key fhek of the QFHE is used to generate this third message. Specifically, the bank obtains (x', z') by decryption. Having fhek is clearly not possible for the QFHE adversary $\mathcal{A}_{\mathbf{QHE}}$, and the reduction needs to overcome this difficulty.

We prove the reduction by a hybrid argument, and use three previously known tools in the process.

Subspace-hiding obfuscation: We use the well-known subspace-hiding [8] property of indistinguishability obfuscation, which says that (as long as quantum-secure injective one-way functions exist) the obfuscation $\mathsf{O}_{S+x} \leftarrow \mathsf{iO}(C_{S+x})$ is indistinguishable from an obfuscation $\mathsf{O}_{T+x} \leftarrow \mathsf{iO}(C_{T+x})$, for a random superspace $S \subseteq T$ - as long as the dimension of T is not too largeFor any constant $\delta \in (0, 1]$, the indistinguishability holds for dimension bounded by $\lambda - \lambda^\delta$., even if S is known to the attempting distinguisher.

Sub-exponential security of QFHE: Another aid we use is the assumption that the QFHE has sub-exponential security[6], which in turn implies that it should not be possible to get a non-zero vector in S with probability greater than $\approx 2^{-\lambda^{\delta'}}$. Note that since we can pick δ the parameter indicating the dimension of the subspaces T_0, T_1 to be any constant, we can take it as a function of δ', in particular, $\delta := \frac{\delta'}{2}$. Such choice of parameters implies $2^{-\lambda^\delta} >> 2^{-\lambda^{\delta'}}$.

Blind sampling of obfuscations: As part of the security argument in [7] it is shown that given any fixed pair T_0, T_1 of subspaces with dimension $\lambda - \lambda^\delta$ each, even if we do not know x', z', we can successfully sample from a distribution indistinguishable from $(\mathsf{O}_{T_0+x'}, \mathsf{O}_{T_0+w+x'}, \mathsf{O}_{T_1+z'})$ with probability $\approx 2^{-\lambda^\delta}$.

Together, the above seemingly paves the way to finish the proof by a hybrid argument:

- Hyb_0 : In the first hybrid $\mathcal{A}_{\mathbf{QHE}}$ acts exactly like the bank and computes the third message $\left(\mathsf{O}_{S_0+x'}, \mathsf{O}_{S_0+w+x'}, \mathsf{O}_{S^\perp+z'}\right)$ using the secret QFHE key fhek. As we know, two valid signatures σ_0, σ_1 in this setting indeed imply that $\sigma_0 + \sigma_1$ is a non-zero vector in S.

[6]The sub-exponential security says that there exists some constant $\delta' \in (0, 1]$ such that it is impossible for any quantum polynomial-time attacker to distinguish encryptions of differing plaintexts with advantage greater than $2^{-\lambda^{\delta'}}$.

- Hyb_1 : In the next hybrid $\mathcal{A}_{\mathbf{QHE}}$ still holds fhek, but sends $(\mathsf{O}_{T_0+x'}, \mathsf{O}_{T_0+w+x'}, \mathsf{O}_{T_1+z'})$ instead. This is indistinguishable from the previous hybrid by the subspace hiding property of the iO. Recall the sub-exponential security of the QFHE where the exponent constant is $\delta' \in (0, 1]$. We take the dimension of the random superspaces $S_0 \subseteq T_0$, $S^{\perp} \subseteq T_1$ to be both $\lambda - \lambda^{\delta}$, for $\delta := \frac{\delta'}{2}$.
- Hyb_2 : In the next hybrid $\mathcal{A}_{\mathbf{QHE}}$ still holds fhek, but the subspaces T_0, T_1 are fixed by an averaging argument, to be the pair of subspaces that maximize the probability for a successful attack i.e. $\sigma_0 + \sigma_1 \in (S \setminus \{0\})$. Note that S is a random subspace of dimension $\frac{\lambda}{2}$ subjected to $S_0 \subseteq T_0$, $T_1^{\perp} \subseteq S$. By the sub-exponential security of the QFHE and by the fact that this restriction on S still leaves it enough entropy, it is still computationally impossible to find a non-zero vector in S with probability $>> 2^{-\lambda^{\delta'}}$.
- Hyb_3 : In this experiment $\mathcal{A}_{\mathbf{QHE}}$ does not hold fhek, and given the fixed subspaces T_0, T_1 samples from $(\mathsf{O}_{T_0+x'}, \mathsf{O}_{T_0+w+x'}, \mathsf{O}_{T_1+z'})$ and still succeeds with probability $\approx 2^{-\lambda^{\delta}}$, by blind sampling of the obfuscated circuits.

All hybrids from Hyb_0 to Hyb_2 are indistinguishable, thus in Hyb_2 we still have $\sigma_0 + \sigma_1 \in (S \setminus \{0\})$, but the secret QFHE key fhek is still needed. Hyb_3 then successfully samples from the same output distribution of Hyb_2, without holding fhek and with probability $\approx 2^{-\lambda^{\delta}} >> 2^{-\lambda^{\delta'}}$, which finishes the proof as with this same probability we get a non-zero vector in S, in contradiction to the sub-exponential security of the QFHE.

Key point of difference - quantumness in the reduction. We inserted one small, but fatal inaccuracy to the above hybrid argument: When we use subspace-hiding techniques to hide S, it becomes no longer correct that getting *any* vector $s \in (S \setminus \{0\})$ is sufficient to break the QFHE security. More precisely, in the last hybrid Hyb_2 and on, the subspaces T_0, T_1 are fixed and moreover, $T_1^{\perp} \subseteq S$. This makes getting $s \in (S \setminus \{0\})$ not only possible, but trivial: any $s \in (T_1^{\perp} \setminus \{0\})$ will do. In order to break the QFHE we will need $s \in (S \setminus T_1^{\perp})$.

To understand why needing $s \in (S \setminus T_1^{\perp})$ rather than only $s \in (S \setminus \{0\})$ tears apart the above security proof sketch for signature tokens, let us first understand why the above argument actually holds when we want to prove that the tokens in the scheme maintain the weaker, CCD security guarantee. In a nutshell, the key difference is that in the CCD security reduction we are able to use the *quantumness* of the output of the adversary Rec^*.

A successful adversary Rec^* against CCD security manages to output not only two classical strings as signatures, σ_0, σ_1, but one certificate $\mathsf{crt} \in (S + x')$ along with the quantum state $|S\rangle^{x',z'} := \sum_{u \in S} (-1)^{\langle z', u \rangle} |x' + u\rangle$. The use of such output in the reduction is by adding crt to the superposition $|S\rangle^{x',z'}$; this only cancels the x'-pad and gets us $|S\rangle^{0^{\lambda},z'}$. Now, the quantum state $|S\rangle^{0^{\lambda},z'}$

does not give us just an arbitrary non-zero vector in S, but measuring it gives us a *uniform sample* from S. In particular, it is easy to get $s \in (S \setminus T_1^\perp)$ from such measurement, because the fraction of $T_1^\perp$ in S is negligible, which means that with overwhelming probability, the random sample lands outside $T_1^\perp$.

Technically, the above hybrid argument fails to prove tokenized signing already in Hyb_1; Even though the hybrids $\mathsf{Hyb}_0, \mathsf{Hyb}_1$ are indeed indistinguishable, and even though in both of them we can know S_0, w, x', z' and check whether the output of Rec^* still maintains $\sigma_0 \in (S_0 + x')$, $\sigma_1 \in (S_0 + w + x')$, it can still be the case that $\sigma_0 + \sigma_1 \in T_1^\perp$. Then, this fact that $\sigma_0 + \sigma_1 \in T_1^\perp$ is dragged for the remaining hybrids, which invalidates the proof - the reduction does not find a vector in $(S \setminus T_1^\perp)$, and thus QFHE security is unbroken.

Avoiding the dual subspace to prove tokenized signing security. It seems that we need a property of the indistinguishability obfuscator that is of different nature from the subspace-hiding property. We want to claim that given an obfuscation O_{T_1} of a random superspace of $S^\perp$, it is computationally hard to find a vector in the dual subspace $T_1^\perp$. Note that such hardness property will finish our proof: We can use it after moving from the above Hyb_0 to Hyb_1, claiming that in Hyb_1, the adversary cannot find vectors in $T_1^\perp$. Finally, since the adversary does find vectors in S, we know that the vector in S we found $\sigma_0 + \sigma_1$ is in $\left(S \setminus T_1^\perp\right)$. This property can then be carried for the rest of the hybrid experiments, to break the security of the QFHE in the end.

Ideally we indeed would like to prove such strong hardness property, but we do not manage to do so, in fact, it isn't even true that it is always hard: If the dimension of $T_1^\perp$, the subspace of S is big enough (which means that the randomly sampled primal superspace T_1 is not that much bigger than $S^\perp$), just by outputting a vector in S, we must be able to land inside $T_1^\perp$ with good probability.

What we do manage to show in our main technical Lemma is a *dual subspace anti-concentration* property, that says that while it may be possible to hit the dual subspace $T_1^\perp$ after getting an obfuscation $\mathsf{O}_{T_1} \leftarrow \mathsf{iO}(C_{T_1})$ (for a random high-dimensional superspace of $S^\perp$), it is hard to concentrate there exclusively. In other words, such adversary will always have to make a *near miss*, i.e. even if it tries to avoid S, if it manages to hit $T_1^\perp$ with a noticeable probability, it has to accidentally hit the background subspace S sometimes, that is, also with a noticeable probability.

2.4 Hardness of Concentration in Dual of Obfuscated Subspace

The last part remaining is to state and prove our anti-concentration Lemma. The statement roughly says the following: Assume that quantum-secure injective one-way functions exist, that iO is a quantum-secure indistinguishability obfuscator

for classical circuits and that $S \subseteq \{0,1\}^\lambda$ is a subspace of dimension $\frac{\lambda}{2}$. Let $\delta \in (0,1]$ a constant, and for a quantum polynomial-time adversary $\mathcal{A}$, which is given $\mathsf{O}_T \leftarrow \mathsf{iO}(T)$ an obfuscation of a random $(\lambda - \lambda^\delta)$-dimensional subspace $T \subseteq \{0,1\}^\lambda$ subjected to $S \subseteq T$, denote by $s := \mathcal{A}(\mathsf{O}(T))$ the output of $\mathcal{A}$. Then, for any quantum polynomial-time $\mathcal{A}$, if the output vector s satisfies $s \in (T^\perp \setminus \{0\})$ i.e. inside the dual of T, with a noticeable probability ε_0, then, there is a noticeable probability ε_1 such that s also satisfies $s(S^\perp \setminus T^\perp)$ i.e. it is inside $S^\perp$ but outside $T^\perp$.

Our strategy for proving the lemma is showing a hardness reduction from obfuscated subspace distinguishing (breaking subspace hiding) to obfuscated subspace dual concentration (breaking the anti-concentration Lemma). More formally, assume there is a quantum polynomial-time adversary $\mathcal{A}$ that violates our anti-concentration lemma. This means that given $\mathsf{O}(T)$, the output $\mathcal{A}(\mathsf{O}_T)$ is in $T^\perp \setminus \{0\}$ with a noticeable probability, but is also concentrated there (with respect to $S^\perp$), that is, the output $A(\mathsf{O}_T)$ is in $S^\perp \setminus T^\perp$ with only a negligible probability. We aim to use this output pattern to construct a new adversary $\mathcal{A}_{\mathsf{sh}}$ that distinguishes between an obfuscation O_S of S and an obfuscation O_T of T, a random $(\lambda - \lambda^\delta)$-dimensional superspace of S. Our proof will consider two logical cases, which are split with accordance to the behavior of the output $\mathcal{A}(\mathsf{O}_S)$ of $\mathcal{A}$ on a random obfuscation of the membership circuit for the subspace S.

First Case: $\mathcal{A}(\mathsf{O}_S)$ is in some small subspace of $S^\perp$ with good probability. In the first case there exists some subspace $T_{\mathcal{A}}$ inside $S^\perp$ such that the output $\mathcal{A}(\mathsf{O}_S)$ of $\mathcal{A}$, for a random obfuscation $\mathsf{O}_S \leftarrow \mathsf{iO}(S)$, is in $T_{\mathcal{A}}$ with some non-negligible probability ε.

The main observation in the first case is given by three points, the combination of which gives us a way to use $\mathcal{A}$ in order to break subspace hiding:

1. The output of $\mathcal{A}$ for an obfuscation $\mathsf{O}_S \leftarrow \mathsf{iO}(S)$ of S is in some specific subspace $T_{\mathcal{A}}$ with a non-negligible probability, which means in particular that given $\mathcal{A}$, there exists a basis $B_{\mathcal{A}}$ for the subspace $T_{\mathcal{A}}$. This basis $B_{\mathcal{A}}$ can serve the new adversary $\mathcal{A}_{\mathsf{sh}}$ as non-uniform classical advice. This means that $\mathcal{A}_{\mathsf{sh}}$ can check membership in $T_{\mathcal{A}}$ efficiently by Gaussian elimination.
2. T is a random superspace of S with $\lambda - \lambda^\delta$ dimensions. This means that the dual $T^\perp$ of T is a relatively small subspace (of only λ^δ dimensions) and is random inside $S^\perp$. Combining this fact with the fact that $T_{\mathcal{A}}$ is a fixed and small subspace of $S^\perp$, the probability for $T^\perp$ and $T_{\mathcal{A}}$ to have a non-zero-vector intersection is exponentially small and in particular negligible.
3. The output $\mathcal{A}(\mathsf{O}_T)$ of $\mathcal{A}$ for an obfuscation of T is concentrated in $T^\perp$ with respect to $S^\perp$. This means that there is at most a negligible chance that $\mathcal{A}(\mathsf{O}_T)$ is in $S^\perp$ but outside $T^\perp$.

Recalling that $T_{\mathcal{A}}$ is inside $S^{\perp}$, it can be verified by the reader that the combination of 2 and 3 implies that the output $\mathcal{A}(\mathsf{O}_T)$ hits the subspace $T_{\mathcal{A}}$ with at most a negligible probability. On the other hand, the output $\mathcal{A}(\mathsf{O}_S)$ hits $T_{\mathcal{A}}$ with a non-negligible probability. Finally, an adversary $\mathcal{A}_{\mathsf{sh}}$ can get an obfuscation z which is either from $\mathsf{O}_S \leftarrow \mathsf{iO}(S)$ or from $\mathsf{O}_T \leftarrow \mathsf{iO}(T)$ (for an appropriately random T), execute $\mathcal{A}(z)$ and check if the result is in $T_{\mathcal{A}}$ - this gives an adversary that breaks subspace hiding with a non-negligible advantage in quantum polynomial time.

Second Case: $\mathcal{A}(\mathsf{O}_S)$ is scattered. In the second case we assume the negation of the first case, that is, there is no small subspace $T_{\mathcal{A}}$ of $S^{\perp}$ such that the output $\mathcal{A}(\mathsf{O}_S)$ is inside this subspace with a non-negligible probability. In first glance on the second case, it seems that the output of $\mathcal{A}$ is indistinguishable between the two input distributions O_S and O_T: while by the definition of the second case, the output $\mathcal{A}(\mathsf{O}_S)$ is scattered in $S^{\perp}$ (and also outside $S^{\perp}$), the output $\mathcal{A}(\mathsf{O}_T)$ is constrained to be in $T^{\perp}$ when inside $S^{\perp}$, but since the *subspace itself $T^{\perp}$ is scattered* in $S^{\perp}$, this also implies that $\mathcal{A}(\mathsf{O}_T)$ is somewhat of a random vector.

The key to solving the second case is first asking what happens over many samples, that is, let us assume for a moment that subspace hiding is extendable to many samples, and $\mathsf{O}_S^{(1)}, \mathsf{O}_S^{(2)}, \cdots, \mathsf{O}_S^{(\lambda)}$ is indistinguishable from $\mathsf{O}_T^{(1)}, \mathsf{O}_T^{(2)}, \cdots, \mathsf{O}_T^{(\lambda)}$. Because $\mathcal{A}(\mathsf{O}_S)$ is scattered, when looking on the outputs $\mathcal{A}(\mathsf{O}_S^{(1)}), \mathcal{A}(\mathsf{O}_S^{(2)}), \cdots, \mathcal{A}(\mathsf{O}_S^{(\lambda)})$, we get random vectors with no particular pattern (at the least, with respect to small subspaces of dimension $\leq \lambda^{\delta}$). However, when we look at the series of outputs $\mathcal{A}(\mathsf{O}_T^{(1)}), \mathcal{A}(\mathsf{O}_T^{(2)}), \cdots, \mathcal{A}(\mathsf{O}_T^{(\lambda)})$, since $\mathcal{A}(\mathsf{O}_T)$ is obligated to either be in $T^{\perp}$ or not be in the larger $S^{\perp}$ at all, and $T^{\perp}$ has a small dimension λ^{δ}, we get an interesting and efficiently recognizable pattern.

Second Case: Dimension Recognition rather than Subspace Recognition and Double Obfuscation. If we are building $\mathcal{A}_{\mathsf{sh}}$, an adversary against subspace hiding, T is unknown to $\mathcal{A}_{\mathsf{sh}}$. However, the adversary $\mathcal{A}_{\mathsf{sh}}$ can use $\mathcal{A}$ to recognize the *dimension* of the dual subspace $T^{\perp}$. Given the samples $\mathsf{O}_T^{(1)}, \mathsf{O}_T^{(2)}, \cdots, \mathsf{O}_T^{(\lambda)}$, we can check whether $\mathcal{A}(\mathsf{O}_T^{(i)})$ is in $S^{\perp}$ or not - if it is inside $S^{\perp}$, we put that vector v_i aside. At the end of this process, we get some set B^* of size $\leq \lambda$. Now, from the fact that $\mathcal{A}(\mathsf{O}_T)$ is concentrated in $T^{\perp}$, one can verify that the dimension of B^* is bounded by λ^{δ} with overwhelming probability. On the other hand, the same process of building B^*, when we are given samples $\mathsf{O}_S^{(1)}, \mathsf{O}_S^{(2)}, \cdots, \mathsf{O}_S^{(\lambda)}$ that are obfuscations of S rather than T, will yield a subspace B^* of dimension $> \lambda^{\delta}$. The last fact follows exactly because $\mathcal{A}(\mathsf{O}_S)$ is scattered, and does not hit a particular subspace $T_{\mathcal{A}}$ with a non-negligible probability (unlike the behavior of $\mathcal{A}(\mathsf{O}_S)$ in the first logical case). In conclusion, an adversary $\mathcal{A}_{\mathsf{sh}}$ that gets $z^{(1)}, z^{(2)}, \cdots, z^{(\lambda)}$ which are either i.i.d samples from $\mathsf{iO}(S)$ or

from $\mathsf{iO}(T)$, executes $\mathcal{A}(z^{(1)}), \cdots, \mathcal{A}(z^{(\lambda)})$, builds the basis B^* and checks if its dimension is bigger or bounded by λ^δ, can distinguish between the two extended distributions.

We saw that if the distributions $D_S := (\mathsf{O}_S^{(1)}, \mathsf{O}_S^{(2)}, \cdots, \mathsf{O}_S^{(\lambda)})$ and $D_T := (\mathsf{O}_T^{(1)}, \mathsf{O}_T^{(2)}, \cdots, \mathsf{O}_T^{(\lambda)}$ $\{T$ superspace of $S, \dim(T) = \lambda - \lambda^\delta\})$ are indistinguishable, we are done. However, a standard hybrid argument does not show this: a hybrid argument easily shows that subspace hiding implies that D_S is indistinguishable from $D'_T := (\mathsf{O}_{T_1}, \mathsf{O}_{T_2}, \cdots, \mathsf{O}_{T_\lambda})$, that is, a distribution where in every one of the λ obfuscations T_i is sampled independently of the previous samples and is only subjected to being a superspace of S with dimension $\lambda - \lambda^\delta$. In our target distribution D_T, the superspace T is sampled *once*, and all λ obfuscations are of the same subspace T.

Finally, we show how to use *double obfuscation* in order to complete our reduction. Specifically, what we show in the body of the paper (in the proof of the anti concentration Lemma) is that given a single sample $z_T \leftarrow \mathsf{iO}(T)$, by the standard security of indistinguishability obfuscation, λ second obfuscations $(\mathsf{O}^{(1)} \leftarrow \mathsf{iO}(z_T), \cdots, \mathsf{O}^{(\lambda)} \leftarrow \mathsf{iO}(z_T))$ of the already obfuscated once z_T are indistinguishable from D_T. It is also shown that a double obfuscation of S, that is, $\mathsf{O}_{\mathsf{O}_S} \leftarrow \mathsf{iO}(\mathsf{iO}(S))$ preserves all of the properties of a single obfuscation, in particular, the output $\mathcal{A}(\mathsf{O}_{\mathsf{O}_S})$ is also scattered like $\mathcal{A}(\mathsf{O}_S)$.

The above finishes our reduction: An adversary $\mathcal{A}_{\mathsf{sh}}$ that breaks subspace hiding will get z either from $\mathsf{iO}(S)$ or from $\mathsf{iO}\left(\{T \text{ superspace of } S, \dim(T) = \lambda - \lambda^\delta\}\right)$, and sample λ i.i.d. second obfuscations $(\mathsf{O}_z^{(1)}, \cdots, \mathsf{O}_z^{(\lambda)})$ of z. $\mathcal{A}_{\mathsf{sh}}$ will then execute $\mathcal{A}$ on each of the λ samples, get a vector v_i for each execution and put it aside if $v_i \in S^\perp$ - these vectors put aside are denoted with B^*. As we saw, if z is an obfuscation of S then the dimension of B^* is bounded by λ^δ with a negligible probability, and if z is an obfuscation of T then B^* has dimension $\leq \lambda^\delta$ with a non-negligible probability, which finishes our proof.

3 Semi-Quantum Tokenized Signatures Construction

In this section we present our construction of a semi-quantum tokenized signatures (SQTS) scheme, proof of correctness and proof of security against quantum and classical sabotage.

Ingredients and notation:

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

In Figure 1 we describe the token generation protocol and token quantum verification procedures. In Figure 2 we describe the quantum signing algorithm and the classical signature verification procedures.

Protocol 1

Token Generation Protocol: Sen is classical and Rec is quantum. The joint input is the security parameter $\lambda \in \mathbb{N}$.

1. Sen 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 $\mathsf{fhek} \leftarrow \mathsf{QHE.Gen}(1^\lambda, 1^{\ell(\lambda)})$ for some polynomial $\ell(\cdot)$, $\mathsf{ct}_{p_x} \leftarrow \mathsf{QHE.Enc}_{\mathsf{fhek}}(p_x)$. Sen sends the encryption $(\mathbf{M}_S^{(p_x)}, \mathsf{ct}_{p_x})$ to Rec.
2. Let C the quantum circuit that for an input matrix $\mathbf{M} \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$, outputs a uniform superposition of its row span. The receiver Rec homomorphically evaluates C: $(|S\rangle^{x,z}, \mathsf{ct}_{x,z}) \leftarrow \mathsf{QHE.Eval}\left((\mathbf{M}_S^{(p_x)}, \mathsf{ct}_{p_x}), C\right)$, saves the quantum part $|S\rangle^{x,z}$ and sends the classical part $\mathsf{ct}_{x,z}$ to Sen.
3. Sen decrypts $(x,z) = \mathsf{QHE.Dec}_{\mathsf{fhek}}(\mathsf{ct}_{x,z})$. If $x \in S$, the interaction is terminated. Let $\mathbf{M}_{S^\perp} \in \{0,1\}^{\frac{\lambda}{2}\times\lambda}$ a basis for $S^\perp$ (as a matrix), let w the first row in $\mathbf{M}_S$ and let $\mathbf{M}_{S_0} \in \{0,1\}^{(\frac{\lambda}{2}-1)\times\lambda}$ the rest of the matrix $\mathbf{M}_S$, without w.
   Sen computes indistinguishability obfuscations $\mathsf{O}_{S_0+x} \leftarrow \mathsf{iO}(\mathbf{M}_{S_0}, x)$, $\mathsf{O}_{S_0+w+x} \leftarrow \mathsf{iO}(\mathbf{M}_{S_0}, w+x)$, $\mathsf{O}_{S^\perp+z} \leftarrow \mathsf{iO}(\mathbf{M}_{S^\perp}, z)$, all with padding $\mathrm{poly}'(\lambda)$ for some polynomial poly'.
   The output of Sen is $\mathsf{pk} := \left(\mathsf{O}_{S_0+x}, \mathsf{O}_{S_0+w+x}, \mathsf{O}_{S^\perp+z}\right)$, the output of Rec is $|\mathsf{qt}\rangle_{\mathsf{pk}} := |S\rangle^{x,z}$.

Quantum Token Verification:

– $\mathsf{QV}\left(\left(\mathsf{O}_{S_0+x}, \mathsf{O}_{S_0+w+x}, \mathsf{O}_{S^\perp+z}\right), \mathsf{QT}\right)$: Given a public key and a λ-qubit quantum register QT, the verifier checks two things:
   - Checks that the output qubit of $(\mathsf{O}_{S_0+x} \vee \mathsf{O}_{S_0+w+x})(\mathsf{QT})$ is 1.
   - Executes Hadamard transform $H^{\otimes\lambda}$ on QT and then checks that the output qubit of $\mathsf{O}_{S^\perp+z}(\mathsf{QT})$ is 1.

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

Fig. 1: Token generation protocol between the classical sender and quantum receiver, and quantum token verification procedure of our semi-quantum tokenized signature scheme.

Protocol 2

Quantum Signing Algorithm:

- $\mathsf{Sign}\left(\left(\mathsf{O}_{S_0+x}, \mathsf{O}_{S_0+w+x}, \mathsf{O}_{S^\perp+z}\right), \mathsf{QT}, b\right)$: Given a public key, a λ-qubit quantum register QT and $b \in \{0,1\}$, the signing algorithm repeats the following procedure λ times and if the loop did not terminate in the middle, it outputs $\perp$.
  1. Measure the output qubit of $\mathsf{O}_{S_0+b\cdot w+x}(\mathsf{QT})$, let $m \in \{0,1\}$ the measurement result.
     (a) If $m = 1$, measure the register QT to get measurement σ_b, output σ_b and terminate.
     (b) If $m = 0$, execute $H^{\otimes\lambda}$ on QT, measure the output qubit of $\mathsf{O}_{S^\perp+z}(\mathsf{QT})$, and execute $H^{\otimes\lambda}$ on QT once again. Restart the loop.

Classical Signature Verification:

- $\mathsf{CV}\left(\left(\mathsf{O}_{S_0+x}, \mathsf{O}_{S_0+w+x}, \mathsf{O}_{S^\perp+z}\right), \sigma_b, b\right)$: To verify a classical signature candidate σ_b for the bit b, the verifier outputs the bit $\mathsf{O}_{S_0+b\cdot w+x}(\sigma_b)$.

Fig. 2: The quantum signature algorithm and the classical signature verification procedure of our semi-quantum tokenized signature scheme.

3.1 Correctness and Security Against Sabotage

We first prove that our scheme is correct, which includes two steps: (1) If the scheme's algorithms are ran honestly then the protocol ends successfully, with the output of the honest receiver having negligible trace distance to $|S\rangle^{x,z}$. (2) We recall that $|S\rangle^{x,z}$ passes the quantum verification with probability 1, which overall means that the probability to pass the quantum verification is $1-\mathrm{negl}(\lambda)$.

Claim. If the token generation protocol is executed honestly, the quantum token $|\mathsf{qt}\rangle_{\mathsf{pk}}$ has negligible trace distance from the state $|S\rangle^{x,z} := \sum_{u\in S}(-1)^{\langle z,u\rangle}|x+u\rangle$ (the output of the protocol is defined to be $\perp$ in case the honest sender aborted the interaction), where x, z are the values obtained by the decryption executed by the sender in step 3 of the protocol.

Proof. By the statistical correctness of the QFHE, at the end of step 2 of the generation protocol, the quantum state that the honest Rec holds in its quantum-evaluated register has negligible trace distance to $|S\rangle^{x,z}$, that is, this negligible distance holds with probability 1 over the first two messages of the protocol.

Now, we claim that the probability for such honest Rec to have $x \in S$ is negligible. So, assume towards contradiction it was noticeable. Because the probability for $x \in S$ is noticeable, it has to be the case that with a noticeable

probability, when we execute the honest protocol, at the end of step 2 the receiver holds a state with negligible trace distance to $|S\rangle^{x,z}$ for $x \in S$. Now, for any $x \in S$ it follows that $|S\rangle^{x,z} = |S\rangle^{0^\lambda,z}$. This means that by measuring the receiver's state we get a non-zero vector in S with overwhelming probability, and overall, with a noticeable probability we can get a non-zero vector in S without even knowing the QFHE secret key.

Getting a non-zero vector in S violates the security of the QFHE, due to the fact that S is chosen at random and it covers only a negligible fraction out of $\{0,1\}^\lambda$. So, the honest execution of the protocol terminates on with a negligible probability.

Overall, with probability $1 - \mathrm{negl}(\lambda)$, we have $x \notin S$, the protocol ends successfully and the receiver holds a quantum state with negligible trace distance to $|S\rangle^{x,z}$.

We explain how Claim 3.1 implies the statistical correctness of our scheme.

Proposition 1. *The scheme presented in Protocol 1 has statistical correctness.*

Proof. In Claim 3.1 we saw that with probability $1-\mathrm{negl}(\lambda)$, the honest receiver Rec holds a quantum state with negligible trace distance to $|S\rangle^{x,z}$.

Finally, our public quantum verification QV is the standard QFT-based verification procedure of a coset state, and a well-known fact in the literature that a successful verification of such procedure is a projection of the verified state onto the subspace spanned only by the coset state [2, 3]. Because the trace distance of $|\mathsf{qt}\rangle_{\mathsf{pk}}$ from $|S\rangle^{x,z}$ is negligible, the probability for the state to be verified is overwhelming.

Overall, with probability $1 - \mathrm{negl}(\lambda)$ over the execution of the honest protocol, the receiver's quantum state passes the quantum verification $\mathsf{QV}(\mathsf{pk},\cdot)$.

Security against quantum sabotage. From the fact that the quantum verification $\mathsf{QV}(\mathsf{pk},\cdot)$ is a projector on the coset state, it follows that after a single successful quantum verification, $|\mathsf{qt}\rangle_{\mathsf{pk}}$ is now $|S\rangle^{x,z}$, which passes the next quantum verification with probability 1.

It remains to prove the security of the scheme against classical sabotage.

Proposition 2. *The scheme presented in Protocol 1 has security against classical sabotage.*

Proof. The starting point of the algorithm is the state after passing successfully the verification $\mathsf{QV}(\mathsf{pk},\cdot)$, which, as we stated above, means the state is exactly $|S\rangle^{x,z}$. After the first step of an iteration, if $m = 1$ we are done as we have $|S_0 + b \cdot w\rangle^{x,z}$ after the measurement, which means that by measuring we get $\sigma_b \in (S_0 + b \cdot w)$ with probability 1. If $m = 0$ we now have $|S_0 + (\neg b) \cdot w\rangle^{x,z}$.

Regarding the second step 1b, denote by $m' \in \{0,1\}$ the measured output bit of $\mathsf{O}_{S^\perp+z}(\mathsf{QT})$, that is, in step 1b of the signing procedure we execute QFT on QT, then measure the output qubit of $\mathsf{O}_{S^\perp+z}(\mathsf{QT})$ (we denoted by m' the outcome of this 1-qubit measurement) and then execute QFT on QT again.

One can verify that if $m' = 1$ then we have $|S^\perp\rangle^{z,x}$ before the second QFT, and thus back to $|S\rangle^{x,z}$ after the second QFT. On the other hand, if $m' = 0$, after the second QFT we have $|S_0\rangle^{x,z} - |S_0 + w\rangle^{x,z}$.

In any case, regardless of the value m', at the end of step 1b of the signing procedure, the state (which is either $|S\rangle^{x,z}$ or $|S_0\rangle^{x,z} - |S_0 + w\rangle^{x,z}$) maintains the property that after measuring the the output bit of $\mathsf{O}_{S_0+b\cdot w}(\mathsf{QT})$ (which will come up in upcoming step 1 of the next iteration) will project the state to be the correct $|S_0 + b \cdot w\rangle^{x,z}$ with probability $1/2$ and with the remaining probability $1/2$ it will be projected to $|S_0 + (\neg b) \cdot w\rangle^{x,z}$.

We deduce that at the beginning of each of the λ iterations we make, when we start with step 1, before the step is executed, we have a state that is projected to $|S_0 + b \cdot w\rangle^{x,z}$ with probability $1/2$ and to $|S_0 + (\neg b) \cdot w\rangle^{x,z}$ with probability $1/2$. The entire process will thus fail only if we fail consecutively λ times, where each experiment is independent from the rest and succeeds with probability $1/2$. Overall, this implies a failure probability of $1 - 2^{-\lambda}$.

Acknowledgements

We are grateful to Tamer Mour, for helpful discussions during the writing of this work.

References

1. Scott Aaronson. Quantum copy-protection and quantum money. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 229–242. IEEE, 2009.
2. 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.
3. Shalev Ben-David and Or Sattath. Quantum tokens for digital signatures. *arXiv preprint arXiv:1609.09047*, 2016.
4. 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.
5. Roy Radian. Semi-quantum money. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, pages 132–146, 2019.
6. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009.
7. Omri Shmueli. Public-key quantum money with a classical bank. *Cryptology ePrint Archive*, 2021.
8. 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.