Security Analysis of Quantum Lightning

Bhaskar Roberts

UC Berkeley
bhaskarr@eecs.berkeley.edu

Abstract. Quantum lightning is a new cryptographic object that gives a strong form of quantum money. Zhandry recently defined quantum lightning and proposed a construction of it based on superpositions of low-rank matrices. The scheme is unusual, so it is difficult to base the scheme's security on any widespread computational assumptions. Instead, Zhandry proposed a new hardness assumption that, if true, could be used to prove security.
In this work, we show that Zhandry's hardness assumption is in fact false, so the proof of security does not hold. However, we note that the proposal for quantum lightning has not been proven *insecure*. This work is the first step in analyzing the security of [3]'s proposal and moving toward a scheme that we can prove to be secure.

1 Introduction

A cryptographic protocol for money should satisfy two conditions:[1]

1. *Verification by untrusted users*: Any untrusted user, even an adversary seeking to counterfeit, can distinguish between valid and counterfeit banknotes.
2. *No counterfeiting*: Only the mint, a trusted administrator, can produce valid banknotes.

A classical bitstring can be easily duplicated, and will fail the *no counterfeiting* condition. However an arbitrary string of qubits cannot be duplicated, so quantum information is the first setting where *no counterfeiting* may hold. Therefore, there is interest in creating uncounterfeitable money from quantum states. This is known as public-key quantum money. However, we do not yet know how to construct public-key quantum money from widely used cryptographic assumptions, despite many attempts including [2] and [1].

More recently, [3] defined a new cryptographic object, called quantum lightning, that gives a strong form of public-key quantum money in which not even the mint can produce two copies of the same banknote. Zhandry also proposed a construction of quantum lightning, but it is unknown whether the scheme is secure. Instead, Zhandry proposed a plausible computational hardness assumption and proved that if it is true, then the scheme is secure. However, the assumption was untested.

[1] There are several variations on the quantum money problem, each with slightly different conditions. These are adapted from ones presented in [2].

Here, we show that the hardness assumption is false. Therefore the proof of security for [3]'s scheme does not hold. However, our work does not prove the scheme insecure, and it may be possible to fix the hardness assumption. Our work is the first step in determining whether [3]'s proposal is secure and whether a similar approach is viable.

The rest of the paper is organized as follows: first we summarize [3]'s proposed scheme for quantum lightning. Then we show that the hardness assumption that was used to prove security is false. Finally, we suggest where our work may lead: to a new plausible hardness assumption or to a stronger attack that proves the scheme insecure.

2 Proposed Construction of Quantum Lightning

For context, we summarize [3]'s proposed construction of quantum lightning in this section. The lightning bolt is a superposition that can be sampled efficiently, but not duplicated. Anyone can generate a random lightning bolt, and a verifier can check that the bolt was generated honestly. But it is supposedly hard to generate two states that appear to the verifier to be the same bolt.

Here is a simplified version of the construction. There is a collision-resistant hash function, $f_{\mathcal{A}}$, and the bolt is a superposition over the pre-image of some value output by $f_{\mathcal{A}}$. To generate a random bolt, we create a superposition over the domain of $f_{\mathcal{A}}$, apply $f_{\mathcal{A}}$ to the superposition, and write the output to a separate register. The output register is a superposition over the image of $f_{\mathcal{A}}$, and it is entangled with the first register. Finally, we measure the output register, which collapses to a single random eigenstate $|\mathbf{y}\rangle$, called the hash or the serial number. Since the two registers were entangled, the first register becomes a uniform superposition over the pre-image of $\mathbf{y}$. The first register's state is the bolt, and $\mathbf{y}$ is the classical serial number that identifies the bolt.

The bolt is unclonable if $f_{\mathcal{A}}$ is collision-resistant. If we can create two bolts that hash to the same serial number, then we can find a collision in $f_{\mathcal{A}}$ by simply measuring both bolts in the computational basis. Each measurement will give a random value in the pre-image of $\mathbf{y}$, and the two values are very likely to be distinct. These values represent a collision in $f_{\mathcal{A}}$, which contradicts the collision-resistance of $f_{\mathcal{A}}$.

More formally, the construction comprises three polynomial-time quantum algorithms: Setup, Gen, and Ver. Setup samples the hash function and the public verification key. This is performed by an honest administrator, called the mint. Gen generates a random bolt, and can be run by anyone, even the adversary. Finally, Ver verifies that a given state is an honestly generated bolt. Like Gen, Ver is also public-key. We describe the construction's variables, as well as the three algorithms, below.

Variables

- The scheme takes as parameters the positive integers m, q, d, e, k for which $m - d < e$ and $d < e$.
- Let $n = \binom{m+1}{2} - \binom{e+1}{2}$. n is the dimension of the image of $f_{\mathcal{A}}$.
- Let $\mathcal{D}$ be the set of $m \times m$ symmetric matrices over $\mathbb{Z}_q$ with rank $\leq d$. $\mathcal{D}$ is the domain of $f_{\mathcal{A}}$.
- Let $\mathcal{A} = \{\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_n\}$ be some subset of the symmetric $m \times m$ matrices over $\mathbb{Z}_q$. $\mathcal{A}$ determines $f_{\mathcal{A}}$.
- Let $f_{\mathcal{A}} : \mathcal{D} \to \mathbb{Z}_q^n$ such that for an input $\mathbf{M} \in \mathcal{D}$ and each $i \in [n]$,

$$[f_{\mathcal{A}}(\mathbf{M})]_i = \sum_{j=1}^{m} \sum_{k=1}^{m} (\mathbf{A}_i)_{j,k} \cdot \mathbf{M}_{j,k} = \mathrm{Tr}\big(\mathbf{A}_i^T \mathbf{M}\big) \tag{1}$$

  $f_{\mathcal{A}}$ is the hash function used to sample the bolt. It maps matrices to vectors. $[f_{\mathcal{A}}(\mathbf{M})]_i$ is the dot product of $\mathbf{A}_i$'s entries with $\mathbf{M}$'s entries. To take the dot product of two matrices, we unfurl the entries of each matrix into a vector and dot the vectors together. This procedure is captured by (1).
- Let $|\lightning\rangle$ be the lightning bolt, which is an unclonable state.

Setup

Setup samples a verification trapdoor $\mathbf{R}$ and a hash function $f_{\mathcal{A}}$. $f_{\mathcal{A}}$ is chosen so that $\mathbf{R}^T\mathbf{R}$ is in the kernel of $f_{\mathcal{A}}$, a fact that will be useful in Ver.

Setup

1. Sample $\mathbf{R} \in_R \mathbb{Z}_q^{e \times m}$. $\mathbf{R}$ is the verification trapdoor.
2. Choose $\mathcal{A}$ such that $\mathbf{R} \cdot \mathbf{A}_i \cdot \mathbf{R}^T = \mathbf{0}$, $\forall i \in [n]$, and no $\mathbf{A}_i$ is a linear combination of the others. The purpose of this step is to ensure that $\mathbf{R}^T\mathbf{R}$ is in the kernel of $f_{\mathcal{A}}$.
3. Publish $\mathbf{R}$, $\mathcal{A}$, and the parameters n, m, q, d, e, k.

Note that the space of $m \times m$ symmetric matrices $\mathbf{A}$ for which $\mathbf{R} \cdot \mathbf{A} \cdot \mathbf{R}^T = \mathbf{0}$ has dimension $\binom{m+1}{2} - \binom{e+1}{2} = n$, so $\mathcal{A}$ is a basis for this space.

Gen

Gen generates a bolt. The bolt is statistically close to a tensor product of $k+1$ mini-bolts. A mini-bolt is a uniform superposition over the pre-image of $\mathbf{y}$, and all the mini-bolts that belong to a bolt have the same $\mathbf{y}$-value.

Gen

1. Create $\left|\phi^0\right\rangle$, a uniform superposition over all sets of $k+1$ rank-d matrices in $\mathcal{D}$ that are mapped to the same $\mathbf{y}$-value. Within a set of $k+1$ matrices, all matrices must map to the same value, but the various sets can map to any value in the image of $f_{\mathcal{A}}$. [3] explains how this step is accomplished.

2. Compute $f_{\mathcal{A}}(\left|\phi^0\right\rangle)$ in superposition, and measure the function's output, $\mathbf{y}$. After the measurement, $\left|\phi^0\right\rangle$ collapses to $\left|\phi^1\right\rangle$, a superposition over all sets of $k+1$ rank-d matrices in $\mathcal{D}$ that are pre-images of $\mathbf{y}$.
3. Let $|\lightning\rangle = \left|\phi^1\right\rangle$; then output $|\lightning\rangle$ and $\mathbf{y}$.

Ver

Ver verifies a purported bolt. It takes as input a serial number $\mathbf{y}$ and a purported bolt $|P\rangle$, which comprises the purported mini-bolts, $\left|P^{(1)}\right\rangle, \ldots, \left|P^{(k+1)}\right\rangle$. Ver checks each purported mini-bolt separately, and the bolt is accepted if all mini-bolts pass and have the same serial number $\mathbf{y}$.

Ver makes two measurements to verify the mini-bolt, one in the computational basis, the other in the Fourier basis. The computational basis test checks that the eigenstates of the mini-bolt are indeed in the pre-image of $\mathbf{y}$. The Fourier basis test checks that the mini-bolt is a superposition over many eigenstates, rather than a single eigenstate. See [3] for an explanation of why the test works.

Ver

1. For each purported mini-bolt, $\left|P^{(i)}\right\rangle$, let $|\mathbf{M}\rangle$ be a generic computational-basis eigenstate of $\left|P^{(i)}\right\rangle$. Compute and measure whether: $\mathbf{M} \in \mathcal{D}$ and $f_{\mathcal{A}}(\mathbf{M}) = \mathbf{y}$.
2. Take the quantum Fourier transform of the state. Let $|\mathbf{N}\rangle$ be a generic Fourier-basis eigenstate. Measure whether $\operatorname{rank}(\mathbf{R} \cdot \mathbf{N} \cdot \mathbf{R}^T) \leq m - d$.
3. Take the inverse quantum Fourier transform, and output the resulting state. The mini-bolt passes if and only if our measurements in steps 1 and 2 passed.
4. The purported bolt passes if and only if all the mini-bolts passed relative to the same $\mathbf{y}$.

Crucially, the Fourier basis test uses the trapdoor $\mathbf{R}$ to check that the mini-bolt has the right structure, and Ver does not work without $\mathbf{R}$. However, $\mathbf{R}$ also gives information about the kernel of $f_{\mathcal{A}}$, which we will use to break the hardness assumption.

3 Analysis of the Security Proof

It is difficult to base the scheme's security on any widespread computational assumptions because superpositions of low-rank matrices are not well studied. Instead, Zhandry proposed a plausible new hardness assumption (1) and showed that if assumption 1 is true, then the proposed construction of quantum lightning is secure.

Essentially, assumption 1 says that $f_{\mathcal{A}}$ is $(2k+2)$-multi-collision-resistant (MCR) *even when we publish the trapdoor* $\mathbf{R}$.

Assumption 1 ([3]). *For some functions d, e, k in m for which $n = \binom{m+1}{2} - \binom{e+1}{2} < dm - \binom{d}{2}$, $kn \leq dm - \binom{d}{2} < (2k+1)n$, and $e > d$, $f_{\mathcal{A}}$ is $(2k+2)$-multi-collision-resistant, even if $\mathbf{R}$ is public.*

Before this work, assumption 1 was untested, but here we will show that it is false.

Breaking Assumption 1

We will show that $\mathbf{R}$ allows us to construct more than $2k + 2$ low-rank matrices that are in the pre-image of $\mathbf{y}$. $\mathbf{R}^T\mathbf{R}$ is in the kernel of $f_{\mathcal{A}}$, so we use $\mathbf{R}^T\mathbf{R}$ to construct many low-rank matrices that are in the kernel of $f_{\mathcal{A}}$. All of these matrices hash to the same value: $\mathbf{y} = \mathbf{0}$.

First, observe that $\mathbf{R}^T\mathbf{R}$ is in the kernel of $f_{\mathcal{A}}$:

$$f_{\mathcal{A}}(\mathbf{R}^T\mathbf{R})_i = \mathrm{Tr}\big(\mathbf{A}_i^T\mathbf{R}^T\mathbf{R}\big) = \mathrm{Tr}\big(\mathbf{R}\mathbf{A}_i\mathbf{R}^T\big) = 0$$

Second, we will use the rows of $\mathbf{R}$ to construct a set of low-rank matrices in the kernel of $f_{\mathcal{A}}$. Let the rows of $\mathbf{R}$ be $\{\mathbf{r}_1, \ldots, \mathbf{r}_e\} \subset \mathbb{Z}_q^m$, expressed as column vectors. For any row $\mathbf{r}_j$, $\mathbf{r}_j\mathbf{r}_j^T$ is a symmetric matrix with rank $= 1$, so $\mathbf{r}_j\mathbf{r}_j^T \in \mathcal{D}$.

For any $i \in [n]$, $\mathbf{r}_j^T \cdot \mathbf{A}_i \cdot \mathbf{r}_j = 0$. This means that

$$f_{\mathcal{A}}(\mathbf{r}_j\mathbf{r}_j^T)_i = \mathrm{Tr}\big(\mathbf{A}_i^T\mathbf{r}_j\mathbf{r}_j^T\big) = \mathrm{Tr}\big(\mathbf{r}_j^T\mathbf{A}_i\mathbf{r}_j\big) = 0$$

Therefore, $\mathbf{r}_j\mathbf{r}_j^T$ is in the kernel of $f_{\mathcal{A}}$.

Third, let $K = \{\mathbf{r}_1\mathbf{r}_1^T, \ldots, \mathbf{r}_e\mathbf{r}_e^T\}$ be the e matrices that we constructed. Then take any linear combination of d of the matrices in K. The resulting matrix is also a symmetric matrix of rank $\leq d$ that maps to $\mathbf{0}$. This procedure can be easily modified to produce matrices in the pre-image of another output value.

Lastly, this procedure produces many more than $2k+2$ colliding inputs. Due to the restrictions on m, n, d, e, k, it is the case that $k < d < e$. It suffices to find $4e$ colliding inputs because $2k + 2 < 2e + 2 < 4e$. Since $\mathbf{R}$ is random, with overwhelming probability, $\mathbf{R}$ has rank e. Then the matrices in K are linearly independent, and the number of matrices we can construct from this procedure is on the order of $\binom{e}{d}q^d$ matrices, which is much more than $4e$.

In summary, we've given a procedure that uses $\mathbf{R}$ to construct many ($\geq 2k + 2$) inputs to $f_{\mathcal{A}}$ that map to $\mathbf{0}$. Therefore assumption 1 is false.

Implications and Future Work

The proof of security given in [3] was based on assumption 1, and since assumption 1 is false, the proof of security does not hold.

However we are optimistic that the construction can be patched (modified) to rule out the attack on assumption 1 that we presented, and any similar attacks. We would need to find an $\mathbf{R}$ that is useful for verification but that does not give a matrix in the kernel of $f_{\mathcal{A}}$. Patching the construction is an open problem.

Additionally, we wonder whether [3]'s existing construction can be proven insecure with an attack similar to the one presented in this paper. After all, a similar attempt at constructing quantum lightning can be proven insecure with a similar attack. [3]'s scheme is similar to an attempted folklore construction of quantum lightning based on the SIS problem ([3], section 1.1). Where [3]'s construction uses matrices of low rank, the SIS-based construction uses vectors of small norm. The SIS-based construction is insecure because the verification trapdoor can be used to construct a superposition over short vectors in the kernel of the hash function, and this state passes verification. Analogously, we hypothesize that $\mathbf{R}$ could be used to create a superposition of low-rank matrices in the kernel of $f_{\mathcal{A}}$ that passes verification.

4 Acknowledgements

I thank Mark Zhandry for useful discussions.

References

1. Aaronson, S., Christiano, P.: Quantum money from hidden subspaces. Theory of Computing **9**(9), 349–401 (2013). https://doi.org/10.4086/toc.2013.v009a009, http://www.theoryofcomputing.org/articles/v009a009
2. 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. pp. 276–289. ITCS '12, ACM, New York, NY, USA (2012). https://doi.org/10.1145/2090236.2090260, http://doi.acm.org/10.1145/2090236.2090260
3. Zhandry, M.: Quantum lightning never strikes the same state twice **11478**, 408–438 (2019). https://doi.org/10.1007/978-3-030-17659-$4_1$4