1 Definitions

Non-collapsing hash function. A hash function that is collision-resistant and not infinitely-often collapsing. (See [Zha17] for definitions of 'collision-resistant' and 'infinitely-often collapsing'. We can assume that there are adversaries who win the collapsing game for such functions with probability near 1. See [GYZ17], [Zha17] for examples of how one can boost the success probability of any bad adversary who breaks infinitely-often collapsing security so that it becomes a very good adversary.)

Part-probabilistic non-collapsing hash function. A function $F(k, x, r)$ whose output takes the form $H(k, x) \mid\mid R(H(k, x), r)$, where H is a (deterministic) collision-resistant hash function, and R is another function whose output is permitted to depend upon the randomness r and the output of H. We require the following security properties of F:

1. Collision resistance for H: For any quantum polynomial time adversary A,

$$\Pr[H(k, x_0) = H(k, x_1) \wedge x_0 \neq x_1 \ : \ (x_0, x_1) \leftarrow A(k), k \leftarrow \{0,1\}^\lambda] < \mathsf{negl}(\lambda)$$

2. (Infinitely-often) non-collapsing: There exists an adversary A (consisting of two phases, A_0 and A_1) who can win the following game with probability $1 - \gamma$, where γ is negligible.
   - The challenger has an input bit b.
   - The challenger chooses a random key k, which it gives to A_0.
   - A_0 creates a superposition $|\psi\rangle = \sum_x \alpha_x |x\rangle$ and submits this state to the challenger.
   - The challenger generates a random r. It evaluates $F(k, \cdot, r)$ in superposition on $|\psi\rangle$, to get the state $\sum_x \alpha_x |x, H(k, x), R(H(k, x), r)\rangle$.
   - The challenger does one of the following:
     - If $b = 0$, it measures the last two registers, and returns the state $\sum_{x:H(k,x)=y} \alpha_x |x, y, R(y, r)\rangle$ to A.
     - If $b = 1$, it measures the entire state, and returns the state $|x_0, y, R(y, r)\rangle$ (for some x_0) to A.
   - A_1 outputs a guess for b. If A_1 is correct, A wins the game.

Note that, under this definition of 'non-collapsing' (which mimics [Zha17]'s definition), the distinguisher A_1 is only guaranteed to exist *if the challenger behaves honestly.* A_1's success may depend upon r being honestly generated, and upon R being honestly run; it has no way of verifying that either is the case. In the generic hash function setting, we cannot guarantee that R will be run honestly on a random r, and we cannot guarantee that the distinguisher A_1 which wins the game above will still be useful if this is not the case.

It is evident that the (deterministic) non-collapsing hash function is a special case of the part-probabilistic non-collapsing hash function, so that any NCH function is also a PP-NCH function.

Chosen-y-secure hash function. A hash function $H(k, x)$ for which no quantum polynomial time adversary can win the following game with more than negligible probability:

- The challenger chooses a random key k, which it gives to A.
- The challenger creates a uniform superposition over all inputs x in the input space of H, and evaluates $H(k, \cdot)$ upon this superposition to obtain the state $\sum_x \alpha_x |x, H(k, x)\rangle$. It then measures the output register to obtain a state $|\psi_y\rangle = \sum_{x:H(k,x)=y} \alpha_x |x, y\rangle$ for some random y.
- The challenger gives $|\psi_y\rangle$ to the adversary. The adversary wins the game if it can recover x_0, x_1 such that $H(k, x_0) = H(k, x_1) = y$.

The collapsing security game can be defined in the same way for CYS hash functions that it is for collision-resistant hash functions.

Note that it is easy to construct a collision-resistance adversary from a chosen-y adversary, and that, therefore, any collision-resistant hash function is also a chosen-y-secure hash function. This notion of 'chosen y' security is close to that of second preimage resistance security; the former can be considered a strengthening of the latter to suit the quantum setting.

Note, in addition, that any PP-NCH function can be transformed into a CYS-NCH function. Chosen-y security follows directly from the assumption of collision resistance for the deterministic part of the PP-NCH function. The non-collapsing property, meanwhile, follows from the fact that the chosen-y setting already demands that y is chosen randomly (presumably by some trusted party) if the CYS function's preimage security is to hold. Any construction using a chosen-y hash function for its preimage security properties, therefore, must generate honest randomness when it generates y; and, as such, y can be used as a source of randomness for the randomised part of the PP-NCH function.

2 NCH implies BZ-GYZ

Claim. Any non-collapsing hash function can be used to build a one-time signature scheme that is Boneh-Zhandry secure but not Garg-Yuen-Zhandry secure.

Scheme. Given a non-collapsing hash function F, and an arbitrary BZ-secure one-time signature scheme $(\mathsf{Gen}_{\mathsf{BZ}}, \mathsf{Sign}_{\mathsf{BZ}}, \mathsf{Ver}_{\mathsf{BZ}})$ (these are easy to produce; for example, the standard Lamport construction of a one-time signature scheme from a collision-resistant hash function is BZ-secure, according to [BZ13]), we construct a BZ-GYZ scheme $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ as follows.

- Gen simply runs $\mathsf{Gen}_{\mathsf{BZ}}$ to generate a pair of keys $(\mathsf{pk}, \mathsf{sk})$ for the BZ signature scheme.
- $\mathsf{Sign}(\mathsf{sk}, m) = F(m) \parallel \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, F(m)) = \sigma$. In other words, Sign applies the non-collapsing hash function F to the message, signs the hashed message using the BZ scheme, and outputs the hashed message concatenated with the signature it obtains from the BZ signing oracle.
- $\mathsf{Ver}(\mathsf{pk}, m, \sigma)$ firstly hashes the message m to obtain $F(m)$, and then verifies $F(m)$ using $\mathsf{Ver}_{\mathsf{BZ}}$ and pk.

Proof. We firstly prove that this scheme is BZ-secure, assuming that $(\mathsf{Gen}_{\mathsf{BZ}}, \mathsf{Sign}_{\mathsf{BZ}}, \mathsf{Ver}_{\mathsf{BZ}})$ is BZ-secure.

Suppose we have some adversary A who is able to break the BZ-security of $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$. The adversary B can then use A to break the BZ-security of $(\mathsf{Gen}_{\mathsf{BZ}}, \mathsf{Sign}_{\mathsf{BZ}}, \mathsf{Ver}_{\mathsf{BZ}})$ as follows:

- B receives pk from its challenger. It passes pk on to A.
- A creates a superposition of messages $\sum_m \alpha_m |m\rangle$ and gives it to B as a query. B computes F on it in superposition, and then passes $\sum_m \alpha_m |m, F(m)\rangle$ on to its challenger, who computes $\mathsf{Sign}_{\mathsf{BZ}}$ on it and returns the state $\sum_m \alpha_m |\ m, F(m), \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, F(m))\ \rangle$ to B. B gives this state to A.
- A outputs its (classical) forgery for $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$. This forgery will take the form $((m_0, F(m_0), \sigma_0), (m_1, F(m_1), \sigma_1))$, where m_0 and m_1 are two distinct messages.
- If $F(m_0) = F(m_1)$, then we have found a collision for F, which ought to be impossible, because we assume that F, a non-collapsing hash function, is collision-resistant. If $F(m_0) \neq F(m_1)$, then B outputs $((F(m_0), \sigma_0), (F(m_1), \sigma_1))$ as its forgery for the $(\mathsf{Gen}_{\mathsf{BZ}}, \mathsf{Sign}_{\mathsf{BZ}}, \mathsf{Ver}_{\mathsf{BZ}})$ scheme.
- If A's success probability is non-negligible, then so is B's.

Therefore, $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ is BZ-secure if $(\mathsf{Gen}_{\mathsf{BZ}}, \mathsf{Sign}_{\mathsf{BZ}}, \mathsf{Ver}_{\mathsf{BZ}})$ is BZ-secure.

We now prove that the proposed scheme is not GYZ-secure. To do this, we use the fact that F is non-collapsing. Let D be an adversary which can break collapsing security for F. Following the proof to Theorem 13 in [GYZ17], we construct an adversary A who uses D to break the GYZ-security of $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$. A acts as follows:

- A queries D to get a superposition of 'messages' (preimages) $\rho = \sum_{m,m'} \alpha_m \alpha^*_{m'} |m\rangle\langle m'|$, and places this superposition ρ in its message register. A then flips a coin with outputs in $\{0,1\}$ and measures the message register iff the coin gives 0. This results in the following state:

$$\frac{1}{2}\left[\, |0\rangle\langle 0| \otimes \sum_m |\alpha_m|^2 |m\rangle\langle m| + |1\rangle\langle 1| \otimes \rho \,\right] \tag{1}$$

- A sends this state to the GYZ signing oracle, which signs it and places the signature in a newly created pair of signature registers. The result is then

$$\begin{aligned}\tfrac{1}{2}\Big[\quad & \textstyle\sum_m |\alpha_m|^2 \quad |0\rangle\langle 0| \otimes |m\rangle\langle m| \otimes |F(m)\rangle\langle F(m)| \otimes |\,\mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, F(m))\,\rangle\langle\, \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, F(m))\,| \\ & + \textstyle\sum_{m,m'} \alpha_m \alpha^*_{m'} \quad |1\rangle\langle 1| \otimes |m\rangle\langle m'| \otimes |F(m)\rangle\langle F(m')| \otimes |\,\mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, F(m))\,\rangle\langle\, \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, F(m'))\,| \quad \Big]\end{aligned} \tag{2}$$

  A measures the '$F(m)$' register (the third register from the left), to obtain the state

$$\begin{aligned}\tfrac{1}{2}\textstyle\sum_y \beta_y \Big[\quad & \textstyle\sum_{m:F(m)=y} |\alpha_m|^2 \quad |0\rangle\langle 0| \otimes |m\rangle\langle m| \otimes |y\rangle\langle y| \otimes |\,\mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,\rangle\langle\, \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,| \\ & + \quad |1\rangle\langle 1| \otimes |\psi_y\rangle\langle \psi_y| \otimes |y\rangle\langle y| \otimes |\,\mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,\rangle\langle\, \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,| \quad \Big]\end{aligned} \tag{3}$$

  where $|\psi_y\rangle = \sum_{m:F(m)=y} \alpha_m |m\rangle$.

- A applies D to the second and third registers from the left and saves D's output in a newly created ancilla register. If D is a very good distinguisher which gives the right answer with probability $1-\gamma$, then, by the gentle measurement lemma, the resulting state is $4\sqrt{2\gamma}$ close to

$$\begin{aligned}\tfrac{1}{2}\textstyle\sum_y \beta_y \Big[\quad & \textstyle\sum_{m:F(m)=y} |\alpha_m|^2 \quad |0\rangle\langle 0| \otimes |m\rangle\langle m| \otimes |y\rangle\langle y| \otimes |\,\mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,\rangle\langle\, \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,| \otimes |0\rangle\langle 0| \\ & + \quad |1\rangle\langle 1| \otimes |\psi_y\rangle\langle \psi_y| \otimes |y\rangle\langle y| \otimes |\,\mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,\rangle\langle\, \mathsf{Sign}_{\mathsf{BZ}}(\mathsf{sk}, y)\,| \otimes |1\rangle\langle 1| \quad \Big]\end{aligned} \tag{4}$$

  (It is possible to boost the success probability of any bad distinguisher so that it becomes a good distinguisher, as long as the bad distinguisher still breaks collapsing security for F. See Section 4.1 of [GYZ17].)

- A sends this state to its challenger for verification. Because all the signatures in this state were legally obtained, this state passes verification with probability 1, and is not perturbed by verification. Therefore, the challenger's output `GYZ-Exp`(A) is equal to the state above.

- By the same reasoning that is used in the proof of Theorem 13 in [GYZ17], there is no basis-respecting adversary that could produce such an output, as basis-respecting adversaries must commute with measurement in the computational basis. (Intuitively, it is clear that the distinguisher does not commute with measurement, because its very purpose is to determine whether or not a state has been measured.)

Therefore, if F is non-collapsing, $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ is not GYZ-secure.

3 NCH implies quantum tokens

Claim. Any non-collapsing hash function can be used to build a testable tokenised signature scheme.

Scheme. Given an arbitrary non-collapsing hash function F, we construct a one-bit, one-time, testable tokenised signature scheme. (By Section 5 of [BS16], a one-bit, one-time tokenised signature scheme can be extended to a fully-fledged tokenised signature scheme via a series of reductions.)

Because F is non-collapsing, there exists an adversary A that can break collapsing security for F with very high success probability $1-\gamma$. We use A, along with an arbitrary classical digital signature scheme $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$, to construct a tokens scheme.

- Let $(\mathsf{pk}, \mathsf{sk})$ denote the public and secret keys for the classical signature scheme, and let (pk, sk) stand for the keys to our tokens scheme.

- A consists of two phases: 1) the phase which outputs a superposition of messages for its challenger to hash, and 2) the phase which guesses whether the challenger measured its entire state or only the output registers. Following [Zha17], we let A_0 denote the first phase and A_1 the second.

- $\mathsf{key\text{-}gen}$ runs Gen to generate $(\mathsf{pk}, \mathsf{sk})$ for the digital signature scheme. It then outputs $sk = (\mathsf{sk}, A_0)$ and $pk = (\mathsf{pk}, A_1)$.

- $\mathsf{token\text{-}gen}(sk)$ firstly runs A_0, twice, to generate two superpositions of messages $\sum_{m_0} \alpha_{m_0} |m_0\rangle$ and $\sum_{m_1} \alpha_{m_1} |m_1\rangle$. It computes F in superposition on both, and measures both output registers. The result is two states $|\psi_{y_0}\rangle, |\psi_{y_1}\rangle$, where $|\psi_{y_b}\rangle = \sum_{m:F(m)=y_b} |m, F(m)\rangle$. It then signs the tuple (y_0, y_1) using sk and Sign. $\mathsf{token\text{-}gen}$ outputs $\big(|\psi_{y_0}\rangle, |\psi_{y_1}\rangle, \mathsf{Sign}(\mathsf{sk}, (y_0, y_1))\big)$ as $|\text{♙}\rangle$.

- $\mathsf{sign}(\alpha \in \{0,1\}, |\text{♙}\rangle)$ measures the input register of $|\psi_{y_\alpha}\rangle$, and outputs, as a signature for α, a preimage m_α for y_α under F, along with an unaltered $|\psi_{y_{1-\alpha}}\rangle$ and the signature $\mathsf{Sign}(\mathsf{sk}, (y_0, y_1))$.

- $\mathsf{verify}\Big(pk, \alpha, \sigma = \big(m_\alpha, |\psi_{y_{1-\alpha}}\rangle, \mathsf{Sign}(\mathsf{sk}, (y_0, y_1))\big)\Big)$ firstly verifies the signature on (y_0, y_1) using pk and Ver. Following this, it checks that m_α hashes to y_α, and that the superposition of messages in the message register of $|\psi_{y_{1-\alpha}}\rangle$ hashes to $y_{1-\alpha}$. It then checks, using A_1, that the purported $|\psi_{y_{1-\alpha}}\rangle$ really does still have an unmeasured message register. In more precise terms, if A_1 outputs 'measured', verify outputs F; otherwise, verify outputs T.

- $\mathsf{verify\text{-}token}(|\text{♙}\rangle)$ applies Ver_0, from Section 4.2 of [Zha17], to both candidate states $|\psi_{y_0}\rangle$ and $|\psi_{y_1}\rangle$.

Proof.

Testability. Testability follows directly from Zhandry's work in Section 4.2 of [Zha17]. The correctness portion of testability is identical to the correctness requirement for quantum money, which Zhandry's construction satisfies. The security portion of testability also follows from Section 4.2. By Zhandry's proof of security, we can assume that two dishonest candidate states $|\phi_0\rangle$ and $|\phi_1\rangle$ pass $\mathsf{verify\text{-}token}$ with at most negligible probability. Therefore, except with negligible probability, any two states passing $\mathsf{verify\text{-}token}$ will be honest states. Because verify always accepts honest tokens, our scheme satisfies equation (8) of [BS16].

Unforgeability. Suppose that there is an adversary who can, after seeing a single token $|\text{♙}\rangle = (|\psi_{y_0}\rangle, |\psi_{y_1}\rangle)$, produce two signatures $(x_0, |\phi_0\rangle)$ and $(x_1, |\phi_1\rangle)$ that both pass verify with non-negligible probability. (Assume, for the present, that the adversary did not attempt to forge a signature on (y_0, y_1).)

Without loss of generality, consider the $(x_0, |\phi_0\rangle)$ tuple. To pass the hash tests which verify executes, x_0 must be a valid preimage to y_0 under F. Note that the only states $|\phi_0\rangle$ which the adversary can hold without violating the collision resistance of F, given that he already has x_0, are states negligibly close to $|x_0, y_0\rangle$. However, these states will almost invariably fail A_1's distinguishing test, because A_1 is—under the assumption that we have already boosted its success probability to some $1-\gamma$—able to tell, with probability $1-\gamma$, the difference between any $|x_0, y_0\rangle$ and the state $|\psi_{y_0}\rangle$. The adversary's only alternative is to forge a signature on (y_0, y_1), but his succeeding would violate the security of $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$, which we assume is impossible. Therefore, there is no adversary which can produce two signatures that pass verify, having seen only one token—except with negligible probability.

Remarks.

- The part of this scheme which is applicable to public-key quantum money is no different from Zhandry's construction of [Zha17]. We have essentially produced an extension of that construction which happens to be a tokens scheme.

- One curious property of this NCH-based construction for tokens is that the signatures σ it produces are quantum, while [BS16]'s signatures were classical. In consequence, our construction loses some of the properties which [BS16] considered desirable, such as the ability to convert quantum money into 'classical cheques' which could be sent over classical channels. The sacrifice is not a fruitless one, however: since our scheme is based on non-collapsing hash functions, we also acquire the properties of 'collision-free quantum money' which [Zha17] considered desirable, such as the ability to ensure that even the bank cannot forge quantum money (or signing tokens).

4 NCH from iO

4.1 Definition of NCH function

Let q be a positive even integer. Let L be a map from $\mathbb{F}_2$ to $\mathbb{Z}_q$ such that

$$L(b) = \begin{cases} 0 & b = 0 \\ \frac{q}{2} & b = 1 \end{cases}. \tag{5}$$

When we write $L(x)$, where x is a vector with entries in $\mathbb{F}_2^n$, or $L(M)$, where M is a matrix with entries in $\mathbb{F}_2^n$, we mean the entry-wise application of L to x or to M. Note that L, in all its forms, is injective.

The non-collapsing hash function is defined as follows:

- $\mathsf{key\text{-}gen}$: Generates the following constants which define the hash function.
  - A randomly-chosen $m \times n$ LWE matrix, A, whose entries are in $\mathbb{Z}_q$;
  - k n-bit binary strings $s_1, \cdots, s_k$, which represent k secrets.

  The hash function key that $\mathsf{key\text{-}gen}$ outputs consists of the following:

  - The matrix A.
  - k LWE samples $A\big(L(s_i)\big) + e_i$, for i in $1, \cdots, k$. e_i are randomly chosen error vectors in $\mathbb{Z}_q^m$ with sufficiently small bounded magnitude.
  - A program P. Let $\mathcal{S}$ denote the $k \times n$ matrix over $\mathbb{F}_2$ whose rows are $s_1, \cdots, s_k$. $P(\cdot)$ is the obfuscation under shO of the subspace membership oracle for the subspace $\ker(I_k \mid \mathcal{S})$, where I_k is the $k \times k$ identity matrix, and $I_k \mid \mathcal{S}$ is the $k \times (n+k)$ block matrix obtained by joining I_k and $\mathcal{S}$ together, with I_k on the left and $\mathcal{S}$ on the right. For convenience, we denote this subspace $(\ker(I_k \mid \mathcal{S}))$ by S_0.
- Evaluation: Let $s_b = \bigoplus_i b_i s_i$, where b_i denotes the ith bit of b. Similarly, let $e_b = \sum_i b_i e_i$. (The sum in the latter uses addition modulo q.) $F(b, x, e) = A\big(L(x)\big) + e + A\big(L(s_b)\big) + e_b$. F can be evaluated publicly using the hash function key.

4.2 Non-collapsing

P implements a subspace membership program which takes in d $(= d_1 || d_2)$ as argument and checks whether the following matrix equation holds:

$$\left[\begin{array}{c|c} \underbrace{\quad I_k \quad}_{k \text{ bits}} & \underbrace{\begin{matrix} s_1 \\ \vdots \\ s_k \end{matrix}}_{n \text{ bits}} \end{array}\right] \begin{bmatrix} d_1 \\ \hline d_2 \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \end{matrix}\right\} k \text{ bits} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} n \text{ bits} \end{matrix} = 0.$$

Note that this is equivalent to checking that, for all i, $d_2 \cdot s_i = d_{1,i}$ (where $d_{1,i}$ represents the ith bit of d_1).

We now prove that F is non-collapsing, viz., there is an adversary A (consisting of two phases, A_0 and A_1) who wins the collapsing game with non-negligible probability. When A_0 is called upon to provide a superposition over the message registers, it prepares the superposition

$$q^{-\frac{n}{2}} \sum_{\substack{b \in \{0,1\}^k \\ x \in \{0,1\}^n \\ e \in \mathbb{Z}_q^m}} \sqrt{D_{\mathbb{Z}_q^m, B_P}(e)} \, |b, x, e\rangle \tag{6}$$

(This superposition can be produced efficiently by creating the superposition $q^{-\frac{n}{2}} \sum_{e \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m, B_P}(e)} \, |e\rangle$—which is possible by Lemma 3.12 of [Reg05]—and adding uniform superpositions over all b and all x.)

The challenger evaluates F on this superposition, and obtains the state

$$q^{-\frac{n}{2}} \sum_{\substack{b \in \{0,1\}^k \\ x \in \{0,1\}^n \\ e \in \mathbb{Z}_q^m}} \sqrt{D_{\mathbb{Z}_q^m, B_P}(e)} \, |b, x, e\rangle \, \big|A\big(L(x)\big) + e + A\big(L(s_b)\big) + e_b\big\rangle \,. \tag{7}$$

The challenger then measures either the entire state or else the output register only, and returns one of the two following states to A_1:

- $|\psi_{\text{meas}}\rangle = |b, x, e\rangle \, |F(b, x, e)\rangle$ for some (b, x, e).
- $|\psi_y\rangle$ for some y. We claim that $|\psi_y\rangle$ has (up to normalisation) the specific form

$$|\psi_y\rangle = \sum_b \sqrt{D_{\mathbb{Z}_q^m, B_P}(e^* - e_b)} \, |b, x^* \oplus s_b, e^* - e_b\rangle \, |y\rangle \,. \tag{8}$$

To show that it has this form, we prove the following lemma.

Lemma. Fix a y. For each $b \in \{0,1\}^k$, there is exactly one (x', e') with $x' \in \mathbb{F}_2^n, e' \in \mathbb{Z}_q^m$ such that $F(b, x', e') = y$, and it has the form $(x' = x^* \oplus s_b, e' = e^* - e_b)$, where $s_b = \bigoplus_i b_i s_i$, $e_b = \sum_i b_i e_i$, and (x^*, e^*) is the unique solution to the equation $A\big(L(x^*)\big) + e^* = y$.

Proof. Suppose that a y has been measured. Note that this y must be in the form $A\big(L(x^*)\big) + e^*$ for some $x^* \in \mathbb{F}_2^n$, and some $e^* \in \mathbb{Z}_q^m$ with sufficiently small magnitude. We recall that L is injective, apply the inversion theorem (Theorem 5.1 of [MP11]), and conclude that (x^*, e^*) is unique. Given that this is so, the task of finding collisions becomes that of finding solutions (b, x', e') to the following equation:

$$A\big(L(x^*)\big) + e^* = A\big(L(x')\big) + e' + A\big(L(s_b)\big) + e_b. \tag{9}$$

The equation can be rearranged to yield:

$$A\big(L(x')\big) + e' = A\big(L(x^*)\big) + e^* - \Big(A\big(L(s_b)\big) + e_b\Big). \tag{10}$$

$$\implies A\big(L(x')\big) + e' = A\big(L(x^* \oplus s_b)\big) + e^* - e_b. \tag{11}$$

Applying the inversion theorem once again, we conclude that

$$L(x') = L(x^* \oplus s_b) \tag{12}$$

$$\implies x' = x^* \oplus s_b; \tag{13}$$

$$e' = e^* - e_b. \tag{14}$$

For each $b \in \{0,1\}^k$, there will be exactly one (x', e') such that the pair of equations above are satisfied, and x' and e' have the form we claimed they would have. The lemma is thus proved.

Given the lemma, it is clear that (up to normalisation)

$$|\psi_y\rangle = \sum_b \sqrt{D_{\mathbb{Z}_q^m, B_P}(e^* - e_b)}\, |b, x^* \oplus s_b, e^* - e_b\rangle\, |y\rangle\,. \tag{15}$$

If we require that the maximum amplitude of e_b is small compared to the Gaussian parameter of the distribution D, $\sqrt{D_{\mathbb{Z}_q^m, B_P}(e^* - e_b)}$ should be approximately the same for all b. More precisely, Lemma 34 in [BCM+18] guarantees that, if we choose e_i so that $||e_b|| \leq \sqrt{m}B_V$ for all e_b—and B_V is a bound such that the ratio $\frac{B_P}{B_V}$ is superpolynomial in n—we have

$$H^2\big(D_{\mathbb{Z}_q^m, B_P}(e^*), D_{\mathbb{Z}_q^m, B_P}(e^* - e_b)\big) \leq 1 - e^{-\frac{2\pi m B_V}{B_P}}\,, \tag{16}$$

where H^2 is the Hellinger distance. Note that the right-hand-side is negligible in n. It follows that the trace distance between the state $|\psi_y\rangle$ and the state

$$\sum_b \sqrt{D_{\mathbb{Z}_q^m, B_P}(e^*)}\, |b, x^* \oplus s_b, e^* - e_b\rangle\, |y\rangle\,. \tag{17}$$

is negligible in n. In other words,

$$|\psi_y\rangle \approx \alpha \sum_b |b, x^* \oplus s_b, e^* - e_b\rangle\, |y\rangle\,. \tag{18}$$

In the following analysis, therefore, we assume $|\psi_y\rangle$ is of the above form.

In order to tell whether it has been given $|\psi_{\text{meas}}\rangle$ or $|\psi_y\rangle$, A_1 performs the following procedure.

- Uncompute the e register. This is possible because, using the hash function key, the value of the e register for any member of the superposition can be computed from its corresponding values of y, x and b.
- Apply the Hadamard transform to the remaining registers. In the case where A_1 was given $|\psi_y\rangle$, this results (up to normalisation) in the following state:

$$\sum_d \left(\sum_b (-1)^{d\cdot(b\,||\,x^*\oplus s_b)}\right)|d\rangle = \sum_d \left(\sum_b (-1)^{d_1\cdot b}(-1)^{d_2\cdot(x^*\oplus s_b)}\right)|d\rangle \tag{19}$$

Not all $d \in \{0,1\}^{n+k}$ will be supported in this superposition (and we will make precise which ones are). In the case where A_1 was given $|\psi_{\text{meas}}\rangle$, however, every d will appear in the superposition, because there will be no opportunity for the amplitude of d to cancel.

Note that, in the $|\psi_y\rangle$ case, we can rewrite the amplitude of d in the following way:

$$\begin{aligned}
\sum_b (-1)^{d\cdot(b\,||\,x^*\oplus s_b)} &= \sum_b (-1)^{d_1\cdot b}(-1)^{d_2\cdot(x^*\oplus s_b)} && (20)\\
&= (-1)^{d_2\cdot x^*}\sum_b (-1)^{d_1\cdot b}(-1)^{d_2\cdot\left((x^*\oplus s_b)\oplus x^*\right)} && (21)\\
&= (-1)^{d_2\cdot x^*}\sum_b (-1)^{d_1\cdot b}(-1)^{d_2\cdot s_b} && (22)\\
&= (-1)^{d_2\cdot x^*}\sum_b (-1)^{\sum_i b_i\cdot(d_{1,i}\oplus d_2\cdot s_i)} && (23)\\
&= (-1)^{d_2\cdot x^*}\prod_{i=1}^{n}\left(1 + (-1)^{d_{1,i}\oplus d_{2,i}\cdot s_i}\right) && (24)
\end{aligned}$$

(22) is *nonzero* if and only if $d_2 \cdot s_i = d_{1,i}$ for all i. Therefore, the only ds that will be supported in the superposition after Hadamard are those such that $P(d) = 1$.

- Apply P to the superposition of ds and measure the result. If A_1 was given $|\psi_y\rangle$, this measurement will yield 1 with probability 1. If A_0 was given $|\psi_{\text{meas}}\rangle$, the probability that this measurement yields 1 is negligible, because [the state $|\psi_{\text{meas}}\rangle$ assumes after Hadamard is negligibly close to the uniform distribution, and] the probability $Pr\Big[d \in S_0 \,\Big|\, d \xleftarrow{r} \mathbb{F}_2^{n+k}\Big]$ is exponentially small if k is linear in n. This completes the proof that A wins the collapsing game with non-negligible probability $1 - \mathsf{negl}(n)$.

4.3 Collision resistance

Note that, if an adversary can recover $\big((b, x, e), (b', x', e')\big)$ such that $F(b, x, e) = F(b', x', e')$, he can evaluate $x \oplus x'$ and recover $s_{\hat{b}}$ for some $\hat{b}$. Therefore, in order to prove that it is impossible to find collisions, we prove that it is impossible to recover any $s_{\hat{b}} = \bigoplus_i \hat{b}_i s_i$ given the hash function key. We do so through a sequence of hybrids.

- H_0: The challenger outputs the key as it is defined in section 4.1.
- H_1: Relying upon the security of shO, the challenger swaps P for another subspace membership program that instead checks membership in a random, higher-dimensional subspace $S_1 \supset S_0$. H_0 and H_1 are indistinguishable by the security of shO.
- H_2: The challenger gives out a full description of S_1, instead of a membership program. Any adversary who can break collision resistance in H_1 can also do so in H_2.
- H_3: The challenger outputs k random strings instead of the samples $A\big(L(s_i)\big) + e_i$. It also outputs S_1 and A as before.

  We show that this hybrid and the last are indistinguishable by reducing the problem of telling them apart to the problem of decisional LWE with secrets drawn from any distribution $\mathcal{D}$ that has sufficient entropy ([GKPV10]). Following [GKPV10], we denote this problem by $\mathsf{DLWE}(\mathcal{D})$.

  Let C be the following probabilistic algorithm:

  1. Choose a uniformly random $k \times (n + k)$ matrix M whose kernel is in S_1. Put M in reduced row echelon form. Denote the row-reduced version of M by $\hat{M}$.
  2. If the first k columns of $\hat{M}$ are the k-dimensional identity matrix, continue. Otherwise, repeat step 1.
  3. If $\hat{M}$ made it past step 2, then it has the form $I_k \,|\, \mathcal{S}'$ for some $k \times n$ matrix $\mathcal{S}'$. Output $\mathcal{S}'$.

  Note that C samples uniformly at random from the matrices M satisfying the following two criteria:

  1. The kernel of M is in S_1;
  2. The first k columns of M are linearly independent.

  For each $\hat{M} = I_k \,|\, \mathcal{S}'$, the maximum number of matrices M (in C's set of potential samples) which row reduce to $\hat{M}$ is given by

  $$(2^k - 1) \cdots (2^k - 2^{k-1}) \leq 2^{k^2}. \tag{25}$$

  The left-hand-side is an expression for the number of ways one can choose a basis for a k-dimensional space over $\mathbb{F}_2$. Since matrices which have the same RREF always have the same kernel, their rows can be interpreted as different choices of basis for the perp space of that kernel.

  We prove the following lemma in order to show that the output of C has sufficient min-entropy to serve as the distribution $\mathcal{D}$ in the $\mathsf{DLWE}(\mathcal{D})$ problem.

  Lemma 1. *Let S_1 be $(n + k - \ell)$ dimensional, where $0 < \ell < k$. Under certain conditions on n, k and ℓ, there are at least exponentially many matrices M which the first step of C could sample.*

Note that, in step 1, C effectively samples at random from the uniform distribution over all dimension-n subspaces contained in S_1. (There is a bijection between these subspaces and the $\hat{M}$s, because a single matrix cannot have two kernels, and two matrices in reduced row echelon form have the same kernel if and only if they are the same matrix.) We argue that the effect of the second step is to eliminate about $\frac{3}{4}$ of these n-dimensional subspaces from consideration, leaving behind a uniform distribution over those subspaces which remain.

By the $\mathsf{DLWE}(\mathcal{D})$ assumption, where $\mathcal{D}$ is the uniform distribution over the set $\{L(\mathcal{S}'^T) \,|\, \mathcal{S}' \leftarrow C(S_1)\}$, no adversary can tell the difference between the samples $A\big(L(s_i)\big) + e_i$ and k random strings. Suppose there were an adversary A who could. We enlist A to build an adversary B who breaks $\mathsf{DLWE}(\mathcal{D})$. B receives a random LWE sample $\Big(A, A\big(L(\mathcal{S}'^T)\big) + E\Big)$, where A is an $m \times n$ matrix, $L(\mathcal{S}'^T)$ is an $n \times k$ matrix, and E is an $m \times k$ matrix; B also knows the distribution $\mathcal{D}$, which means that it knows S_1 and can provide A with a description of it. Notice that the columns of B's sample constitute k vectors of the form $A\big(L(s_i)\big) + e_i$. Therefore, B gives its sample to A column-wise, and outputs whatever A outputs. If A can distinguish between H_3 and H_2, then B can break $\mathsf{DLWE}(\mathcal{D})$. We conclude that H_3 and H_2 are indistinguishable.

- H_4: The challenger simply outputs S_1. It is impossible to recover span$(s_1, \cdots, s_k)$—or any member of that subspace—with non-negligible probability from only S_1, provided that the dimension gap $d_1 - d_0$ (where d_1 is the dimension of S_1, and d_0 is the dimension of S_0) is linear in n: there are exponentially many equally probable possibilities, and the adversary has no way of distinguishing between them. This completes our proof that F is collision resistant.

References

[BCM+18] Z. Brakerski, P. Christiano, U. Mahadev, U. Vazirani, and T. Vidick. Certifiable Randomness from a Single Quantum Device. *ArXiv e-prints*, April 2018.

[BS16] S. Ben-David and O. Sattath. Quantum Tokens for Digital Signatures. *ArXiv e-prints*, September 2016.

[BZ13] Dan Boneh and Mark Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. In *Proc. of Crypto*, volume 8043 of *LNCS*, pages 361–379, 2013.

[GKPV10] Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Robustness of the learning with errors assumption. In *ICS*, 2010.

[GYZ17] Sumegha Garg, Henry Yuen, and Mark Zhandry. New security notions and feasibility results for authentication of quantum data. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017*, pages 342–371, Cham, 2017. Springer International Publishing.

[MP11] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. Cryptology ePrint Archive, Report 2011/501, 2011. https://eprint.iacr.org/2011/501.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 84–93, New York, NY, USA, 2005. ACM.

[Zha17] M. Zhandry. Quantum Lightning Never Strikes the Same State Twice. *ArXiv e-prints*, November 2017.