Article

Practically Feasible Robust Quantum Money with Classical Verification

Niraj Kumar [ID]

School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, UK; nkumar@exseed.ed.ac.uk

Featured Application: Information-theoretic secure private quantum money.

Abstract: We introduce a private quantum money scheme with the note verification procedure based on sampling matching, a problem in a one-way communication complexity model. Our scheme involves a bank who produces and distributes quantum notes, noteholders who are untrusted, and trusted local verifiers of the bank to whom the holders send their notes in order to carry out transactions. The key aspects of our money scheme include: note verification procedure requiring a single round classical interaction between the local verifier and bank; fixed verification circuit that uses only passive linear optical components; re-usability of each note in our scheme which grows linearly with the size of note; and an unconditional security against any adversary trying to forge the banknote while tolerating the noise of up to 21.4%. We further describe a practical implementation technique of our money scheme using weak coherent states of light and the verification circuit involving a single 50/50 beam splitter and two single-photon threshold detectors. Previous best-known matching based money scheme proposal involves a verification circuit where the number of optical components increase proportional to the increase in desired noise tolerance (robustness). In contrast, we achieve any desired noise tolerance (up to a maximal threshold value) with only a fixed number of optical components. This considerable reduction of components in our scheme enables us to reach the robustness values that is not feasible for any existing money scheme with the current technology.

Keywords: quantum money; private cryptography; sampling matching; unconditional security

---

1. Introduction

In the 1980s, Wiesner [1] proposed the idea of quantum money to create unforgeable banknotes with quantum states. In his scheme, the banknotes are several BB84 states prepared by an honest authority, a bank, which then distributes them to the untrusted holders. When the holders need to carry out a transaction with their note, they send the entire note to the bank for verification, who declares its validity. The unforgeability property of the note in Wiesner's scheme relies on the no-cloning property of quantum mechanics which prevents the holder from creating multiple copies of the notes with just a single copy [2]. This idea was incidentally also among the first quantum cryptographic primitives to be introduced. Subsequently, other cryptographic tasks based on quantum mechanics have been proposed such as quantum key distribution, digital signatures, coin flipping, secure multi-party computation, etc [3–8].

This Wiesner money scheme, as analyzed independently by Lutomirski et al. [9] and Brodutch et al. [10], soon ran into security problems. The first issue was (a) verification of the note required a quantum communication channel between the holder and the bank. As pointed out by Gavinsky [11], an adversary can interfere in the channel and possibly modify or destroy the note; (b) the scheme is insecure against several previously un-analyzed attacks, the adaptive attacks [10] where an adversary can substantially increase the note forging probability by communicating with the bank in a few "auxiliary" number of rounds.

These two drawbacks were first addressed by Gavinsky [11] in his proposal of a private quantum money scheme based on quantum retrieval games (QRG). Informally, these games consist of a set of challenges where answering a single challenge from the set is easy, however answering multiple challenges simultaneously is hard i.e., cannot be answered with a unit probability. Gavinsky's scheme involves an honest bank that prepares and distributes notes which are quantum states prepared some random basis, untrusted noteholders and trusted local verifiers of the bank who run a test on holder's note to check for its validity. The security of this scheme relies on the fact that an honest user must always pass the verifier test who picks a single challenge from the set at random. A dishonest user who wants to forge the note has to simultaneously succeed in answering any two challenges picked from the set at random by two independent verifiers. Thus the hardness of forging the banknote links to the hardness in answering any two different challenges picked at random from the set. The verification of the note in Gavinsky's scheme requires two rounds of classical communication between the bank and the verifier and is proven to be secure against any type of adaptive adversary attacks. This scheme, however, is not realistic since it only works when no experimental noise is taken into consideration. Also, since the verification procedure requires two rounds of communication, this forces the bank to have a temporary "active" classical memory during the verification phase which would limit the number of independent note verifications that the bank can perform at any given instant.

Further independent works on similar lines by Georgiou et al. [12] and Amiri et al. [13] have reduced the classical communication required for note verification to a single round. The scheme of [12] is based on one-out-of-two QRG i.e., the holder can deterministically answer any one of the two challenges, however, it is impossible for him to answer both the challenges simultaneously with a unit probability. This scheme tolerates the noise of up to 12.5%. The other scheme of [13] is based on hidden matching quantum retrieval games, HM-QRG [14,15], and exhibits the noise tolerance of up to 23.3%. They further conjecture that maximal noise tolerance for money schemes based on Matching QRGs can reach up to 25%. Here the noise tolerance, a measure of the robustness of the scheme, is defined as the maximum theoretical probability that an honest verifier returns an incorrect outcome. The higher the noise tolerance, the more robust is the money-scheme against errors incurred on the honest holder's note due to experimental imperfections. As long as the errors on the honest holder's note is within the noise tolerance, the money scheme would demonstrate an information-theoretic security against a forger trying to forge the banknote. In parallel, several other theoretical proposals of quantum money schemes have been proposed, both as private and public money schemes [16–19]. Besides, there has been a recent proposal on semi-quantum money [20] where the bank only has classical resources and delegates a classical message to the user who prepares the quantum note. The security of this scheme is, however, computational and relies on the security provided by learning with errors (LWE), a problem believed to be post-quantum secure.

To date, there have been two proof-of-principle experimental demonstrations for private quantum money based on one round of classical verification with the bank. The first by Bozzio et al. [21] is based on the theoretical scheme of [12]. Their encoding of quantum money is in polarized weak coherent states and achieves the honest noteholder error rate slightly below $\beta = 4\%$ which is well under the maximum noise tolerance of 12.5%. The other demonstration is by Guan et al. [22] which is based on the theoretical scheme of [13] has an encoding based on phase parity of corresponding pairs of weak coherent states. Their implementation achieves a measured honest holder error rate of $\beta = 3\%$, while the theoretical noise tolerance of the scheme is 16.6%, thus making it secure. Inspite of the maximal noise tolerance of 23.3% proposed in the theoretical scheme, the experimental scheme was limited to 16.6%. The reason for this being that for the protocols based on matching schemes, the tolerance against the noise increases with the input size of the note. Thus the money scheme becomes more robust against experimental imperfections and forging by going to higher input sized banknotes. For the schemes based on hidden matching, the verification protocol involves a complex circuit with the number of optical elements (active switches, delays, beam splitter) increasing at least logarithmically with the input size. This gets increasingly difficult to implement the circuit for large

input sizes. Hence the only implementation based on hidden matching has been shown for input size $n = 4$ which leads to a noise threshold of 16.6%. This is the primary motivation of our theoretical work which simplifies the verification circuit to be able to experimentally achieve much higher noise tolerance than any other current private money scheme.

In this work, we introduce a private quantum money scheme using single-photon quantum states and the verification protocol based on sampling matching (SM) scheme. This scheme was proposed by Kumar et al. [23] as a problem in a one-way randomized communication complexity model. The authors showed that solving this problem using quantum resources exhibits an exponential reduction in communication compared to classical resources. They further experimentally demonstrated the quantum protocol using weak coherent states and linear optics operations. The simplicity of the problem allowed for the construction of such a quantum protocol consisting of only $\mathcal{O}(1)$ 50/50 beam splitters and two threshold detectors independent of the problem size. This led to the first experimental realization of quantum advantage in one-way communication complexity. This problem is described in more detail in the Section 2.2.

We propose two distinct encoding of private quantum money schemes. The first proposal uses single-photon states to create a quantum note. The single-photon encoding is the realization of a qubit/qudit with linear optics. Thus defining our money scheme in this picture makes it more translatable to other qubit encoding pictures for experimental demonstrations. Our money scheme, while achieving a noise tolerance of 21.4%, offers much simplicity in the implementation of the verification protocol compared to the existing hidden matching-based protocols [13]. In our verification scheme with single-photon states, even though the number of linear optical components grows linearly with the input size, the optical components required are just passive 50/50 beam splitters with no need for active switch components. This allows our circuit to be fixed for a given input size of the note, in contrast to the scheme of [13] which requires a programmable verification circuit, thus necessitating the need for having active components.

In the second part, we describe a practical implementation technique of our money scheme using weak coherent states and the verification circuit consisting of a single 50/50 beam splitter and two single-photon threshold detectors. Here we show that it is experimentally possible to achieve a noise tolerance as high as 21.4%, something not possible for any other scheme with the current technology.

The paper is organized as follows. In Section 2.1 we define the private quantum money including the notions of correctness and unforgeability. In Section 2.2 we introduce the tools required to construct our money scheme. This includes defining a modified version of the SM problem and SM scheme with single-photon states. In Section 3.1 we formally introduce our quantum money scheme using SM verification. Sections 3.2 and 3.3 analyses the security of our money scheme and prove that it exhibits an information-theoretic security. Finally, in Section 3.4 we describe the practical implementation technique of our quantum money scheme using coherent states and threshold detectors.

2. Materials and Methods

This section focuses on the necessary definitions and tools required for the construction of our quantum money scheme.

2.1. Definitions for Private Quantum Money

A private quantum money scheme involves an algorithm used by a trusted entity, a bank, to produce multiple notes, and a protocol that is run between a holder H of the note and the bank to verify the authenticity of the note. The requirement for the verification protocol to be secure is that it must be impossible for an adversary noteholder to create more notes than what it received from the bank.

Definition 1. *Private quantum money. A quantum money scheme with classical verification consists of an algorithm by the bank, and a protocol, verification, such that,*

1. *The bank algorithm produces a quantum note* $\$ = (\rho, s.n.)$ *where* ρ *is a quantum state of the note and s.n. is the classical serial number of the note.*
2. *Verification is a protocol with classical communication that is run on the note* $\$$*, between the noteholder H who claims to possess the note* $\$$ *and the bank. The output of the protocol is a bit declared by the bank to denote whether the note is valid or not. We denote this final bit as* $Ver_H^B(\$)$ *which is 1 when the bank validates the note and 0 otherwise.*

For this scheme to be secure, it must satisfy two important properties,

- Correctness: The scheme is ϵ correct if, for every honest holder H, it holds that

$$\mathbb{P}[Ver_H^B(\$) = 1] \geqslant 1 - \epsilon \tag{1}$$

- Unforgeability: the scheme is ϵ unforgeable if for any quantum adversary who possesses m notes, has interacted a finitely bounded number of times with the bank and has managed to produce m' notes $\$_1, \$_2, \cdots, \$_{m'}$, it holds that,

$$\mathbb{P}\left[\left(\bigwedge_{i \in [m']} Ver_H^B(\$_i) = 1\right) \wedge (m' > m)\right] \leqslant \epsilon, \tag{2}$$

where H is any honest noteholder. The probability in the ϵ-unforgeability property is taken over all possible strategies of an unbounded adversary.

The correctness condition ensures that all the honest noteholders get their note verified with an exponentially close to 1 probability (by setting ϵ exponentially close to 0). While the unforgeability condition ensures that an adversary trying to create more notes than what she had originally taken from the bank, would fail with an exponentially close to 1 probability in being able to verify all the notes. Our definition includes the possibility of adaptive attacks by the adversary since we allow the interaction with the bank a finite number of times during the verification protocol.

However, proving the security of such a general money-scheme can be a cumbersome task. To mitigate this, Aaronson and Christiano [17] introduced the concept of a smaller public money scheme (mini-scheme) and showed that it is sufficient to prove the security of this smaller version to guarantee security of the full scheme. Subsequently Ben-David and Sattath [24] showed similar results for private money schemes using their construction of a private tokenized signature scheme. Such a signature scheme is used to produce signed documents that are publicly verifiable, unfeasible to forge by a third party, and has the property that the signing authority can produce and distribute one-use quantum signing tokens that allow the holder to sign only 'one' document of her choice. The cryptographic assumption for the existence of such a private tokenized signature scheme is the existence of a collision-resistant hash function which is secure against quantum adversaries. There are various candidates for collision-resistant hash functions that are believed to be secure, so its reasonably valid assumption [25].

Under this mini-scheme, the bank produces a single quantum note $\$$. The goal of the note adversary is, after finite interactions with the bank, to produce two quantum notes $\$_1$ and $\$_2$ which successfully passes the verification test of the bank. In this scheme, since the bank produces only a single note $\$$, hence it does not require an attached classical serial number.

Definition 2. *Private quantum money mini-scheme. A quantum money mini-scheme with classical verification consists of an algorithm by the bank, and a protocol, verification, such that,*

1. *The bank algorithm produces a quantum note* $\$ = \rho$ *where* ρ *is a quantum state of the note.*
2. *Verification is a protocol with classical communication that is run on the note* $\$$*, between the noteholder H who claims to possess the note* $\$$ *and the bank. The output of the protocol is a bit declared by the bank to denote whether the note is valid or not. We denote this final bit as* $Ver_H^B(\$)$ *which is 1 when the bank validates the note and 0 otherwise.*

For this scheme to be secure, it must satisfy two important properties,

- Correctness: the scheme is ϵ correct if, for every honest holder H, it holds that

$$\mathbb{P}[\mathrm{Ver}_H^B(\$) = 1] \geqslant 1 - \epsilon \tag{3}$$

- Unforgeability: the scheme is ϵ unforgeable if for any quantum adversary who possesses the note $, has interacted a finitely bounded number of times with the bank and has managed to produce two notes $\$_1$ and $\$_2$, it holds that,

$$\mathbb{P}\left[\left(\mathrm{Ver}_H^B(\$_1) = 1 \wedge \mathrm{Ver}_H^B(\$_1) = 1\right)\right] \leqslant \epsilon \tag{4}$$

where H is any honest noteholder.

To go from a private quantum money mini-scheme to a full scheme, it is enough for the bank to add a serial number to a note of the mini-scheme. Then the bank can just run the verification protocol of the mini-scheme for that note with the serial number. We, therefore, propose a quantum money mini-scheme and rely on the above results to extend this mini-scheme into full scheme.

2.2. Tools for the Money Scheme

In this sub-section, we describe the primary tool required for the construction of our money scheme, the sampling matching problem. This problem was originally defined by Kumar et al. to demonstrate a quantum advantage in a one-way communication complexity setting [23]. We use a variant of the original problem to construct the verification scheme for the honest verifier.

2.2.1. Sampling Matching Problem

The sampling matching problem as illustrated in Figure 1 consists of two players, Alice and Bob. For any positive integer n, Alice receives a binary string $x \in \{0,1\}^n$. Bob, on the other hand, does not receive any input. His task is to sample a tuple (k,l) on the complete graph of n vertices (with the vertices being indexed with numbers $\{1, 2, \cdots, n\}$) uniformly at random from a set of $\mathcal{T}_n$ containing $n(n-1)/2$ distinct tuples. An example of the tuple set for $n = 4$ is $\mathcal{T}_4 :$ $\left[(1,2), (3,4), (1,3), (2,4), (1,4), (2,3)\right]$.
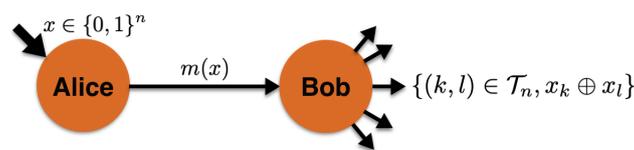
Figure 1. Sampling matching problem. Alice receives an input $x \in \{0,1\}^n$ and Bob does not receive any input. Alice sends a message $m(x)$ to Bob who outputs the tuple $\{(k,l) \in \mathcal{T}_n, x_k \oplus x_l\}$ where the from the message $m(x)$, a tuple (k,l) is sampled from the set of possible distinct tuples $\mathcal{T}_n$. Bob's objective is to output the parity correctly with high probability.

The objective of the problem is for Bob to output any tuple (k,l) from $\mathcal{T}_n$ and the parity $x_k \oplus x_l$ (where x_k, x_l are the kth and lth bit of x respectively). We look at the model of one-way communication where we only allow a single message from Alice to Bob.

For our quantum money proposal, we study the case when Alice is the untrusted noteholder and Bob is an honest note verifier. In the following, we construct a scheme for Bob to sample a tuple (k,l) from the set $\mathcal{T}_n$ when Alice sends a quantum message to Bob. We analyze Bob's scheme when the message sent by Alice is a single-photon state in a superposition over n modes.

2.2.2. Sampling Matching Scheme with Single-Photon States

Sampling matching scheme is Bob's testing scheme to sample the parity outcome of a tuple from the set $\mathcal{T}_n$ containing $n(n-1)/2$ distinct tuples. Here we look at the testing scheme when Alice's quantum message to Bob is a single-photon state.

The technique is depicted as follows: When an honest Alice receives the binary string $x \in \{0,1\}^n$, she encodes the information of this string into a single-photon state in a superposition over n different modes,

$$|x\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} (-1)^{x_k} \hat{a}_k^\dagger |0\rangle, \tag{5}$$

where x_k is the k-th bit of the string x. The operator $\hat{a}_k^\dagger$ is the creation operator for the k-th mode, and, $\hat{a}_k^\dagger |0\rangle = |1\rangle_k$. Figure 2 illustrates a method to create of equal superposition state of Equation (5) by passing the initial state $\hat{a}^\dagger |0\rangle$ through the cascade of $n-1$ 50/50 beam splitters and adding the phase information of each bit of the input in the n modes.
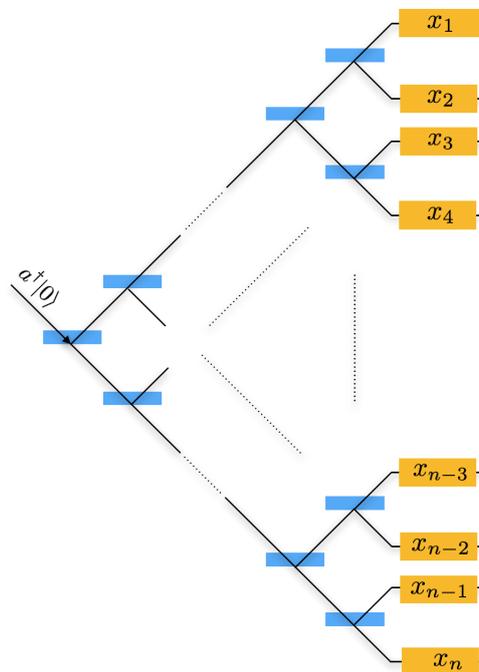

Figure 2. Superposition circuit of Alice to create a single-photon state in equal superposition over n modes. This is realized by passing a single-photon through a cascade of $n-1$ 50/50 beam splitters and subjecting each output mode to a phase-shift that depends on the corresponding string $x \in \{0,1\}^n$.

Alice sends this state $|x\rangle$ to Bob. In order to determine the parity outcome of a tuple, Bob first prepares his local superposition state in n-modes,

$$|\beta\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} \hat{b}_k^\dagger |0\rangle, \tag{6}$$

where $\hat{b}_k^\dagger$ is the creation operator for the k-th mode with $\hat{b}_k^\dagger |0\rangle = |1'\rangle_k$.

Bob's action is to apply mode-by-mode beam splitter operation on the state $|x\rangle \otimes |\beta\rangle$. This is illustrated in Figure 3.
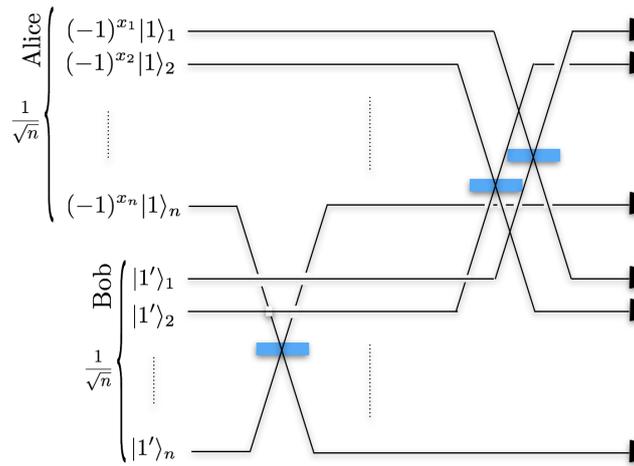
Figure 3. Sampling matching circuit model in single-photon encoding. Alice encodes a secret string $x \in \{0, 1\}^n$ in the single-photon state $|x\rangle$ in an equal superposition over n modes. This is then sent to Bob. Bob creates his local superposition state and applies mode-by-mode beam splitter operation with Alice's state. The results are observed in the $2n$ photon number resolving detectors.
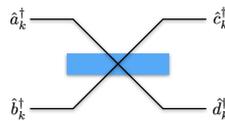
Prior to the beam splitter operation, the input of Bob is,

$$|x\rangle \otimes |\beta\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} (-1)^{x_k} \hat{a}_k^\dagger |0\rangle \otimes \frac{1}{\sqrt{n}} \sum_{l=1}^{n} \hat{b}_l^\dagger |0\rangle = \frac{1}{n} \sum_{k,l=1}^{n} (-1)^{x_k} \hat{a}_k^\dagger \hat{b}_l^\dagger |00\rangle \qquad (7)$$

where $|00\rangle = |0\rangle \otimes |0\rangle$.

At each mode $k \in [n]$, the beam splitter transforms the input operators $\{\hat{a}_k^\dagger, \hat{b}_k^\dagger\}$ into the output modes $\{\hat{c}_k^\dagger, \hat{d}_k^\dagger\}$ as depicted in Figure 4. This input to output mode conversion for the 50/50 beam splitter is given as,

$$\hat{a}_k^\dagger \to \frac{1}{\sqrt{2}} (\hat{c}_k^\dagger + \hat{d}_k^\dagger), \quad \hat{b}_k^\dagger \to \frac{1}{\sqrt{2}} (\hat{c}_k^\dagger - \hat{d}_k^\dagger) \qquad (8)$$

Figure 4. Illustration of a 50/50 beam splitter transforming input modes $\{\hat{a}_k^\dagger, \hat{b}_k^\dagger\}$ into the output modes $\{\hat{c}_k^\dagger, \hat{d}_k^\dagger\}$.

From Figure 3, we see that the k-th mode of Alice's state interacts with k-th mdoe of Bob's state, for all $k \in [n]$. The output operator $\hat{O}^\dagger$ corresponding to this joint interaction for all the modes is,

$$\begin{aligned}
\hat{O}^\dagger &= \frac{1}{n} \sum_{k,l=1}^{n} (-1)^{x_k} \frac{\hat{c}_k^\dagger + \hat{d}_k^\dagger}{\sqrt{2}} \cdot \frac{\hat{c}_l^\dagger - \hat{d}_l^\dagger}{\sqrt{2}}, \\
&= \frac{1}{2n} \sum_{k,l=1}^{n} (-1)^{x_k} (\hat{c}_k^\dagger \hat{c}_l^\dagger + \hat{d}_k^\dagger \hat{c}_l^\dagger - \hat{c}_k^\dagger \hat{d}_l^\dagger - \hat{d}_k^\dagger \hat{d}_l^\dagger), \\
&= \frac{1}{2n} \sum_{k=1}^{n} (-1)^{x_k} (\hat{c}_k^{\dagger 2} - \hat{d}_k^{\dagger 2}) + \\
&\quad \frac{1}{2n} \sum_{(k,l) \in \mathcal{T}_n} ((-1)^{x_k} + (-1)^{x_l}) (\hat{c}_k^\dagger \hat{c}_l^\dagger - \hat{d}_k^\dagger \hat{d}_l^\dagger) + \\
&\quad \frac{1}{2n} \sum_{(k,l) \in \mathcal{T}_n} ((-1)^{x_k} - (-1)^{x_l}) (\hat{d}_k^\dagger \hat{c}_l^\dagger - \hat{c}_k^\dagger \hat{d}_l^\dagger)
\end{aligned} \qquad (9)$$

where the commutation $[\hat{c}_k^\dagger, \hat{d}_l^\dagger] = 0$ for all $k, l \in [n]$. This is because by notation $\hat{c}^\dagger$ acts on the first qubit while the $\hat{d}^\dagger$ acts on the second qubit. Here $\mathcal{T}_n$ is the set of all possible $\frac{n(n-1)}{2}$ distinct tuples. The output state is,

$$\hat{O}^\dagger |00\rangle \qquad (10)$$

The output Equation (10) is then observed in the photon number resolving detectors. Since there are two input photons to the circuit, Bob will observe two-photons clicks across the $2n$ output modes (here we assume perfect set-up and detection) labelled as $\{c_1, d_1, \cdots, c_n, d_n\}$. From Equation (9) Bob will observe one of the following scenarios,

- Simultaneous single-photon clicks in $\{c_k, c_l\}$ or $\{d_k, d_l\}$, for two distinct modes (k, l), implies $x_k \oplus x_l = 0$.
- Simultaneous single-photon clicks in $\{c_k, d_l\}$ or $\{d_k c_l\}$, for two distinct modes (k, l), implies $x_k \oplus x_l = 1$.
- Two photons in the same mode c_k or d_k does not reveal the parity outcome for Bob and hence results in inconclusive outcome.

We denote the output state corresponding to single-photon detection in the two distinct modes $\{c_k, d_l\}$ as $|11\rangle_{c_k, d_l}$. The same notation is followed for output states corresponding to single-photon detection in distinct modes $\{c_k, c_l\}$ or $\{d_k, d_l\}$. The output state corresponding to two-photon detection in the same modes c_k or d_k is denoted as $|20\rangle_{c_k}$ and $|02\rangle_{d_k}$ respectively.

From Equation (9), we see that the probability of observing two photons in the same mode c_k is,

$$p_2^{c_k} = |\langle 20|_{c_k} \hat{O}^\dagger |00\rangle|^2 = \frac{1}{2n^2}, \tag{11}$$

where we have used the property of creation operators that $\hat{c}_k^{2\dagger} |00\rangle = \sqrt{2} |20\rangle_{c_k}$. Similarly, the probability of observing two photons in the same mode $\hat{d}_k$ is,

$$p_2^{d_k} = |\langle 02|_{d_k} \hat{O}^\dagger |00\rangle|^2 = \frac{1}{2n^2}, \tag{12}$$

where the similar property of creation operators $\hat{d}_k^2 |00\rangle = \sqrt{2} |02\rangle_{d_k}$ has been used. Over all the $2n$ modes, the probability of having two photons in the same mode is,

$$p_2 = \sum_{k=1}^{n} p_2^{c_k} + p_2^{d_k} = \frac{1}{n}. \tag{13}$$

In these cases, Bob does not get a conclusive parity outcome of any two bits of x. When this occurs, he outputs the outcome $d = \varnothing$.

On the other hand, with probability $1 - \frac{1}{n}$, Bob always gets exactly two single-photon clicks in two different time modes $k, l \in [n]$ with correct parity outcome $d = x_k \oplus x_l$. The probability that he outputs the parity outcome of a tuple $(k, l) \in \mathcal{T}_n$,

$$p_{kl} = |\langle 00| \hat{T}_{kl} \cdot O^\dagger |00\rangle|^2 = \frac{2}{n^2}, \tag{14}$$

where $\hat{T}_{kl}^\dagger = \frac{1}{2\sqrt{2}} \left(((-1)^{x_k} + (-1)^{x_l})(\hat{c}_k^\dagger \hat{c}_l^\dagger - \hat{d}_k^\dagger \hat{d}_l^\dagger) + ((-1)^{x_k} - (-1)^{x_l})(\hat{c}_k^\dagger \hat{d}_l^\dagger - \hat{d}_k^\dagger \hat{c}_l^\dagger) \right)$ is the operator corresponding to the correct parity outcome for the tuple (k, l). Operationally this operator generates only single-photon clicks in modes $\{c_k, c_l$ or $d_k, d_l\}$ if $x_k \oplus x_l = 0$, while single-photon clicks are generated in $\{c_k, d_l$ or $d_k, c_l\}$ if the corresponding parity is 1. Note that for the incoming state $|x\rangle$ from honest Alice, if Bob observes simultaneous single-photon clicks in two distinct modes, it's parity outcome is correct with certainty. Hence if Alice is dishonest and sends a state different from $|x\rangle$, it is enough to show that after the mode-by-mode beam splitter interaction with Bob's state $|\beta\rangle$, there is a non-zero probability for Bob to obtain an incorrect parity outcome $x_k \oplus x_l$ across the tuples $(k, l) \in \mathcal{T}_n$. This forms the basis of our unforgeability test using sampling matching based verification.

3. Results

With the tools at our disposal, we can now formally introduce our private quantum money scheme using the verification protocol based on the sampling matching scheme.
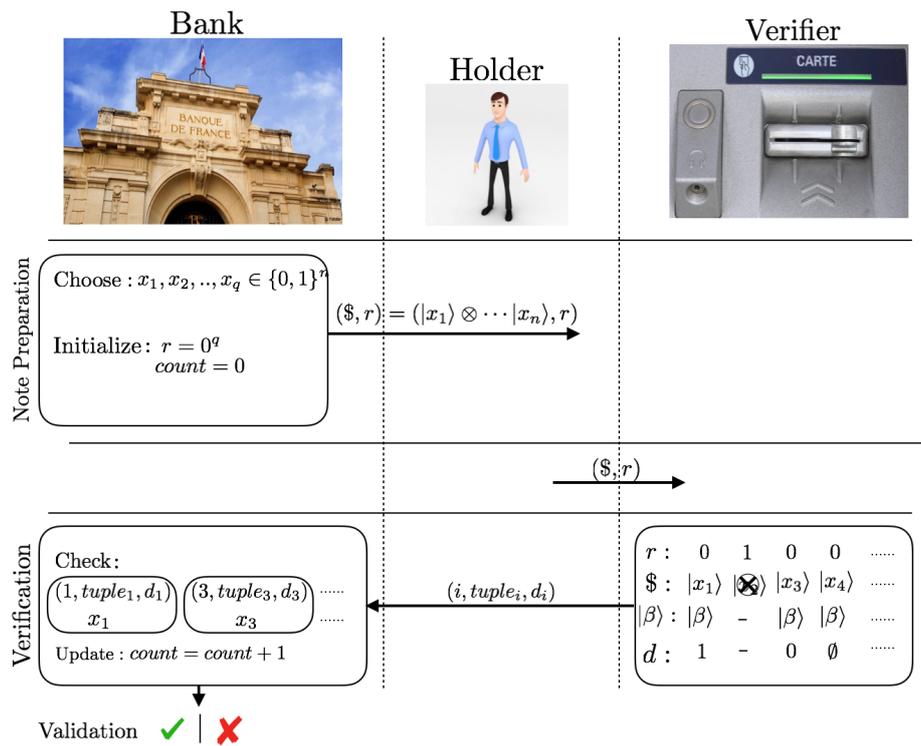
3.1. Private Quantum Money Scheme

Our quantum money scheme involves a bank that produces the notes which are quantum states and distributes them to untrusted noteholders. Each time a holder needs to carry out a transaction, he sends the note to the trusted local verifiers of the bank who help the bank decide the validity of the note. The salient features of our money scheme include:

- Note verification procedure requiring a single round classical communication between the local verifier and the bank,
- Fixed verification circuit for a given input size of the note,
- Multiple note re-usability, meaning the same note can be reused by the holder a number (linear in the size of the note) of times,
- Unconditional security against any adversary trying to forge the banknote while tolerating a noise of up to 21.4%.

Our scheme involves two phases. First is the note preparation phase, where the bank chooses multiple n-bit binary strings independently and uniformly randomly. Each of these strings are then encoded into single-photon states in superposition over n modes. The quantum note $ is the combination of single-photon states corresponding to all the chosen strings. This is then distributed among the untrusted holders. In the verification phase, the holder's note is sent to the local verifier for testing. The verifier randomly selects some copies of the note state (here the note consists of multiple copies, where one copy corresponds to the single-photon state that encodes one n-bit string). He runs the verification protocol using the SM scheme (Section 2.2.2) and does a local check based on the statistics of the classical outcome obtained. If the statistics are different from the ones produced by the note of an honest holder, he invalidates the note. If the note passes this test then the outcomes are classically communicated to the bank. The bank compares these outcomes with his private n-bit strings. If a high fraction of the outcomes is correct, he outputs the bit $\text{Ver}_H^B = 1$ implying that the note is valid. Otherwise, he outputs $\text{Ver}_H^B = 0$.

The money scheme we use here is the quantum money mini-scheme. Under this scheme, the bank produces a single quantum note $, consisting of many copies of single-photon states. The goal of the note adversary is, after finite interactions with the bank, to produce two quantum notes $_1$ and $_2$ which successfully pass the verification test by two independent verifiers. We have already emphasized that we make use of previous results [24] which lifts the security against any adversary from a private quantum money mini-scheme to the general scheme with multiple notes and a classical serial number assigned to them.

We now describe the quantum money mini-scheme based on single-photon states, 50/50 beam splitter linear optics transformations and photon number resolving detectors. The description of the scheme is provided in Figure 5.

Figure 5. Illustration of our private quantum money scheme based on the verification protocol using the sampling matching scheme. In the note preparation phase, the bank independently and uniformly randomly selects q n-bit binary strings $\{x_1, \cdots, x_q\}$ and prepares single-photon superposition note states $\$ = |x_1\rangle \otimes |x_2\rangle \otimes \cdots \otimes |x_q\rangle$. The bank further initializes a r register to keep a track of the number of positions in $[q]$ where the states have been used for verification and the count register to keep track of a number of verification attempts on the note. The note $(\$, r)$ is then sent to the holder. To be able to carry out any transaction, the holder sends the note to an honest verifier. In the verification phase, the verifier selects a fraction of the q copies of the note state which have $r = 0$. He creates his local state $|\beta\rangle$ and applies the sampling matching (SM)-scheme on those selected copies. The verifier sends the outcome of the measurement scheme to the bank. Finally, the bank compares the outcomes with his secret string x_j's and outputs a bit Ver_H^B stating whether the note is valid or not.

3.1.1. Note Preparation Phase

1. The bank independently and uniformly randomly chooses q n-bit binary strings $x_1, x_2, .., x_q \in \{0,1\}^n$
2. The bank encodes each binary string x_j into the single-photon state in superposition over n modes,

$$|x_j\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} (-1)^{x_{j,k}} \hat{a}_k^\dagger |0\rangle, \tag{15}$$

   where $x_{j,k}$ is the k-th bit value of string x_j and $\hat{a}_k^\dagger$ is the creation operator for the mode k with $\hat{a}_k^\dagger |0\rangle = |1\rangle_k$. The note is a combination of all the q copies of the single-photon state $\$ = |x_1\rangle \otimes |x_2\rangle \otimes \cdots \otimes |x_q\rangle$.
3. The bank creates a classical binary register r and initializes it to 0^q. This register keeps the track of positions j where the states have been used for the verification.
4. The bank creates a counter variable count and initializes it to 0. This keeps a track of the number of verification attempts.
5. The bank sends the quantum note ($\$$, r) to the holder.

3.1.2. Verification Phase

Once the bank distributes the note, the holder in order to carry out any transaction has to get the note verified from an honest local verifier Ver. The verification procedure is described below.

Local testing

1. The holder gives the note $\$'(=: \$$ if the holder is honest) to Ver.
2. Ver checks for the number of times the note has been re-used by verifying that the hamming distance of r register $d(r, 0^q) \leqslant T$, where T is a predefined maximum number of copies in the note that are allowed for verification. If $d(r, 0^q) > T$, the note is rendered useless.
3. Ver uniformly and randomly selects a subset $L \subset [q]$ copies from the states marked $r = 0$. All the corresponding L copies in the r register are then marked to 1.
4. For each chosen copy $j \in L$, Ver prepares his local coherent state $|\beta\rangle$ and runs the SM scheme (Section 2.2.2).
5. jVer first checks if he gets two photon clicks in all the chosen L copies. If not, he rejects (this is a check against all those attacks where the adversary removes the single-photon state and introduces either vacuum or a multi-photon state).
6. Ver counts the number of successful copies l_{succ}, where he obtains two single-photon clicks in two different modes. For these copies he outputs the parity outcome $d_j = x_{j,k} \oplus x_{j,l}$ where the clicks have been obtained in modes k and l. For the rest of the copies, he sets $d_j = \varnothing$.
7. Ver checks if $l_{succ} \geqslant l_{min}$, where $l_{min} = \mathbb{E}_h[l_{succ}](1 - \epsilon)$ is the minimum number of copies that will locally guarantee his acceptance of the note, where $0 \leqslant \epsilon \leqslant 1$ is the desired security factor. Here $\mathbb{E}_h[l_{succ}]$ is the expected number of copies where the honest noteholder obtains two single-photon clicks in two different modes when Ver runs the SM scheme.
8. Ver proceeds to the classical communication step with the bank only when the note passes this test.

Communication with the bank

1. Ver forwards the outcomes $\{j, (k, l), d_j\}$ for each $j \in L$ to the bank through a classical authenticated channel.
2. The bank checks if $count < \lceil \frac{T}{|L|} \rceil$, otherwise the verification attempt is rendered invalid. Here $\lceil \cdot \rceil$ is the ceiling function.
3. For each copy $j \in L$ with $d_j \neq \varnothing$, the bank compares the parity value d_j with the secret string x_j. He validates the note if the number of correct outcomes

$$l_{succ}^{cor} \geqslant \mathbb{E}_h[l_{succ}^{cor}](1 - \delta), \tag{16}$$

where $\mathbb{E}_h[l_{succ}^{cor}]$ is the expected number of copies that give the correct parity outcome when the noteholder is honest, and $0 \leqslant \delta \leqslant 1$ is a positive constant whose optimal value is determined by the forging probability (Section 3.3).
4. The bank updates the count by 1.

3.2. Correctness

Let us start by computing the probability that an honest noteholder fails the verification test. We use the Chernoff-Hoeffding inequality [26] to prove our results.

We first remark that the honest noteholder always passes step 5 of the verification phase, since he sends the entire banknote to the verifier Ver, who after performing the SM-scheme on the chosen L copies, always obtains the two-photon clicks.

However, the noteholder can fail the step 7 of the verification phase if the number of successful copies, where he obtains two single-photon clicks in two different time modes, $l_{succ} < l_{min} = \mathbb{E}[l_{succ}](1 - \epsilon)$, where $\mathbb{E}[l_{succ}]$ is the expected number of copies where Ver obtains two single-photon clicks in two different modes when he runs the SM scheme, and ϵ is the security parameter chosen

by Ver. Equation (13) tells us that for each of these chosen copy $j \in L$, the probability that the verifier obtains two single clicks in two different time modes is,

$$p_{11} = 1 - \frac{1}{n}. \tag{17}$$

Thus for L copies chosen from the note state, the expected number of successful copies is,

$$\mathbb{E}[l_{succ}] = |L|p_{11}. \tag{18}$$

Using the Chernoff–Hoeffding bound, the probability that the holder fails this test is,

$$\mathbb{P}[l_{succ} < l_{min}] \leqslant \exp\left(-\frac{2\epsilon^2 \mathbb{E}^2[l_{succ}]}{|L|}\right) = \exp\left(-2\epsilon^2 p_{11}^2 |L|\right). \tag{19}$$

Now after applying the SM scheme, Ver forwards the parity outcomes to the bank. From Equation (9), we see that whenever Ver obtains a parity outcome of a tuple (k, l), it is always correct. Thus the probability that the bank obtains correct parity outcome for each of the l_{succ} copy is $c = 1$. This implies $l_{succ}^{cor} = l_{succ}$ and that the only stage where an honest noteholder can fail is the local verifier stage.

$$\mathbb{P}[\text{Honest fail}] = \mathbb{P}[l_{succ} < l_{min}] \leqslant \exp\left(-2\epsilon^2 (1 - \frac{1}{n})^2 |L|\right). \tag{20}$$

This probability of failing goes down exponentially with $\epsilon, |L|$ and n.

3.3. Unforgeability of Banknotes

In this section, we explicitly calculate the forging probability for the adversary when she has in possession the valid banknote \$ and his objective is to duplicate the note to create two copies $\$_1$ and $\$_2$, which successfully pass the verification tests from two independent verifiers, Ver1 and Ver2, simultaneously (our proof technique has been inspired from the work of Amiri et al. [13] which is a private quantum money scheme with the verification based on hidden matching [27], another well-defined problem in one-way communication complexity. Kumar et al. [23] showed similarities between sampling matching and hidden matching communication problems. Here we manage to find that our proof of unforgeability has a similar structure to [13] even though the circuit to realize them is vastly different.)

Based on our scheme construction, any action by the adversary who is trying to maximize the forging probability can be put into three categories as specified detailed below. The first two categories, register manipulation and adaptive foraging, arise due to the addition of multiple reusability feature in our note. This can be leveraged by the adversary by performing 'auxiliary' verification attempts on the note and thereby learning more information on the note state leading to an increase in the forging probability. The third category relates to an optimal quantum operation performed on the note state such that the resulting states sent to Ver1 and Ver2 can pass the verification test with high enough probability. Our proof of unforgeability takes into account the optimal action by any adversary on all these three categories.

1. Register manipulation. First, we address how manipulating the r register can be used by an adversary to increase the forging probability. Since in each verification attempt the verifier chooses $|L|$ copies from the note state, and the maximum number of note verification attempts allowed by the bank is T, hence the adversary can set at most $(T - 1)|L|$ positions in the r register to 1 before sending the note state to the verifier. This ensures that the verifier does not render the note useless while checking the hamming distance of the r register. Further, since the verifier only selects copies from the note state marked 0 in the r register, setting maximum possible positions in r to 1 allows the adversary the need to deal with lesser copies (0 marked states) of the note state.

Suppose the adversary creates two notes $(\$_1, r_1)$ and $(\$_2, r_2)$ and sends it to the verifiers Ver1 and Ver2 respectively. If the adversary sets $r_1(j) = 0$ and $r_2(j) = 1$ for the j-th copy of the note, she is sure that Ver2 will not select the state at j-th position for verification. With this approach she can set a maximum of $(T - 1)|L|$ positions in r_1 and r_2 register to 1. In the positions where she has set $r_1 = 1$,

she can send the correct banknote states to Ver2, and similarly for the positions where she has set $r_2 = 1$, she can send banknote states to Ver1. This results in her exactly replicating the $2(T-1)|L|$ copies of the note state for both verifiers.

2. Adaptive forging. Let us now also consider the possibility of the adaptive attack by the adversary where multiple 'auxiliary' verification attempts made on the note state would help increase her forging probability. Since the bank allows a maximum of T verification attempts on the note, hence the adversary can use this to his advantage by querying for validation of $(T-2)|L|$ copies (since he needs to leave 2 verification attempts, one each from Ver1 and Ver2). We assume the worst-case scenario where the adversary completely obtains the information of the state for those copies used in auxiliary verification attempts.

The above two categories of attacks (r register manipulation and the adaptive attack) combined allow the adversary to have full information of a combined $(3T-4)|L|$ copies of the note state.

3. General forging. Now to prove the unforgeability condition, we consider what happens in the remaining $q' = q - (3T-4)|L|$ copies of the states sent to Ver1 and Ver2 where the adversary has no auxiliary information of the states and for which $r_1(j)$ and $r_2(j)$ are 0. In this scenario, an optimal adversary operation is to produce two-note states with q' copies each (one each for Ver1 and Ver2) such that the average fidelity of the prepared states with the correct note state is maximized. This maximization directly translates to the probability with which verifier obtains correct outcomes upon performing the sampling matching scheme. (The problem of fidelity estimation can be cast directly as a semi-definite optimization (SDP) problem to upper bound the adversary's forging probability [28].) There are two possible attack models that the adversary can employ on the note state. First is the collective attack based on optimal manipulation on individual copies of the note state. The second, more general model, is coherent attack where the adversary performs a combined operation on the entire note state. We start with the collective attack and subsequently use the results by Croke and Kent [29] to argue that any coherent strategy by the adversary is no more powerful than the optimal collective strategy.

3.1 Collective attack: here we look at the optimal manipulation strategy on individual copies of the note state. First, we remark that the adversary has to send a single-photon state across each the q' copies to Ver1 and Ver2, otherwise he fails the step 5 test in verification phase with certainty. For each copy $j \in [q']$, the adversary possesses the valid banknote state $|x_j\rangle$ (Equation (15)). Using this copy, his most general operation would be,

$$|x_j\rangle \otimes |\mathrm{anc}\rangle \xrightarrow{\varepsilon} \rho_{j,HV_1V_2}, \tag{21}$$

where ρ_{j,HV_1V_2} is a general tri-partite quantum state that the adversary (H) creates between her, Ver1 (V_1) and Ver2 (V_2). The state received by Ver1 is then $\eta_{x_j} = Tr_{HV_2}(\rho_{j,HV_1V_2})$, where $Tr_{HV_2}(.)$ is partial trace over adversary and Ver2's state. Similarly the state received by Ver2 is $\tau_{x_j} = Tr_{HV_1}(\rho_{j,HV_1V_2})$. Any normalized mixed state sent to Ver1 over n-modes can be expressed as,

$$\eta_{x_j} = \sum_{k,l}^{n} A_{kl} \hat{a}_k^\dagger |0\rangle \langle 0| \hat{a}_l, \tag{22}$$

where $\hat{a}_k^\dagger$ is the creation operator of the kth mode, and $A_{kl} \in \mathbb{C}$ for all $k,l \in [n]$. The normalization of the mixed state invokes the condition $\sum_{k=1}^{n} A_{kk} = 1$.

Ver1 runs the sampling matching scheme on the input state as shown in Figure 3. The input of the interaction of the adversary state with the local state of Ver1, $|\beta\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} \hat{b}_k^\dagger |0\rangle$, can be written as a combined density matrix state,

$$\eta_j^{In} = |\beta\rangle \otimes \eta_{x_j} \otimes \langle\beta| = \frac{1}{n} \left(\sum_{k=1}^{n} \hat{b}_k^\dagger |0\rangle \otimes \sum_{l,m}^{n} A_{lm} \hat{a}_l^\dagger |0\rangle \langle 0| \hat{a}_m \otimes \sum_{o=1}^{n} \langle 0| \hat{b}_o \right) \tag{23}$$

This state undergoes mode-by-mode beam splitter operation (Figure 3) resulting the transformation of the modes $\hat{a}_i \rightarrow \frac{\hat{c}_i + \hat{d}_i}{\sqrt{2}}$ and $\hat{b}_i \rightarrow \frac{\hat{c}_i - \hat{d}_i}{\sqrt{2}}$ for all $i \in [n]$. The corresponding output state is,

$$
\begin{aligned}
\eta_j^{Out} &= \frac{1}{4n} \left(\sum_{k=1}^{n} (\hat{c}_k^\dagger - \hat{d}_k^\dagger) |0\rangle \otimes \sum_{l,m=1}^{n} A_{lm} (\hat{c}_l^\dagger + \hat{d}_l^\dagger) |0\rangle \langle 0| (\hat{c}_m + \hat{d}_m) \otimes \sum_{o=1}^{n} \langle 0| (\hat{c}_o - \hat{d}_o) \right) \\
&= \frac{1}{4n} \left(\sum_{k,l,m,o=1}^{n} A_{lm} (\hat{c}_k^\dagger - \hat{d}_k^\dagger)(\hat{c}_l^\dagger + \hat{d}_l^\dagger) |00\rangle \langle 00| (\hat{c}_m + \hat{d}_m)(\hat{c}_o - \hat{d}_o) \right)
\end{aligned}
\tag{24}
$$

Similar to the analysis in Section 2.2.2, we first compute the probability (p_{11}) with which Ver1 observes single-photon clicks in two distinct modes. This probability $p_{11} = 1 - p_2$, where p_2 is total probability of observing two-photon clicks in the same mode. For output mode c_k,

$$
p_2^{c_k} = \langle 20|_{c_k} \, \eta_j^{Out} \, |20\rangle_{c_k} = \frac{A_{kk}}{2n}.
\tag{25}
$$

The corresponding probability of observing two-photon clicks in mode d_k is,

$$
p_2^{d_k} = | \langle 02|_{d_k} \, \eta_j^{Out} \, |02\rangle_{d_k} = \frac{A_{kk}}{2n}.
\tag{26}
$$

Summing over all the $2n$ modes,

$$
p_2 = \sum_{k=1}^{n} p_2^{c_k} + p_2^{d_k} = \sum_{k=1}^{n} \frac{A_{kk}}{n} = \frac{1}{n}.
\tag{27}
$$

where we have used the normalization condition $\sum_{k=1}^{n} A_{kk} = 1$. Comparing Equation (13) and Equation (27), we see that the total probability of obtaining two photons in the same mode for an adversary is the same as that for an honest noteholder. Thus for any adversarial state with a single-photon over n modes, Ver1 observes two single-photon clicks in two distinct output modes with a probability $p_{11} = 1 - \frac{1}{n}$. Over the L copies chosen by Ver1, he receives on average $|L| p_{11}$ copies with single-photon clicks in two distinct modes. This implies that the adversary passes the local step 7 of Ver1's verification phase test with probability,

$$
\mathbb{P}[\text{Ver1 accepts}] = \mathbb{P}[l_{succ} \geqslant l_{min}] \geqslant 1 - \exp\left(-2\epsilon^2 p_{11}^2 |L|\right),
\tag{28}
$$

where l_{succ} is the total copies where he gets single-photon clicks in two different modes and $l_{min} = |L| p_{11} (1 - \epsilon)$. If the adversary passes this test then Ver1 communicates the parity outcomes of l_{succ} copies to the bank who checks the outcomes with his secret strings.

We now calculate the average probability that the tuple $(e, f) \in \mathcal{T}_n$ that Ver1 obtains for the j-th copy, returns an incorrect parity outcome. Based on the analysis in Section 2.2.2, we see that an incorrect outcome is obtained if Ver1 obtains single-photon clicks in the modes $\{c_e, c_f\}$ or $\{d_e, d_f\}$ when $x_{j,e} \oplus x_{j,f} = 1$, while single-photon clicks are obtained in $\{c_e, d_f\}$ or $\{d_e, c_f\}$ if the corresponding parity is 0. This probability of obtaining an incorrect parity outcome for the tuple $(e, f) \in \mathcal{T}_n$ is,

$$
p_{Ver1}^{ef,inc} = \langle 00| \, \hat{I}_{ef} \cdot \eta_j^{Out} \cdot \hat{I}_{ef}^\dagger \, |00\rangle,
\tag{29}
$$

where $\hat{I}_{ef}^\dagger = \frac{1}{2\sqrt{2}} \left((-1)^{x_{j,e}} (\hat{c}_e^\dagger \hat{c}_f^\dagger - \hat{d}_e^\dagger \hat{d}_f^\dagger + \hat{c}_e^\dagger \hat{d}_f^\dagger - \hat{d}_e^\dagger \hat{c}_f^\dagger) - (-1)^{x_{j,f}} (\hat{c}_f^\dagger \hat{c}_e^\dagger - \hat{d}_f^\dagger \hat{d}_e^\dagger + \hat{c}_f^\dagger \hat{d}_e^\dagger - \hat{d}_f^\dagger \hat{c}_e^\dagger) \right)$ is the operator for the tuple (e, f) that corresponds to incorrect parity outcome of the e-th and f-th bit of bank's secret string x_j. Comparing with the operator $\hat{T}_{ef}^\dagger$ in Equation (14), we observe that the operator $\hat{I}_{ef}^\dagger$ has a negative sign in the second term, which corresponds to an incorrect outcome. Operating $\hat{I}_{ef}^\dagger$ on the banknote state $|x_j\rangle$, results in 0 probability of an outcome since the banknote state always

outputs a correct outcome. However, we use the fact that $p_{Ver1}^{ef,inc} \neq 0$ for any η_{x_j} different from $|x_j\rangle\langle x_j|$, and further this probability relates to how far these two states are in the metric of fidelity.

Over all the tuples in $\mathcal{T}_n$, the probability that Ver1 obtains an incorrect outcome is,

$$
\begin{aligned}
p_{Ver1}^{x_j,inc} &= \sum_{(e,f)\in\mathcal{T}_n} p_{Ver1}^{ef,inc} \\
&= \frac{1}{2n} \sum_{(e,f)\in\mathcal{T}_n} (A_{ee} + A_{ff} - (-1)^{x_{j,e}\oplus x_{j,f}}(A_{ef} + A_{fe})) \\
&= \frac{1}{2n}(n - \sum_{e,f}^{n}(-1)^{x_{j,e}\oplus x_{j,f}}A_{ef}) \\
&= \frac{1}{2}(1 - F_{x_j}),
\end{aligned}
\tag{30}
$$

where $F_{x_j} = \langle x_j| \eta_{x_j} |x_j\rangle = \frac{1}{n}\sum_{e,f}^{n}(-1)^{x_{j,e}\oplus x_{j,f}}A_{ef}$ is the square of fidelity between the adversary's state η_{x_j} and the banknote state $|x_j\rangle$. For simplicity we refer to this squared fidelity as fidelity.

The above probability is calculated for a specific string x_j. Since the adversary does not know this string, she instead holds a state which is a mixture of all possible 2^n strings x_j. Thus the averaged out error probability for Ver1 is,

$$
p_{Ver1}^{inc} = \frac{1}{2^n}\sum_{x_j} p_{Ver1}^{x_j} = \frac{1}{2}(1 - F),
\tag{31}
$$

where $F = \frac{1}{2^n}\sum_{x_j} F_{x_j}$.

Similar analysis for Ver2, who receives the mixed state τ_{x_j}, the fidelity with the honest note state is $G_{x_j} = \langle x_j| \tau_{x_j} |x_j\rangle$. The average error probability of obtaining an incorrect outcome is $p_{Ver2}^{inc} = \frac{1}{2}(1 - G)$, where $G = \frac{1}{2^n}\sum_{x_j} G_{x_j}$. We cast the objective problem for the adversary to minimize the average error probability of Ver1 and Ver2,

$$
p_{Ver1}^{inc} + p_{Ver2}^{inc} = 1 - \frac{F + G}{2}.
\tag{32}
$$

This minimization problem can be cast as a semi-definite program (SDP) with the objective to find a lower bound of Equation (32). This can alternatively be viewed as maximizing the average fidelity $\bar{F} = \frac{F+G}{2}$.

The security proof in quantum money proposal by Amiri et al. [13] has a similar reduction of the average error probability to maximizing the average fidelity of the states received by the two verifiers, Ver1 and Ver2, with the original banknote state. Thus one is interested in finding the completely-positive trace-preserving physical channel that takes a valid banknote state and prepares one copies for Ver1 and Ver2 with maximal average fidelity. Translating this problem in an SDP formalism requires using the Choi–Jamiolkowski representation [30] that translates the channel into a corresponding state in a higher dimension using maximally entangled state $|\psi\rangle = \sum_{i=1}^{n}|1\rangle_i|1\rangle_i$. Here we directly use the SDP result of [13]. Further details on this SDP construction can be found directly in their work. They numerically verified that for $n \leqslant 14$,

$$
\bar{F} \leqslant \frac{1}{2} + \frac{1}{n}.
\tag{33}
$$

This implies that the average fidelity of the states received by Ver1 and Ver2 with the banknote state is upper bounded by a factor less than 1 for $n \geqslant 4$. Further, the upper bound on average fidelity decreases with increasing n and if Equation (33) is true for any n (conjectured by [13]), then for $n \gg 4$, any optimal strategy by the adversary results in average fidelity that is upper bounded by 0.5.

Equation (33) allows us to give a lower bound on the average probability that Ver1 and Ver2 obtains an incorrect outcome of a tuple in $\mathcal{T}_n$,

$$
\begin{aligned}
p_{Ver1}^{inc} + p_{Ver2}^{inc} &= 1 - \frac{1}{2}(F + G) \\
&\geqslant \frac{1}{2} - \frac{1}{n}.
\end{aligned}
\tag{34}
$$

This is the probability for a single copy j chosen by the verifier. Across each of the chosen L uniformly random copies, the average error probability is lower bounded by Equation (34) in the collective adversary attack scenario. Since the above equation gives us a lower bound on the average error probability for both verifiers, this implies that the minimum error probability e_{min} for any one of the two verifiers, lets say Ver1, must definitely be,

$$
e_{min} = \frac{1}{4} - \frac{1}{2n}.
\tag{35}
$$

The above error probability has been calculated for $q' = q - (3T - 4)|L|$ copies. Over the remaining $q - q' = (3T - 4)|L|$ copies, we assume that the adversary has full information of the state. Hence the error probability in obtaining the parity outcome of a tuple for the $q - q'$ copies is 0 for Ver1 (same for Ver2). The minimum error probability for Ver1 averaged across the l_{succ} copies (where single clicks have been obtained across two distinct modes) is,

$$
e_{min} = \frac{q - (3T - 4)|L|}{q - (T - 1)|L|}\left(\frac{1}{4} - \frac{1}{2n}\right),
\tag{36}
$$

where the denominator $q - (T - 1)|L|$ is due the fact that the adversary can set $(T - 1)|L|$ positions in the r register to 1, thus effectively ensuring that Ver1 does not select the note copies in those positions. Suppose $T|L| = \lambda q$, for some small fraction $\lambda < 1$ (for example $1/1000$), then Equation (36) is,

$$
e_{min} \approx \frac{997}{999}\left(\frac{1}{4} - \frac{1}{2n}\right) \approx \frac{1}{4} - \frac{1}{2n}.
\tag{37}
$$

We know that if the holder is honest, the probability of him obtaining the correct parity outcomes across all the l_{succ} copies is $c = 1$. From Equation (37), we see that the corresponding probability of obtaining correct parity outcomes for the adversary is upper bounded by $c_{adv} = 1 - e_{min}$. The gap $c - c_{adv}$ is defined as the noise tolerance of our protocol since it denotes the maximum theoretical probability that an honest noteholder returns an incorrect outcome while still maintaining a non-negative gap in the success probability with any strategy by the adversary. Denoting the cut-off $\delta = (c - c_{adv})/2$ and using the Chernoff–Hoeffding bound [31], we can now compute the probability that adversary's note passes the test of verification phase by both Ver1 and Ver2 is,

$$
\begin{aligned}
\mathbb{P}[\text{Adv pass}] &= \mathbb{P}[\text{Ver1 and Ver2 accept}] \times \mathbb{P}[\text{Ver1}_H^B = 1 \text{ and Ver2}_H^B = 1 | \text{Ver1 and Ver2 accept}] \\
&\leqslant \mathbb{P}[\text{Ver1 accepts}] \cdot \mathbb{P}[\text{Ver1}_H^B = 1 | \text{Ver1 accepts}] \\
&\leqslant \mathbb{P}[\text{Ver1}_H^B = 1 | \text{Ver1 accepts}] \\
&\leqslant \exp\left(-\frac{2\delta^2 l_{min}^2}{|L|}\right) = \exp\left(-2\delta^2 p_{11}^2 (1 - \epsilon)^2 |L|\right)
\end{aligned}
\tag{38}
$$

The condition $c > c_{adv}$ always holds as long as $n > 2$, hence the probability that both verifiers pass the verification test is exponentially low. Since the Equation (33) has been verified until $n = 14$, the maximum noise tolerance of the scheme is up to 21.4%. Further, if Equation (33) is true for all n, the maximum noise asymptotic noise tolerance of 25% can be achieved with our scheme.

3.2 Coherent attack: The collective attack strategy focuses on the optimal manipulation of individual copies of the note state. However, the adversary can perform a combined operation on all the copies of the input of the state to potentially decrease the average error probability on each copy of the note state lower than the bound Equation (34). Alternatively, the adversary can hope for a general entangled strategy on the T chosen copies of the verifiers such that conditional on the measurement outcomes obtained in the first $T-1$ copies, the value of average error probability for the last copy is decreased even below the bound of Equation (34). To mitigate any such strategy, we use the results of Croke and Kent [29] which proves the security of a quantum relativistic bit commitment protocol against any adversary. Their reduction of a coherent attack strategy to a collective strategy uses the teleportation based argument to show that any error probability achieved via coherent strategy can also be achieved by individual manipulation of the states by an adversary who uses a maximally mixed entangled state to teleport the original state to the verifier (along with the teleportation correction). Also, since the collective attack is optimal manipulation of individual states, hence it also includes teleportation based strategies. Our setting is very similar to their construction and allows us to use their results in a straightforward manner. Therefore, any coherent strategy cannot beat the lower bounds proved in the collective strategy.

4. Measure and resend attack: Until now we have analyzed a generic way to upper bound the success probability in any adversary attack scenario. To gain intuition on how an adversary attack would play out, we consider a specific example attack called 'measure and resend'. Here for simplicity, we consider that the adversary does not perform register manipulation or adaptive attack, but rather she simply measures each copy of the incoming state and creates two states based on the measurement outcome to be sent to Ver1 and Ver2.

Assume that the bank sends a note with $q = 10^6$ copies of single-photon states to the holder. Each copy is encoded with bank's secret string $x_j \in \{0,1\}^4$ for $j \in [10^6]$. For verification, $|L| = 10^3$ copies of the note states are chosen at random and sampling matching scheme is performed on each of the copies to obtain a parity measurement outcome.

Suppose the strategy by an adversary is to measure each banknote copy $|x_j\rangle$ in the basis,

$$\left\{ \frac{1}{\sqrt{2}}(|1\rangle_1 \pm |1\rangle_2), \quad \frac{1}{\sqrt{2}}(|1\rangle_3 \pm |1\rangle_4) \right\}. \tag{39}$$

With this strategy, Bob can always provide the correct parity outcome either the tuple $\{1,2\}$ or $\{3,4\}$. This is because the outcome $\frac{1}{\sqrt{2}}(|1\rangle_1 + |1\rangle_2)$ only occurs if $x_{j,1} \oplus x_{j,2} = 0$, while $\frac{1}{\sqrt{2}}(|1\rangle_1 - |1\rangle_2)$ only occurs if $x_{j,1} \oplus x_{j,2} = 1$. Similarly the outcomes $\frac{1}{\sqrt{2}}(|1\rangle_3 \pm |1\rangle_4)$ provide a prarity outcome for the tuple $\{3,4\}$. Note that since the state $|x_j\rangle$ contains $\log_2 n$ qubits/bits of information, any strategy by the adversary would limit her information retrieval to only $\log_2 n$ bits. This is a consequence of the Holevo's bound [32] which states that it is impossible to retrieve more than $\log_2 n$ bits of information from a state of the form $|x_j\rangle$. Thus the adversary cannot retrieve the entire information of the secret string x_j and hence she cannot forge the note perfectly.

Using the above strategy, the adversary gets the parity information of exactly one tuple. In the resend phase to Ver1 and Ver2, he correctly encodes that parity information, while for the rest bits of which he has no information, he randomly encodes them in 0 or 1. As an example, for the j-th copy, if the adversary get the information of the tuple $\{1,2\}$, the state she sends to both verifiers Ver1 and Ver2 is

$$|\text{adv}\rangle = \frac{1}{\sqrt{2}}((-1)^{r_1}|1\rangle_1 + (-1)^{r_2}|1\rangle_2 + (-1)^{r_3}|1\rangle_3 + (-1)^{r_4}|1\rangle_4), \tag{40}$$

where he picks r_1, r_2 such that $r_1 \oplus r_2 = x_{j,1} \oplus x_{j,2}$, and r_3, r_4 are picked at random from 0/1. Upon receiving the note, Ver1 (similarly Ver2) runs the sampling matching scheme with his local $|\beta\rangle = \frac{1}{2}\sum_{i=1}^{4}|1\rangle_i$ on the randomly chosen 10^3 copies of the note state. From Equations (13) and (27), we see that across $p_2 \times 10^3 = 250$ copies, Ver1 will receive two photon clicks in the same mode on average

thus not obtaining the parity information of any tuple. Across the remaining 750 copies on average, he receives the parity outcome for any one of the three tuples $\{\{1,2\},\{1,3\},\{2,3\}\}$ with equal probability. Since the adversary managed to successfully extract only a single parity outcome, so the probability that the parity outcome obtained by Ver1 is wrong would be,

$$p_{Ver1}^{x_j,inc} = \sum_{\text{tuple}} p_{\text{tuple}} \times p_{\text{correct}|\text{tuple}} = \frac{1}{3}(0 + \frac{1}{2} + \frac{1}{2}) = \frac{1}{3}, \tag{41}$$

since the adversary will be correct in one parity outcome and the random guessing for remaining two parity outcomes gives him a success rate of 0.5. Thus the sucess probability of the adversary on each copy is $c_{\text{adv}} = 2/3$. If the noteholder was honest, this success probability $c = 1$ since the banknote was not tampered with. Denoting the cut-off $\delta = (c - c_{\text{adv}})/2 = 1/6$ and using Chernoff–Hoeffding bound, the probability that adversary's note passes the test of Verification phase (whene Ver1 communicates the parity outcome to the bank) is,

$$\begin{aligned}
\mathbb{P}[\text{Adv pass}] &= \mathbb{P}[\text{Ver1 accepts}] \cdot \mathbb{P}[\text{Ver1}_H^B = 1 | \text{Ver1 accepts}] \\
&\leqslant \mathbb{P}[\text{Ver1}_H^B = 1 | \text{Ver1 accepts}] \\
&\leqslant \exp\left(-2\delta^2(1 - p_2)|L|\right) \\
&\leqslant \exp(-41) \approx 10^{-17}
\end{aligned} \tag{42}$$

Thus we see that it is virtually impossible for an adversary to forge the note with this message and resend attack.

3.4. Quantum Money Scheme with Coherent states

In this section, we briefly describe the private quantum money scheme when the bank encodes secret strings into weak coherent states instead of the single-photon states. The primary reason we want to encode the note as coherent states is that it facilitates the implementation of sampling matching in a much simpler and elegant manner. With coherent states, as shown by Kumar et al. [23], the sampling matching implementation requires a single 50/50 beam splitter and two single-photon threshold detectors irrespective of input size of the strings.

In the coherent state encoding, the bank independently and randomly chooses q n-bit binary strings $x_1, x_2, .., x_q \in \{0,1\}^n$. Each string x_j is now encoded into the coherent state $\left|\alpha_{x_j}\right\rangle$, with an average photon number 1,

$$\left|\alpha_{x_j}\right\rangle = \bigotimes_{k=1}^{n} \left|(-1)^{x_{j,k}} \frac{1}{\sqrt{n}}\right\rangle_k, \tag{43}$$

where $x_{j,k}$ is the kth bit value of string x_j. The coherent state $\left|\alpha_{x_j}\right\rangle$ is a sequence of n coherent pulses in n modes. The banknote is then the sequence of q coherent states,

$$\$ = \bigotimes_{j=1}^{q} \left|\alpha_{x_j}\right\rangle. \tag{44}$$

This is then distributed among the untrusted noteholders. To carry out a transaction, the holders send the note to the local verifiers. The verifier uniformly and randomly selects few copies of the note state to run the sampling matching scheme. In this scheme, for each selected copy $\left|\alpha_{x_j}\right\rangle$, the verifier prepares his local state $|\beta\rangle = \bigotimes_{k=1}^{n} \left|\frac{1}{\sqrt{n}}\right\rangle_k$ as a sequence of n coherent pulses. This state sequentially interacted with the selected states of the note chosen by the verifier. The interaction is via the 50/50 beam splitter. The coherent pulse modes at the input of verifier's beam splitter in k-th step are

$$\left| (-1)^{x_{j,k}} \frac{1}{\sqrt{n}} \right\rangle_k \otimes \left| \frac{1}{\sqrt{n}} \right\rangle_k , \tag{45}$$

and the output modes are

$$\left| \frac{(1 + (-1)^{x_{j,k}})}{\sqrt{2}} \frac{1}{\sqrt{n}} \right\rangle_{k,D_0} \otimes \left| \frac{(1 - (-1)^{x_{j,k}})}{\sqrt{2}} \frac{1}{\sqrt{n}} \right\rangle_{k,D_1} . \tag{46}$$

The output modes are fed into the single-photon threshold detectors D_0 and D_1 to observe the clicks. When a coherent state $|\alpha\rangle$ is incident on the threshold detector, the probability of the click is given by,

$$p_c = 1 - \exp\left(-|\alpha|^2 \right). \tag{47}$$

Figure 6 is a depiction of the sequential interaction of the coherent pulses of one copy of an honest noteholder and the verifier's local state.
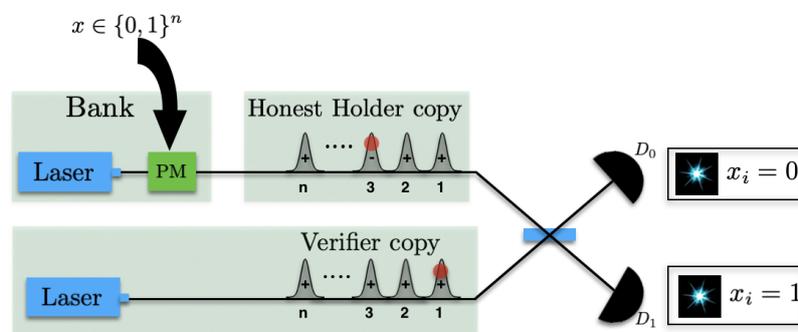
Figure 6. SM circuit implementation using weak coherent states, 50/50 beam splitter (BS) and single-photon threshold detectors. The upper arm illustrates an honest holder's state as a coherent state $|\alpha_x\rangle$, which consists of a sequence of coherent pulses. The coherent state is encoded with a random phase $x \in \{0,1\}^n$ through the phase modulator (PM). The lower arm is used by the verifier to produce a local reference coherent state $|\beta\rangle$, consisting of a sequence of coherent pulses, with an average photon number of 1. The verifier interferes with the states into the 50/50 BS and infers the parity information from the detector clicks in D_0 and D_1. This allows him to obtain the parity outcome of a tuple in $\mathcal{T}_n$. The red dot in the 1st and 3rd time sequence denotes that the verifier observed clicks at D_1 and D_0 detectors respectively for these time steps. Thus he infers the parity outcome for the tuple $(1,3)$, $x_1 \oplus x_3 = 1$.

Let us see how the verifier obtains the parity outcomes of one of the tuples in $\mathcal{T}_n$ from the detector clicks. The output state in Equation (46) denotes that the detector D_0 clicks iff $x_k = 0$ while D_1 clicks iff $x_k = 1$. For each of the chosen copy j, the verifier will be unable to infer the parity outcome of any matching with certainty if he does not obtain single-click in atleast two time steps. This probability $\mathbb{P}$(no two single-clicks) = $\mathbb{P}$(no single-clicks) + $\mathbb{P}$(exactly one single-click). We denote this probability by $p_{\neg 11}$,

$$p_{\neg 11} = (1 - p_1)^n + \binom{n}{1} p_1 (1 - p_1)^{n-1}, \tag{48}$$

where $p_1 = 1 - \exp\left(-\frac{2}{n}\right)$ is the probability of observing a single click in one time step. Thus with probability $1 - p_{\neg 11}$, which grows with n, the verifier obtains two single-photon clicks in the chosen state.

Now suppose the verifier observes single-photon clicks in the k-th and l-th time modes in detectors D_0 and D_1 respectively. This implies $d = x_k \oplus x_l = 1$. This enables the verifier to output the parity outcome of $(k,l) \in \mathcal{T}_n$. If on the other hand, the verifier does not obtain exactly two clicks in two different time modes, he outputs the parity outcome $d = \varnothing$.

3.4.1. Description of the Money Scheme

We divide our quantum money scheme using coherent states into two phases. First is the note preparation phase, where the bank chooses multiple n-bit binary strings independently and randomly. The bank takes each of these individual strings to produce weak coherent states. The quantum note $ of the bank is the combined tensor product of coherent states corresponding to all the input strings. This is distributed among the untrusted holders. In the verification phase, a noteholder sends the note to the verifier in order to carry out a transaction. Upon receiving the note, the verifier randomly selects some copies of the note state (here the note consists of multiple copies, where one copy corresponds to the coherent state that encodes one n-bit string). For the selected copies, he runs the verification protocol using the sampling matching, SM scheme (Figure 6). He locally checks if the statistics of the measurement outcome obtained by running the SM-scheme is what he should expect from an honest noteholder. If he finds discrepancies, the note is rejected. If the note passes this test, then the outcomes from the SM-scheme are classically communicated with the bank. The bank compares these outcomes with his private n-bit strings. If a high fraction of the outcomes is correct, he outputs the bit $\text{Ver}_H^B = 1$ implying that the note is valid. Otherwise, he outputs $\text{Ver}_H^B = 0$. The description of the scheme is provided in Figure 7.
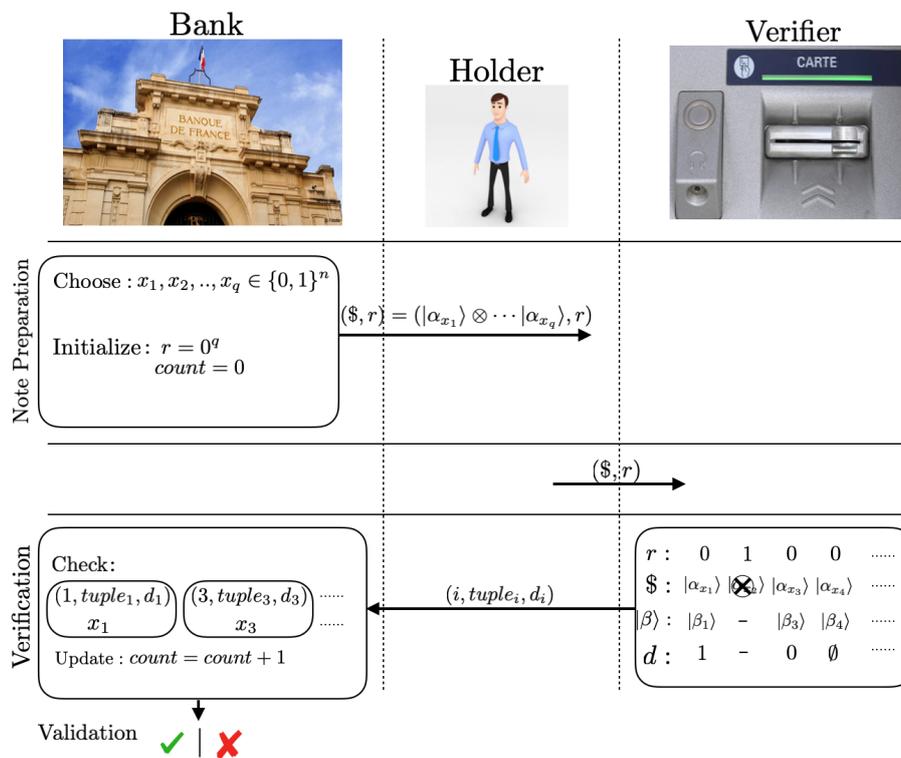
Figure 7. Illustration of our quantum money scheme based on the verification protocol using the SM-scheme. In the note preparation phase, the bank independently and randomly selects q n-bit binary strings and produces note coherent states $\$ = |\alpha_{x_1}\rangle \otimes |\alpha_{x_2}\rangle \otimes \cdots \otimes |\alpha_{x_q}\rangle$. The bank initializes the r register to keep a track of the number of positions in $[q]$ where the states have been used for verification and the count register to keep track of the number of verification attempts on the note. The note $(\$, r)$ is then sent to the holder. To carry out a transaction, the holder sends the note to an honest verifier of the bank. In the verification phase, the verifier selects a fraction of the q copies of the note state with positions 0 in the r register. He creates his local state $|\beta_j\rangle$ and applies the SM-scheme on those selected copies. If the note passes the local test of the verifier, he sends the measurement outcomes of the test to the bank. Finally, the bank compares the outcomes with his secret string x_j's and outputs a bit Ver_H^B stating whether the note is valid or not.

3.4.2. Note Preparation Phase

1. The bank independently and randomly chooses q n-bit binary strings $x_1, x_2, .., x_q \in \{0,1\}^n$
2. The bank encodes each the binary string x_j into the phase randomized coherent state $\left|\alpha_{x_j}\right\rangle$, with an average photon number 1,

$$\left|\alpha_{x_j}\right\rangle = \bigotimes_{k=1}^{n} \left|(-1)^{x_{j,k}} \frac{1}{\sqrt{n}}\right\rangle_k, \tag{49}$$

    where $x_{j,k}$ is the kth bit value of string x_j. The coherent state $\left|\alpha_{x_j}\right\rangle$ is a sequence of n coherent pulses in n modes. The note is a combination of all the q copies of the coherent state \$ $= \left|\alpha_{x_1}\right\rangle \otimes \left|\alpha_{x_2}\right\rangle \otimes \cdots \otimes \left|\alpha_{x_q}\right\rangle$.
3. The bank creates a classical binary register r and initializes it to 0^q. This register keeps the track of positions j where the states have been used for the verification.
4. The bank creates a counter variable count and initializes it to 0. This keeps a track of the number of verification attempts.
5. The bank sends the quantum note (\$, r) to the holder.

3.4.3. Verification Phase

Once the bank distributes the notes, the holder in order to be able to carry out any transaction, has to get the note verified from an honest verifier Ver. The verification procedure is listed below.

Local testing

1. The holder gives the note \$$'$($=:$ \$ if the holder is honest) to Ver.
2. Ver checks the re-usability of the note by verifying that the hamming distance of r register $d(r, 0^q) \leqslant T$, where T is a predefined maximum number of copies in the note that are allowed for verification. If $d(r, 0^q) > T$, the note is rendered useless and must be returned to the bank.
3. Ver uniformly and randomly selects a subset $L \subset [q]$ copies from the states marked $r = 0$. He marks all the corresponding $|L|$ copies in the r register to 1.
4. For each copy $j \in L$, Ver prepares his local coherent state $\left|\beta_j\right\rangle = \bigotimes_{k=1}^{n} \left|\frac{1}{\sqrt{n}}\right\rangle_k$ and runs the SM scheme (Figure 6).
5. Ver counts the number of successful copies l_{succ}, where he obtains exactly two single-photon clicks in two different time modes. For these copies he outputs the parity outcome $d_j = x_{j,k} \oplus x_{j,l}$ where the clicks have been obtained in times modes k and l. For the rest of the copies, he sets $d_j = \varnothing$.
6. Ver checks if $l_{succ} \geqslant l_{min}$, where $l_{min} = \mathbb{E}_h[l_{succ}](1 - \epsilon)$ is the minimum number of copies that will locally guarantee his acceptance of the note, where $0 \leqslant \epsilon \leqslant 1$ is the security factor. Here $\mathbb{E}_h[l_{succ}]$ is the expected number of copies where the honest noteholder obtains exactly two single-photon clicks when Ver runs the SM scheme.
7. Ver proceeds to the communication with the bank only when the note passes this test.

Communication with the bank

8. Ver forwards the outcomes $\{j \in L, (k,l), d_j\}$ to the bank.
9. The bank checks if $count < \left\lceil \frac{T}{|L|} \right\rceil$, otherwise he renders the verification attempt as invalid.
10. For each copy $j \in L$ with $d_j \neq \varnothing$, the bank compares the parity value d_j with the secret string x_j. He validates the note if the number of correct outcomes

$$l_{succ}^{cor} \geqslant \mathbb{E}_h[l_{succ}^{cor}](1 - \delta), \tag{50}$$

where $\mathbb{E}_h[l_{succ}^{cor}]$ is the expected number of copies that give the correct parity outcome when the noteholder is honest, and $0 \leqslant \delta \leqslant 1$ is a positive constant.

11. The bank updates the count by 1.

Here we are not providing full security proof of our quantum money scheme using coherent states. This is due to the fact that the statements of security would be dependent on the specific experimental parametric values. The main objective of this section is to illustrate the simplicity and elegance of mapping the money scheme using coherent states which simplifies the verification circuit requirement to only a $\mathcal{O}(1)$ 50/50 beam splitters and two single-photon threshold detectors. Nevertheless, we expect that a full security proof can be constructed in a straightforward manner from the security proof given for the single-photon superposition states.

4. Discussion

We have introduced the private quantum money scheme as a cryptographic task using the sampling matching verification scheme. Sampling matching is an experimentally motivated verification framework to ease out the implementation of quantum money schemes. Our proposed money scheme demonstrates an information-theoretic security against any adversary with a noise tolerance of 21.4%. We mostly focus on the scheme with the banknotes being single-photon states. This has been done keeping in mind that single-photon states are a direct realization of qubit/qudit with linear optics, thus our money scheme definition can be directly translatable to other qubit encoding pictures. The security analysis of our money scheme considers the most general attack by the adversary trying to duplicate the banknotes which can be passed by the bank with high probability. In the second half, we have proposed a money scheme to encode the secret strings into a sequence of coherent states with the verification using sampling matching consisting of a single 50/50 beam splitter and two single-photon threshold detectors. This scheme provides a dramatic reduction in the component requirements by the verifier thus experimentally facilitate achieving higher noise tolerance than what is realistically feasible with other quantum money schemes.

Another major challenge in quantum money schemes is the storage of quantum states of the bank in a quantum memory which can then be distributed to the holder. There has been considerable ongoing works in the development of quantum memory for storing and retrieving the quantum data [33–36]. Even though this is not the specific focus of our work, as our approach is to simplify the verifier's circuit, we still address the above memory issue partially by introducing the verifier circuit (when the banknotes are encoded coherent states) that works even when the states are sent on the fly i.e., no memory is required by the verifier to store them. This is due to the fact that the coherent state encoding is a tensor product of individual coherent pulses and the beam splitter interaction with the verifier's state occurs separately on each of these pulses.

Finally, the verification tool using sampling matching is a generic tool that has the potential to be applied to other cryptographic tasks such as key-distribution, and also in quantum verification protocols such as efficient verification of prover's state with limited verifier resources in scenarios where the prover does not want to fully reveal the secret encoding [37]. Since most cryptographic schemes rely on one-way functions and the fact that it is hard for an adversary to invert the function without having the knowledge of the secret key that was used to prepare the function, the scheme using sampling matching offers an excellent candidate to be such an experimentally realizable function.

Abbreviations

The following abbreviations are used in this manuscript:

SM   sampling matching
HM   hidden matching
SDP   semi-definite programming

References

1. Wiesner, S. S. Wiesner, Sigact News 15, 78 (1983). *Sigact News* **1983**, *15*, 78. [CrossRef]
2. Wootters, W.K.; Zurek, W.H. A single quantum cannot be cloned. *Nature* **1982**, *299*, 802–803. [CrossRef]
3. Bennett, C.H.; Brassard, G. Quantum cryptography: Public key distribution and coin tossing. *Theor. Comput. Sci.* **2014**, *560*, 7–11. [CrossRef]
4. Gottesman, D.; Chuang, I. Quantum digital signatures. *arXiv* **2001**, arXiv:quant-ph/0105032.
5. Ambainis, A. A new protocol and lower bounds for quantum coin flipping. *J. Comput. Syst. Sci.* **2004**, *68*, 398–416. [CrossRef]
6. Broadbent, A.; Fitzsimons, J.; Kashefi, E. Universal blind quantum computation. In Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science, Atlanta, GA, USA, 24–27 October 2009; pp. 517–526.
7. Crépeau, C.; Gottesman, D.; Smith, A. Secure multi-party quantum computation. In Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing, Montreal, QC, Canada, 19–21 May 2002; pp. 643–652.
8. Broadbent, A.; Schaffner, C. Quantum cryptography beyond quantum key distribution. *Des. Codes Cryptogr.* **2016**, *78*, 351–382. [CrossRef]
9. Lutomirski, A. An online attack against Wiesner's quantum money. *arXiv* **2010**, arXiv:1010.0256.
10. Brodutch, A.; Nagaj, D.; Sattath, O.; Unruh, D. An adaptive attack on Wiesner's quantum money. *arXiv* **2014**, arXiv:1404.1507.
11. Gavinsky, D. Quantum money with classical verification. In Proceedings of the 2012 IEEE 27th Annual Conference on Computational Complexity (CCC), Porto, Portugal, 26–29 June 2012; pp. 42–52.
12. Georgiou, M.; Kerenidis, I. New constructions for quantum money. In Proceedings of the 10th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2015), Brussels, Belgium, 20–22 May 2015; Volume 44.
13. Amiri, R.; Arrazola, J.M. Quantum money with nearly optimal error tolerance. *Phys. Rev. A* **2017**, *95*, 062334. [CrossRef]
14. Gavinsky, D.; Kempe, J.; Kerenidis, I.; Raz, R.; De Wolf, R. Exponential separations for one-way quantum communication complexity, with applications to cryptography. In Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, San Diego, CA, USA, 11–13 June 2007; pp. 516–525.
15. Arrazola, J.M.; Karasamanis, M.; Lütkenhaus, N. Practical quantum retrieval games. *Phys. Rev. A* **2016**, *93*, 062311. [CrossRef]
16. Pastawski, F.; Yao, N.Y.; Jiang, L.; Lukin, M.D.; Cirac, J.I. Unforgeable noise-tolerant quantum tokens. *Proc. Natl. Acad. Sci. USA* **2012**, *109*, 16079–16082. [CrossRef]
17. Aaronson, S.; Christiano, P. Quantum money from hidden subspaces. In Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing, New York, NY, USA, 19–22 May 2012; pp. 41–60.
18. Farhi, E.; Gosset, D.; Hassidim, A.; Lutomirski, A.; Shor, P. Quantum money from knots. In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, Cambridge, MA, USA, 8–10 January 2012; pp. 276–289.
19. Moulick, S.R.; Panigrahi, P.K. Quantum cheques. *Quantum Inf. Process.* **2016**, *15*, 2475–2486. [CrossRef]
20. Radian, R.; Sattath, O. Semi-Quantum Money. *arXiv* **2019**, arXiv:1908.08889.
21. Bozzio, M.; Orieux, A.; Vidarte, L.T.; Zaquine, I.; Kerenidis, I.; Diamanti, E. Experimental investigation of practical unforgeable quantum money. *npj Quantum Inf.* **2018**, *4*, 5. [CrossRef]
22. Guan, J.Y.; Arrazola, J.M.; Amiri, R.; Zhang, W.; Li, H.; You, L.; Wang, Z.; Zhang, Q.; Pan, J.W. Experimental preparation and verification of quantum money. *Phys. Rev. A* **2018**, *97*, 032338. [CrossRef]

23. Kumar, N.; Kerenidis, I.; Diamanti, E. Experimental demonstration of quantum advantage for one-way communication complexity surpassing best-known classical protocol. *Nat. Commun.* **2019**, *10*, 1–10. [CrossRef]

24. Ben-David, S.; Sattath, O. Quantum tokens for digital signatures. *arXiv* **2016**, arXiv:1609.09047.

25. Goldreich, O. *The Foundations of Cryptography, Volume 2, Chapter Encryption Schemes*; Cambridge University Press: Cambridge, UK, 2004.

26. Upfal, E.; Mitzenmacher, M. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*; Cambridge University Press: Cambridge, UK, 2005.

27. Bar-Yossef, Z.; Jayram, T.S.; Kerenidis, I. Exponential separation of quantum and classical one-way communication complexity. In Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, 13–15 June 2004; pp. 128–137.

28. Molina, A.; Vidick, T.; Watrous, J. Optimal counterfeiting attacks and generalizations for Wiesner's quantum money. In Proceedings of the Conference on Quantum Computation, Communication, and Cryptography, Tokyo, Japan, 17–19 May 2012; pp. 45–64.

29. Croke, S.; Kent, A. Security details for bit commitment by transmitting measurement outcomes. *Phys. Rev. A* **2012**, *86*, 052309. [CrossRef]

30. Jamiołkowski, A. Linear transformations which preserve trace and positive semidefiniteness of operators. *Rep. Math. Phys.* **1972**, *3*, 275–278. [CrossRef]

31. Yao, A.C. Lower bounds by probabilistic arguments. In Proceedings of the 24th Annual Symposium on Foundations of Computer Science (sfcs 1983), Tucson, AZ, USA, 7–9 November 1983; pp. 420–428.

32. Holevo, A.S. Information-theoretical aspects of quantum measurement. *Probl. Peredachi Informatsii* **1973**, *9*, 31–42.

33. Lvovsky, A.I.; Sanders, B.C.; Tittel, W. Optical quantum memory. *Nat. Photonics* **2009**, *3*, 706. [CrossRef]

34. Julsgaard, B.; Sherson, J.; Cirac, J.I.; Fiurášek, J.; Polzik, E.S. Experimental demonstration of quantum memory for light. *Nature* **2004**, *432*, 482. [CrossRef] [PubMed]

35. Fleischhauer, M.; Lukin, M.D. Quantum memory for photons: Dark-state polaritons. *Phys. Rev. A* **2002**, *65*, 022314. [CrossRef]

36. Kozhekin, A.; Mølmer, K.; Polzik, E. Quantum memory for light. *Phys. Rev. A* **2000**, *62*, 033809. [CrossRef]

37. Arrazola, J.M.; Diamanti, E.; Kerenidis, I. Quantum superiority for verifying NP-complete problems with linear optics. *npj Quantum Inf.* **2018**, *4*, 56. [CrossRef]