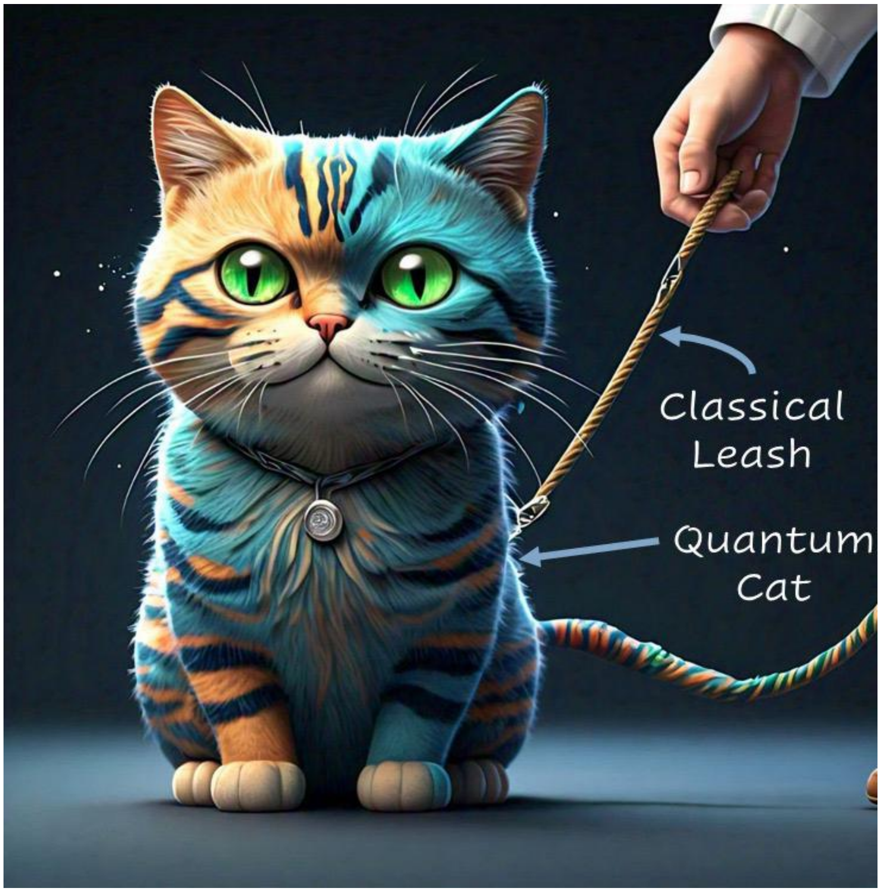
Classical Verification of Quantum Computation

Project Report

COL872: Lattices in Computer Science

Anish Banerjee
2021CS10134

Shankh Gupta
2021CS50604



Contents

§0. Notation

For any $n \in \mathbb{N} := \{1, 2, \ldots\}$, we define $[n]$ to be the set $\{1, 2, \ldots, n\}$.
For any distribution $\mathcal{D}$ over a set S, $x \leftarrow_{\mathcal{D}} S$ denotes that x is sampled from S according to the distribution $\mathcal{D}$.
Similarly for a set A, $x \leftarrow A$ denotes that x is a random, uniformly distributed element from A.
The Pauli matrices are denoted using

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

The Hadamard operator is denoted as

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

§1. Introduction

It has long been established that $\mathsf{IP} = \mathsf{PSPACE}$ [Sha92] and that $\mathsf{BQP} \subseteq \mathsf{PSPACE}$ [BV97], which implies $\mathsf{BQP} \in \mathsf{IP}$. However, since the definition of IP allows the prover to be computationally unbounded, a natural question arises: Can we achieve this with an efficient prover?
The main[1] reference for this discussion is Urmila Mahadev's groundbreaking paper on the Classical Verification of Quantum Computation [Mah23]. This work demonstrated that an efficient classical prover (specifically, a BPP machine) can verify the result of any efficient quantum computation (a BQP machine). This question was first posed by Daniel Gottesman in 2004. Earlier attempts tackled two weaker formulations:

1. [BFK09][FK17][ABOE08][ABOEM17] If the verifier has an access to a small quantum computer, verification of all efficient computations was possible.

2. [RUV12] An efficient classical verifier communicating with two entangled, non-communicating quantum provers can verify the result of an arbitrary quantum computation.

In this report, we provide a comprehensive summary of the measurement protocol described in [Mah23], with a particular focus on the construction of function families based on the Learning with Errors (LWE) problem. We include detailed explanations of the constructions, clarifying aspects that we found unclear in the original paper to enhance understanding.

§2. Overview

Caveat: In the overview, we build the measurement protocol by relying on idealized cryptographic primitives. However, we do not know how to construct these idealized primitives. The rest of the report relies on approximate versions.

2.1. Cryptographic Primitives

2.1.1. Trapdoor Claw-Free Function (TCF) Families

A trapdoor claw-free function family is a family of functions which are two-to-one and for which it is computationally difficult to find a claw. In this paper, we take the function family as $\mathcal{F} = \{f_{k,b} : \mathcal{X} \to \mathcal{Y}, b \in \{0, 1\}\}$ where $f_{k,0}$ and $f_{k,1}$ are injective with the same range, and it is hard to find a claw $f_{k,0}(x_0) = f_{k,1}(x_1)$. Further, we require two hardcore bit properties from it:

1. No QPT adversary can simultaneously return an element x in the domain of f and an *equation* d such that letting $\{x_0, x_1\}$ be the two pre-images of $f_{\mathsf{pk}}(x)$ under f_{pk} it holds that $d \neq 0^m$ and $d \cdot (x_0 + x_1) = 0$.

[1] Additional references include [BCM+21] and [Vid20]

2. There exists a d such that for all claws (x_0, x_1), $d \cdot (x_0 \oplus x_1) = 0$ and it is computationally hard to find.

Now we describe a BQP process of **state commitment**.

$$\begin{aligned}
|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle &\xrightarrow{H^{\otimes \log(|\mathcal{X}|)}|0\rangle} \frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}} (\alpha_0 |0\rangle + \alpha_1 |1\rangle) |x\rangle \\
&\xrightarrow{C-O_f} \frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}} \alpha_0 |0\rangle |x\rangle |f_{k,0}(x)\rangle + \alpha_1 |1\rangle |x\rangle |f_{k,1}(x)\rangle \\
&\xrightarrow{\text{Measure third register}} \alpha_0 |0\rangle \left|x_{0,y}\right\rangle + \alpha_1 |1\rangle \left|x_{1,y}\right\rangle, \text{Measurement output } y
\end{aligned}$$

Here $x_{0,y}$ and $x_{1,y}$ are the two preimages of y. Call the qubit containing b the **committed qubit**, the register containing $x_{b,y}$ the **preimage register** and the string y as the **commitment string**. In the interactive setting, due to the claw-free nature of the function, it is difficult for the prover to calculate both inverses $x_{0,y}$ and $x_{1,y}$ given only y. However with access to the trapdoor td, the verifier can compute both of them.
An important property of the above committed state is that it allows a logical Hadamard measurement upto an X Pauli:

$$\begin{aligned}
\alpha_0 |0\rangle \left|x_{0,y}\right\rangle + \alpha_1 |1\rangle \left|x_{1,y}\right\rangle &\xrightarrow{H^{\otimes \log(|\mathcal{X}|)+1}} \sum_{d \in \mathcal{X}} ((-1)^{d \cdot x_{0,y}} \alpha_0 |+\rangle + (-1)^{d \cdot x_{1,y}} \alpha_1 |-\rangle) |d\rangle \\
&\xrightarrow{\text{Measure preimage register}} \alpha_0 |+\rangle + (-1)^{d \cdot (x_{0,y} \oplus x_{1,y})} \alpha_1 |-\rangle = X^{d \cdot (x_{0,y} \oplus x_{1,y})} H |\psi\rangle
\end{aligned}$$

Again, in the interactive setting, the prover measures the state obtained above an sends the measurement results b', d to the verifier. The verifier decodes the measurement b' by xoring it with $d \cdot (x_0 \oplus x_1)$ to obtain the bit m which he stores as the output of the Hadamard basis measurement.

2.1.2. Trapdoor Injective Function (TIF) Families

A trapdoor injective function family $\mathcal{G} = \{g_{k,b} : \mathcal{X} \to \mathcal{Y}\}$ is a family of injective functions such that the range of $g_{k,0}$ and $g_{k,1}$ are disjoint. Given $y = g_{k,b}(x_{b,y})$, the trapdoor td allows the recovery of the $(b, x_{b,y})$. We will also require that $\mathcal{G}$ is computationally indistinguishable from $\mathcal{F}$. We can think of the above commitment process in terms of this family $\mathcal{G}$ as well, but when we measure the third register to obtain the commitment string, the superposition collapses (since the ranges of $g_{k,b}$ are disjoint) and the final state is

$$|b\rangle \left|x_{b,y}\right\rangle$$

where $x_{b,y}$ is the unique preimage of y.

2.2. Measurement Protocol

1. **Basis selection**:

   The verifier selects a random string $h \leftarrow \{0,1\}^n$. Here $h_i = 0$ corresponds to the Standard basis and $h_i = 1$ corresponds to the Hadamard basis

2. **Commitment Round:**

   (a) For $i \in [n]$, if $h_i = 1$, the verifier samples (k_i, td_i) corresponding to the pair of TCF $(f_{k_i,0}, f_{k_i,1}) \in \mathcal{F}$. Otherwise, he samples (k_i, td_i) corresponding to the pair of TIF $(g_{k_i,0}, g_{k_i,1}) \in \mathcal{G}$. He sends k_i to the prover.

   (b) An honest prover generates $n-$qubits of his choice and commits to them as discussed above. He sends the measurement results $y_1, y_2, \ldots, y_n \in \mathcal{Y}$ to the verifier.

3. **Test/Hadamard Round:**

   The verifier now chooses to run a test or a Hadamard round.

(a) ***Test:*** This round is used just for test purposes. The verifier requests a standard basis measurement of the committed qubits and preimage registers for all n qubits. He receives $b_i' \in \{0,1\}$ and $x_i' \in \mathcal{X}$ from the prover and checks if the measurement is a preimage of y_i.

(b) ***Hadamard:*** This is the actual round where the measurement results are obtained. The verifier requests a Hadamard basis measurement of the committed qubits and preimage registers for all n qubits. He receives $b_i' \in \{0,1\}$ and non-zero $d_i \in \mathcal{X}$ from the prover (rejects otherwise).

- For all i for which $h_i = 0$, he ignores the measurement results b_i and d_i. He uses the trapdoor to find the preimage of y_i and the bit b, and stores it as the **Standard basis** measurement result for the i^{th} qubit.
- For all i for which $h_i = 1$, the verifier decodes b_i' by xoring it with $d_i \cdot (x_{0,y_i} \oplus x_{1,y_i})$. The verifier stores the result $m_i = b_i' \oplus d_i \cdot (x_{0,y_i} \oplus x_{1,y_i})$ as the **Hadamard basis** measurement result for the i^{th} qubit.

Completeness follows immediately from the properties of the functions discussed above. Soundness of the protocol is non-trivial and we refer the interested reader to the original paper for the same. We will now discuss the approximate construction of the function families.

§3. Preliminaries

3.1. Hellinger Distance

Definition 3.1 (Hellinger Distance)**.** The Hellinger distance between two probability distributions f_1, f_2 is given by

$$H^2(f_1, f_2) = 1 - \sum_x \sqrt{f_1(x) f_2(x)} = 1 - F(f_1, f_2)$$

where F is the Fidelity between f_1, f_2

[2] Additionally, we have the following relation between the Hellinger distance and the Total Variation Distance:

Lemma 3.1.

$$H^2(f_1, f_2) \leq \|f_1 - f_2\|_{TV}^2 \leq 2H^2(f_1, f_2)$$

Proof.

$$\begin{aligned}
\|f_1 - f_2\|_{TV} &= \frac{1}{2}\sum_x |f_1(x) - f_2(x)| \\
&= \frac{1}{2}\sum_x \left|\sqrt{f_1(x)} - \sqrt{f_2(x)}\right| \left|\sqrt{f_1(x)} + \sqrt{f_2(x)}\right| \\
&\overset{(a)}{\geq} \frac{1}{2}\sum_x \left|\sqrt{f_1(x)} - \sqrt{f_2(x)}\right|^2 \\
&= \frac{1}{2}\sum_x f_1(x) + f_2(x) - 2\sqrt{f_1(x) f_2(x)} \\
&= \sum_x 1 - \sqrt{f_1(x) f_2(x)}
\end{aligned}$$

[2] Trivia: Fieldity is also known as the Bhattacharya Coefficient

Where (a) follows from the triangle inequality. For the other side:

$$\begin{aligned}
\|f_1 - f_2\|_{TV}^2 &= \frac{1}{4}\left(\sum_x \left|\sqrt{f_1(x)} - \sqrt{f_2(x)}\right| \left|\sqrt{f_1(x)} + \sqrt{f_2(x)}\right|\right)^2 \\
&\overset{(b)}{\leq} \frac{1}{4}\left(\sum_x \left|\sqrt{f_1(x)} - \sqrt{f_2(x)}\right|^2\right)\left(\sum_x \left|\sqrt{f_1(x)} + \sqrt{f_2(x)}\right|^2\right) \\
&\leq H^2(f_1, f_2)\left(1 + \sum_x \sqrt{f_1(x) f_2(x)}\right) \\
&= H^2(f_1, f_2)\left(2 - H^2(f_1, f_2)\right) \\
&= 2H^2(f_1, f_2)
\end{aligned}$$

Where (b) follows from Cauchy-Schwarz. ■

3.2. Learning With Errors

For a positive real B and a positive integer q, the truncated Gaussian distribution over $\mathbb{Z}_q$ with parameter B is the distribution supported on $\mathcal{D} = \{x \in \mathbb{Z}_q \mid \|x\| \leq B\}$

$$D_{\mathbb{Z}_q, B}(x) = \frac{e^{\frac{-\pi\|x\|^2}{B^2}}}{\sum_{x \in \mathcal{D}} e^{\frac{-\pi\|x\|^2}{B^2}}}$$

In higher dimensions Z_q^m, for positive integer m and parameter B, supported on $\mathcal{D}^m = \{\mathbf{x} \in \mathbb{Z}_q^m \mid \|\mathbf{x}\| \leq B\sqrt{m}\}$

$$D_{\mathbb{Z}_q^m, B}(\mathbf{x}) = D_{\mathbb{Z}_q, B}(x_1) D_{\mathbb{Z}_q, B}(x_2) \ldots D_{\mathbb{Z}_q, B}(x_m)$$

Observe that by the above definition, even the $\infty-$norm of the vectors in $\mathcal{D}^m$ should be bounded by B.

Lemma 3.2. Let B be a positive real and q, m positive integers. Consider $\mathbf{e} \in \mathcal{D}^m$. Then the Hellinger distance between the distribution $D = D_{\mathbb{Z}_q, B}$ and the shifted distribution $(D + \mathbf{e})(x) := D(x - \mathbf{e})$ satisfies

$$H^2(D, D + \mathbf{e}) \leq 1 - e^{\frac{-2\pi\sqrt{m}\|\mathbf{e}\|}{B}}$$

From this, we also get a bound on the statistical distance since $\|f_1 - f_2\|_{TV} \leq \sqrt{2H^2(f_1, f_2)}$. Thus

$$\|D - (D + \mathbf{e})\|_{TV}^2 \leq 2\left(1 - e^{\frac{-2\pi\sqrt{m}\|\mathbf{e}\|}{B}}\right)$$

Proof. Let $\tau = \sum_{x \in \mathcal{D}^m} e^{\frac{-\pi \|x\|^2}{B^2}}$. Then

$$\begin{aligned}
\sum_{\mathbf{e}_0 \in \mathcal{D}^m} \sqrt{D_{\mathbb{Z}_q,B}(\mathbf{e}_0) D_{\mathbb{Z}_q,B}(\mathbf{e}_0 - \mathbf{e})} &= \tau^{-m} \sum_{\mathbf{e}_0 \in \mathcal{D}^m} e^{\frac{-\pi(\|e_0\|^2 + \|\mathbf{e}_0 - \mathbf{e}\|^2)}{2B^2}} \\
&\geq \tau^{-m} \sum_{\mathbf{e}_0 \in \mathcal{D}^m} e^{\frac{-\pi(\|\mathbf{e}_0\|^2 + (\|\mathbf{e}_0\| + \|\mathbf{e}\|)^2)}{2B^2}} \\
&= \tau^{-m} \sum_{\mathbf{e}_0 \in \mathcal{D}^m} e^{\frac{-\pi\|\mathbf{e}_0\|^2}{B^2}} e^{\frac{-\pi\|\mathbf{e}\|^2}{2B^2}} e^{\frac{-\pi\|\mathbf{e}_0\|\|\mathbf{e}\|}{B^2}} \\
&\overset{(a)}{\geq} e^{\frac{-\pi\|\mathbf{e}\|^2}{2B^2} - \frac{\pi\sqrt{m}\|\mathbf{e}\|}{B}} \tau^{-m} \sum_{\mathbf{e}_0 \in \mathcal{D}^m} e^{\frac{-\pi\|\mathbf{e}_0\|^2}{B^2}} \\
&= e^{-\frac{\pi(\|\mathbf{e}\|^2 + 2B\sqrt{m}\|\mathbf{e}\|)}{2B^2}} \\
&\overset{(b)}{\geq} e^{\frac{-3\pi\|\mathbf{e}\|^2}{2B^2}} \geq e^{\frac{-2\pi\|\mathbf{e}\|^2}{B^2}}
\end{aligned}$$

■

Where $(a), (b)$ follow from $\|\mathbf{e}_0\|, \|\mathbf{e}\| \leq B\sqrt{m}$ respectively.

Definition 3.2 (Learning With Errors)**.** For a security parameter λ, let $n, m, q \in \mathbb{N}$ be integer functions of λ. Let $\chi = \chi(\lambda)$ be a distribution over $\mathbb{Z}$. The $\mathrm{LWE}_{n,m,q,\chi}$ problem is to distinguish between the distributions

$$(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e} \mod q) \qquad \text{and} \qquad (\mathbf{A}, \mathbf{u})$$

where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{s} \leftarrow \mathbb{Z}_q^n, \mathbf{e} \leftarrow \chi^m, \mathbf{u} \leftarrow \mathbb{Z}^m$. Denote $\mathrm{LWE}_{n,q,\chi}$ as the LWE problem when $m = \mathrm{poly}(n \log q)$.

Definition 3.3 (LWE assumption)**.** No quantum polynomial time procedure can solve the $\mathrm{LWE}_{n,q,\chi}$ problem with more than a negligible advantage, even when given access to an advice state dependent on the parameters of the problem.

It was shown in [Reg24] and [PRSD17] that for any $\alpha > 0$ such that $\sigma = \alpha q \geq 2\sqrt{n}$ the $\mathrm{LWE}_{n,q,D_{\mathbb{Z}_q,\sigma}}$ problem is at least as hard as approximating the SIVP to within a factor of $\tilde{O}(n/\alpha)$ in worst case dimension n lattices. The best known classical or quantum algorithm for these problems runs in $2^{\tilde{O}(n/\log\gamma)}$. For our construction, we assume hardness of the problem against a QPT adversary in the case γ is a super-polynomial function in n.
We require the following result, which tells us that trapdoor functions can be built from LWE.

Theorem 3.3 (Theorem 5.1 in [MP11])**.** Let $n, m \geq 1$ and $q \geq 2$ be such that $m = \Omega(n \log q)$. There is an efficient algorithm $(\mathbf{A}, t_{\mathbf{A}}) \leftarrow \textsc{GenTrap}(1^n, 1^m, q)$ such that:

- The distribution of $\mathbf{A}$ is negligibly (in n) close to the uniform distribution.
- There is an efficient inversion algorithm $(\mathbf{s}, \mathbf{e}) \leftarrow \textsc{Invert}(\mathbf{A}, t_{\mathbf{A}}, \mathbf{A}\mathbf{s} + \mathbf{e})$ where $\|\mathbf{e}\| \leq q/(C_T\sqrt{n \log q})$, C_T is a universal constant.

Additionally, we will require another property which is the existence of a "lossy mode" for LWE.

Definition 3.4 (Definition 3.1 in [AKPW13]). Let $\chi = \chi(\lambda)$ be an efficiently sample-able distribution over $\mathbb{Z}_q$. Define a lossy sampler $\tilde{\mathbf{A}} \leftarrow \mathsf{LOSSY}(1^n, 1^m, 1^\lambda, q, \chi)$ by $\tilde{\mathbf{A}} = \mathbf{BC} + \mathbf{F}$, where $\mathbf{B} \leftarrow \mathbb{Z}_q^{m\times l}, \mathbf{C} \leftarrow \mathbb{Z}_q^{l\times n}, \mathbf{F} \leftarrow \chi^{m\times n}$.

Theorem 3.4 (Lemma 3.2 in [AKPW13]). Under the $\mathrm{LWE}_{l,q,\chi}$ assumption, the distribution of a random $\tilde{\mathbf{A}} \leftarrow \mathsf{LOSSY}(1^n, 1^m, 1^\lambda, q, \chi)$ is computationally indistinguishable from $\mathbf{A} \leftarrow \mathbb{Z}_q^{m\times n}$.

3.3. Moderate Matrices

Definition 3.5 (Moderate Matrices). Let $\mathbf{b} \in \mathbb{Z}_q^n$. We say that $\mathbf{b}$ is moderate if it contains at least $n/4$ entries whose unique representative in $(-q/2, q/2]$ has its absolute value in the range $(q/8, 3q/8]$. A matrix $\mathbf{C} \in \mathbb{Z}_q^{l\times n}$ is moderate if its entire row span except 0^n is moderate.

3.4. The Pauli Twirl

The conjugation of a CPTP map by a random Pauli is called a Pauli twirl.

Lemma 3.5 (Z Pauli Twirl). For a CPTP map with Kraus operators $\{B_\tau\}_\tau$, the following CPTP maps are equal:

$$\left\{\frac{1}{\sqrt{2}}(Z^r \otimes \mathbb{I})B_r(Z^r \otimes \mathbb{I})\right\}_{r\in\{0,1\},\tau} = \left\{(X^x \otimes \mathbb{I})B'_{x,\tau}\right\}_{x\in\{0,1\},\tau}$$

where $B_\tau = \sum_{x,z\in\{0,1\}} X^x Z^z \otimes B_{xz\tau}$ and $B'_{x,\tau} = \sum_{z\in\{0,1\}} Z^z \otimes B_{xz\tau}$.

Corollary 3.6 (Z Pauli Twirl with Measurement). For a CPTP map with Kraus operators $\{B_\tau\}_\tau$, the following two CPTP maps are equal:

$$\left\{\frac{1}{\sqrt{2}}(|b\rangle\langle b|\, HZ^r \otimes \mathbb{I})B_r(Z^r \otimes \mathbb{I})\right\}_{r\in\{0,1\},\tau} = \left\{(|b\rangle\langle b|\, H \otimes \mathbb{I})B'_{x,\tau}\right\}_{x\in\{0,1\},\tau}$$

where $B_\tau = \sum_{x,z\in\{0,1\}} X^x Z^z \otimes B_{xz\tau}$ and $B'_{x,\tau} = \sum_{z\in\{0,1\}} Z^z \otimes B_{xz\tau}$.

3.5. QPIP Definition

Definition 3.6. A language $\mathcal{L}$ is said to have a Quantum Prover Interactive Proof (QPIP_τ) with completeness c and soundness s, (with $c - s$ at least a constant) if there exists a pair of algorithms $(calP, \mathcal{V})$ with the following properties:

- $\mathcal{P}$ is a BQP machine, which also has a quantum channel that can transmit τ qubits.
- $\mathcal{V}$ is a hybrid quantum classical machine.
  - The classical part is a BPP machine.
  - The quantum part is a register of τ qubits on which the verifier can perform arbitrary quantum operations and which has access to a quantum channel that can transmit τ qubits.

  The interaction between the quantum and classical parts of $\mathcal{V}$ is the usual one: the classical part controls which operations are to be performed on the quantum register, and outcomes of measurements of the quantum register can be used as input to the classical machine.
- There is also a classical communication channel between $\mathcal{P}$ and $\mathcal{V}$, that can transmit polynomially many bits at any step.
- At any given step, either $\mathcal{V}$ or $\mathcal{P}$ perform computations on their registers and send bits and qubits through the relevant channels to the other party.

We require:

Completeness: $x \in \mathcal{L} \implies$ there exists a $\mathcal{P}$ such that after interacting with it, $\mathcal{V}$ accepts with probability $\geq c$.

Soundness: $x \notin \mathcal{L} \implies$ for all $\mathcal{P}$, the interaction between $\mathcal{P}, \mathcal{V}$ leads $\mathcal{V}$ to accept with probability $\leq s$.

§4. The Fitzsimons-Morimae Protocol

In this section, we describe the receive-and-measure protocol from [FK17], as it will form the base for Mahadev's protocol. This part is mostly based on Lecture 5 of notes by Thomas Vidick [Vid20].

4.1. The Feynman-Kitaev Clock

Recall that in the proof of the Cook-Levin Theorem, we reduce the circuit to a boolean formula. The computation performed by the circuit on some input is represented as a "tableau" such that the property of being a valid tableau can be encoded into a Boolean formula. For quantum circuits, the idea of a tableau of the computation is replaced by a special register called the "clock" to keep a track of the history:

$$|\psi_h\rangle = \frac{1}{\sqrt{T+1}} \sum_{t=0}^{T} |t\rangle\, |\psi_t\rangle$$

It is possible to create such a state with a circuit that is mildly more complex than the original circuit $\mathcal{C}$. Kitaev showed that, assuming the clock register is encoded in unary, it is possible to check the correct propagation of every step of the circuit directly on this superposition by only applying local observables: there is a set of observables H_{in} and an observable H_{out} that checks the output qubit of the circuit in its right state. Further, there is an observable H_{clock} that check that the clock register is well-formed. The final result is the following theorem, which reduces quantum circuit satisfiablity to the Local Hamiltonian problem, showing that LH is QMA complete:

Theorem 4.1 (Local Hamiltonian is QMA Complete)**.** For any integer $n \geq 1$ there are $n' = \operatorname{poly}(n), \alpha = \alpha(n)$ and $\delta \geq 1/\operatorname{poly}(n)$ such that given a $T = \operatorname{poly}(n)$ gate quantum circuit $\mathcal{C} = ((G_k, i_k, j_k)_{k \in [T]})$ acting on n qubits, and an input x, there exists efficiently computable real weights $\{J_{ij} | i, j \in [n']\}$ such that $\forall i, j : |J_{ij}| \leq 1$ and if

$$H_{\mathcal{C}} = -\sum_{ij} \frac{J_{ij}}{2}(X_i X_j + Z_i Z_j)$$

then:

Completeness: If the $\mathcal{C}$ accepts x with probability $\geq 2/3$ then the smallest eigenvalue of $H_{\mathcal{C}}$ is at most α.

Soundness: If the $\mathcal{C}$ accepts x with probability $\leq 1/3$ then the smallest eigenvalue of $H_{\mathcal{C}}$ is at least $\alpha + \delta$.

This theorem gives us the recipe for the verification of quantum circuits: it is sufficient to verify the existence of a quantum state that yields certain statistics (i.e. low ground state energy) when some of its qubits are measured in the X or Z basis. This therefore extends to a protocol in which the Prover $\mathcal{P}$ is fully quantum but the Verifier $\mathcal{V}$ has a one-qubit memory with the ability to measure using either X or Z observables.

4.2. The Protocol

Fitzsimons-Hadjuček-Morimae Protocol

Let $\mathcal{C}$ be the input circuit and $H_{\mathcal{C}}$ be the n-qubit Hamiltonian of Theorem 4.1.

1. $\mathcal{V}$ initializes a counter $\gamma = 0$. They execute the following interaction with $\mathcal{P}$ independently $N = \frac{C}{\delta^2}\binom{n'}{2}^2 \ln\left(\frac{1}{\epsilon}\right)$ for sufficiently large universal constant C.
   (a) $\mathcal{P}$ prepares the ground state $|\psi\rangle$ of $H_{\mathcal{C}}$ and sends it to $\mathcal{V}$ qubit-by-qubit.
   (b) $\mathcal{V}$ selects measurement $W \leftarrow \{X, Z\}$ and measures each qubit in associated basis upon reception. Let $b_{Wi} \in \{1, -1\}$ be the outcome for the i^{th} qubit. Next they select $i \neq j \in [n']$ uniformly at random and update their counter $\gamma \leftarrow \gamma - J_{ij} b_{Wi} b_{Wj}$.
2. $\mathcal{V}$ accepts iff $\frac{\gamma}{N}\binom{n'}{2} \leq a + \delta/2$.

Figure 1: Fitzsimons-Hadjuček-Morimae Protocol, paramterized by circuit $\mathcal{C}$ and error ϵ

Theorem 4.2. Let $\mathcal{C}$ be a quantum circuit and $H_{\mathcal{C}}$ be the Hamiltonian associated to it. Let x be an input to $\mathcal{C}$ and $\epsilon > 0$ a parameter for the protocol. Then the following hold:

Completeness: If the $\mathcal{C}$ accepts x with probability $\geq 2/3$ then there exists a QPT prover that is accepted with probability at least $1 - \epsilon$.

Soundness: If the $\mathcal{C}$ accepts x with probability $\leq 1/3$ then for all provers, the acceptance probability is at most ϵ.

Proof. Let Γ be the random variable denoting the value of γ at the end of the protocol. Then we can decompose it as:

$$\Gamma = \Gamma_1 + \Gamma_2 + \cdots + \Gamma_N$$

where each $\Gamma_k = -J_{ij} b_{Wi} b_{Wj}$ for that iteration. Then, by the linearity of expectation:

$$\mathbb{E}[\Gamma] = -\frac{N}{\binom{n'}{2}} \sum_{i \neq j} \frac{J_{ij}}{2} \langle\psi| X_i X_j + Z_i Z_j |\psi\rangle = \frac{N}{\binom{n'}{2}} \langle\psi| H_{\mathcal{C}} |\psi\rangle$$

Therefore, $\frac{\mathbb{E}[\Gamma]}{N}\binom{n'}{2}$ is the ground state energy, that by Theorem 4.1 will be $\geq \alpha$ or $\leq \alpha + \delta$ depending on x. What remains to show is that the actual value of γ will be close to $\mathbb{E}[\Gamma]$ with a high probability. Since all the trials are independent, this can be easily done using a Hoeffding's bound. This also gives the expression for N in terms of ϵ.

■

§5. Function Definitions

Unfortunately, we don't have exact constructions of the primitives discussed in the introduction. So, we will build approximate function families using Learning With Errors. The major changes from the original definition are:

- The range of the functions is a probability density $\mathcal{D}_\mathcal{Y}$ over $\mathcal{Y}$ instead of being $\mathcal{Y}$. Each function returns a density rather than a point.
- The trapdoor injective pair property , i.e. the pair (f_0, f_1), is defined in terms of support of the densities: their supports should be identical in the case of a colliding pair (claw), and disjoint otherwise.
- We require an QPT procedure which generates the state

  $$\frac{1}{|\mathcal{X}|}\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\sqrt{(f_{k,b}(x))(y)}\,|x\rangle\,|y\rangle$$

  Unfortunately, it is not possible to build this perfectly using the current construction. Nevertheless, we can create an approximation of this using a related family $f'_{k,b}$. This is however not needed in the construction of the TIF family, the above state can be exactly generated.
- The adaptive hardcore bit property is modified too using an injective map $J : \mathcal{X} \to \{0,1\}^w$ which basically maps an integer to its binary representation.

5.1. NTCF Family

Definition 5.1 (Noisy Trapdoor Claw-free Function family)**.** Let λ be the security parameter, $\mathcal{X}, \mathcal{Y}, \mathcal{K}_\mathcal{F}$ be finite sets. A family of functions

$$\mathcal{F} = \{f_{k,b} : \mathcal{X} \to \mathcal{D}_\mathcal{Y}\}_{k\in\mathcal{K}_\mathcal{F}, b\in\{0,1\}}$$

is called a NTCF family if the following hold:

1. **Efficient Function Generation:** There exists a PPT algorithm which generates the key and the trapdoor

   $$(k, \mathsf{td}) \leftarrow \mathrm{GEN}(1^\lambda)$$

2. **Trapdoor injective pair:**
   (a) *Trapdoor:* There exists an efficient deterministic inversion algorithm such that for $y \in \textsc{Supp}(f_{k,b}(x))$

   $$x \leftarrow \textsc{Inv}(k, \mathsf{td}, b, y)$$

   This also implies that for $x \neq x', \textsc{Supp}(f_{k,b}(x)) \cap \textsc{Supp}(f_{k,b}(x')) = \phi$

   (b) *Injective Pair:* There is a perfect matching $\mathcal{R}_k \subseteq \mathcal{X} \times \mathcal{X}$ such that

   $$f_{k,0}(x_0) = f_{k,1}(x_1) \Leftrightarrow (x_0, x_1) \in \mathcal{R}_k$$

3. **Efficient Range Superposition:** For all $k \in \mathcal{K}_\mathcal{F}, b \in \{0,1\}$, there exists a function $f'_{k,b} : \mathcal{X} \to \mathcal{D}_\mathcal{Y}$ such that:

(a) *Inversion:* For all $(x_0, x_1) \in \mathcal{R}_k, y \in \textsc{Supp}(f'_{k,b}(x_b))$

$$x_b \leftarrow \textsc{Inv}_{\mathcal{F}}(\mathsf{td}, b, y) \qquad\qquad x_{b\oplus 1} \leftarrow \textsc{Inv}_{\mathcal{F}}(\mathsf{td}, b \oplus 1, y)$$

(b) *Check:* There exists an efficient deterministic procedure $b' \leftarrow \textsc{Chk}_{\mathcal{F}}(k, b, x, y)$ where $b' = 1$ iff $y \in \textsc{Supp}(f'_{k,b}(x))$. Observe that $\textsc{Chk}_{\mathcal{F}}$ doesn't get the trapdoor.

(c) *Close to $\mathcal{F}$:* For every k, b

$$\mathop{\mathbb{E}}_{x \leftarrow \mathcal{X}}[H^2(f_{k,b}(x), f'_{k,b}(x)] \leq \text{negl}(\lambda)$$

(d) *Efficient Sampling:* There exists an efficient sampling procedure $\textsc{Samp}_{\mathcal{F}}$ that prepares the state

$$\frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} \sqrt{f'_{k,b}(x)}\, |x\rangle\, |y\rangle \leftarrow \textsc{Samp}_{\mathcal{F}}(k, b)$$

4. **Adaptive Hardcore Bit:** For all keys $\mathcal{K}_{\mathcal{F}}$, the following conditions hold for $w = \text{poly}(\lambda)$

(a) For all $b \in \{0, 1\}$ and $x \in \mathcal{X}$, there exists a set $G_{k,b,x}$ such that

$$\Pr_{d \leftarrow \{0,1\}^w}[d \notin G_{k,b,x}] \leq \text{negl}(\lambda)$$

Moreover there exists an efficient algorithm which checks for membership in $G_{k,b,x}$ given k, b, x, td.

(b) There is an efficiently computable injection $J : \mathcal{X} \to \{0, 1\}^w$, such that J can be inverted efficiently on its range and if

$$H_k = \{(b, x_b, d, d \cdot J(x_0) \oplus J(x_1)) | b \in \{0, 1\}, (x_0, x_1) \in \mathcal{R}_k, d \in G_{k,0,x_0} \cap G_{k,1,x_1}\}$$
$$\bar{H_k} = \{(b, x_b, d, c) | (b, x, d, c \oplus 1) \in H_k\}$$

then for any QPT adversary $\mathcal{A}$

$$\left| \Pr_{(k,\mathsf{td}) \leftarrow \text{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(k) \in H_k] - \Pr_{(k,\mathsf{td}) \leftarrow \text{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(k) \in \bar{H_k}] \right| \leq \text{negl}(\lambda)$$

5.2. TIF Family

Definition 5.2 (Trapdoor Injective Function Family)**.** Let λ be the security parameter, $\mathcal{X}, \mathcal{Y}, \mathcal{K}_{\mathcal{G}}$ be finite sets. A family of functions

$$\mathcal{G} = \{g_{k,b} : \mathcal{X} \to \mathcal{D}_{\mathcal{Y}}\}_{k \in \mathcal{K}_{\mathcal{G}}, b \in \{0,1\}}$$

is called a TIF family if the following hold:

1. **Efficient Function Generation:** There exists a PPT algorithm which generates the key and the trapdoor

$$(k, \mathsf{td}) \leftarrow \text{GEN}_{\mathcal{G}}(1^\lambda)$$

2. **Disjoint Trapdoor injective pair:**

(a) *Trapdoor:* There exists an efficient deterministic inversion algorithm such that for $y \in \textsc{Supp}(g_{k,b}(x))$

$$(b, x) \leftarrow \textsc{Inv}_{\mathcal{G}}(\mathsf{td}, y)$$

(b) *Disjoint Injective Pair:* For all $k \in \mathcal{K}_{\mathcal{G}}, b, b' \in \{0,1\}, x, x' \in \mathcal{X}$, if $(b,x) \neq (b',x')$,

$$\textsc{Supp}(g_{k,b}(x)) \cap \textsc{Supp}(g_{k,b}(x')) = \phi$$

3. **Efficient Range Superposition:** For all $k \in \mathcal{K}_{\mathcal{G}}, b \in \{0,1\}$

   (a) *Check:* There exists an efficient deterministic procedure $b' \leftarrow \textsc{Chk}_{\mathcal{G}}(k,b,x,y)$ where $b' = 1$ iff $y \in \textsc{Supp}(g_{k,b}(x))$. Observe that $\textsc{Chk}_{\mathcal{G}}$ doesn't get the trapdoor.

   (b) *Efficient Sampling:* There exists an efficient sampling procedure $\textsc{Samp}_{\mathcal{G}}$ that prepares the state

$$\frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} \sqrt{g_{k,b}(x)} \, |x\rangle \, |y\rangle \leftarrow \textsc{Samp}_{\mathcal{G}}(k,b)$$

5.3. ETCF Family

Now we extend the definition of NTCF family to create the Extended Trapdoor Claw-free function family.

Definition 5.3 (Injective Invariance)**.** A NTCF family $\mathcal{F}$ is **injective invariant** if there exists a trapdoor injective family $\mathcal{G}$ such that:

1. The algorithms $\textsc{Chk}_{\mathcal{F}}$ and $\textsc{Samp}_{\mathcal{F}}$ are the same as the algorithms $\textsc{Chk}_{\mathcal{G}}$ and $\textsc{Samp}_{\mathcal{G}}$.
2. For all QPT adversaries $\mathcal{A}$,

$$\left| \Pr_{(k,\mathsf{td}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})} [\mathcal{A}(k) \in H_k] - \Pr_{(k,\mathsf{td}) \leftarrow \mathrm{GEN}_{\mathcal{G}}(1^{\lambda})} [\mathcal{A}(k) \in \bar{H}_k] \right| \leq \operatorname{negl}(\lambda)$$

Definition 5.4 (Extended Trapdoor Claw-Free Family)**.** A NTCF family $\mathcal{F}$ is an ETCF if:

1. It is **injective invariant**.
2. **Hardcore Bit 2:** For all $k \in \mathcal{K}_{\mathcal{F}}, d \in \{0,1\}^w$ let

$$H'_{k,d} = \{d \cdot (J(x_0) \oplus J(x_1)) | (x_0, x_1) \in \mathcal{R}_k\}$$

   There exists a string $d \in \{0,1\}^w$ such that for all QPT $\mathcal{A}$

$$\left| \Pr_{(k,\mathsf{td}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})} [\mathcal{A}(k) \in H'_{k,d}] - \frac{1}{2} \right| \leq \operatorname{negl}(\lambda)$$

Informally,

$$\text{ETCF = NTCF + HB}_2\text{ + TIF + Injective Invariance}$$

§6. Construction from LWE

6.1. NTCF+HB$_2$

Let λ be the security parameter. Let $q \geq 2$ be a prime, $l, n, m, w \geq 1$ be polynomially-bounded functions of λ, and B_L, B_V, B_P be positive integers. We use the following constraints on the parameters:

(**A**1) $n = \Omega(l \log q + \lambda)$

(**A**2) $m = \Omega(n \log q)$

(**A**3) $w = n\lceil \log q \rceil$

(**A**4) $B_P = \frac{q}{2C_T\sqrt{mn\log q}}$, where C_T is the universal constant in Theorem 3.3

(**A**5) $2\sqrt{n} \leq B_L < B_V < B_P$

(**A**6) The ratios $\frac{B_P}{B_V}, \frac{B_V}{B_L}$ are both super-polynomial in λ.

Let $\mathcal{X} = \mathbb{Z}_q^n$ and $\mathcal{Y} = \mathbb{Z}_q^m$. The key space $\mathcal{K}_\mathcal{F}$ is a subset of $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$. We define the density function for a $b \in \{0,1\}$, $x \in \mathcal{X}$ and key $k = (\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$ as:

$$\forall \mathbf{y} \in \mathcal{Y}, \quad (f_{k,b}(\mathbf{x}))(\mathbf{y}) = D_{\mathbb{Z}_q^m, B_P}(\mathbf{y} - \mathbf{A}\mathbf{x} - b\mathbf{A}\mathbf{s})$$

We now prove the following theorem:

Theorem 6.1. For any choice of parameters satisfying conditions 6.1, the function family $\mathcal{F}_{\text{LWE}}$ is an extended trapdoor claw-free function family under the hardness assumption $\text{LWE}_{l,q,D_{\mathbb{Z}_q,B_L}}$

Proof. We verify each of the properties of an ETCF as follows:

1. **Efficient Function Generation:** $\text{GEN}_\mathcal{F}$ is defined as follows:
   - Sample $(\mathbf{A}, t_\mathbf{A}) \leftarrow \textsc{GenTrap}(1^n, 1^m, q)$, where $\textsc{GenTrap}$ is as defined in Theorem 3.3.
   - Sample $\mathbf{s} \leftarrow \{0,1\}^n$, and a vector $\mathbf{e} \leftarrow_{D_{\mathbb{Z}_q^m, B_V}} \mathbb{Z}_q^m$.
   - Return $k = (\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$ and $\text{td} = t_\mathbf{A}$.
2. **Trapdoor Injective Pair:**

   (a) *Trapdoor:* Observe that, for any key $k = (\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$ and for all $x \in \mathcal{X}$,

$$\textsc{Supp}(f_{k,0}(x)) = \left\{\mathbf{A}\mathbf{x} + \mathbf{e}_0 | \, \|\mathbf{e}_0\| \leq B_p\sqrt{m}\right\}$$
$$\textsc{Supp}(f_{k,1}(x)) = \left\{\mathbf{A}(\mathbf{x}+\mathbf{s}) + \mathbf{e}_0 | \, \|\mathbf{e}_0\| \leq B_p\sqrt{m}\right\}$$

   Given $k = (\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}), \text{td}, b, \mathbf{y}$, the $\textsc{Inv}$ algorithm uses $(\mathbf{x}+b\mathbf{s}, \mathbf{e}_0) \leftarrow \textsc{Invert}(\mathbf{A}, \text{td}, \mathbf{y})$ and using b, returns $\mathbf{x}$. Our choice of B_V ensures that the $\textsc{Invert}$ algorithm of the trapdoor works correctly. Note that we also require $\mathbf{s}$ to recover $\mathbf{x}$, which can be found by inverting the second component of the key. The condition on the supports follows from the correctness of the inversion algorithm of the trapdoor function.

   (b) *Injective Pair:* As seen above, the matching is given by $(\mathbf{x}, \mathbf{x}-\mathbf{s})$ because for both the supports to overlap, $\mathbf{x}_0 = \mathbf{x}_1 + \mathbf{s}$.

3. **Efficient Range Superposition:** Define

$$(f'_{k,b}(\mathbf{x}))(\mathbf{y}) = D_{\mathbb{Z}_q^m,B_V}(\mathbf{y} - \mathbf{A}\mathbf{x} - b(\mathbf{A}\mathbf{s} + \mathbf{e}))$$

   (a) *Inversion:* Observe that

$$\textsc{Supp}(f'_{k,0}(x)) = \left\{\mathbf{A}\mathbf{x} + \mathbf{e}_0 \middle| \, \|\mathbf{e}_0\| \leq B_p\sqrt{m}\right\}$$
$$\textsc{Supp}(f'_{k,1}(x)) = \left\{\mathbf{A}(\mathbf{x} + \mathbf{s}) + \mathbf{e}_0 + \mathbf{e} \middle| \, \|\mathbf{e}_0\| \leq B_p\sqrt{m}\right\}$$

   We no longer have the perfect matching property as above, but the inversion procedure still works correctly. For $b = 0$, there is no change. For $b = 1$,

$$\|\mathbf{e}_0 + \mathbf{e}\| \leq (B_P + B_V)\sqrt{m} \leq 2B_P\sqrt{m}$$

   which still satisfies the condition in Theorem 3.3.

   (b) *Check:* Given $k = (\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e}), b, \mathbf{x}, \mathbf{y}$, the $\textsc{Chk}_{\mathcal{F}}$ procedure simply finds the bit b' for which

$$\left\|\mathbf{y} - \mathbf{A}\mathbf{x} - b'(\mathbf{A}\mathbf{s} + \mathbf{e})\right\| \leq B_P\sqrt{m}$$

   and returns 1 iff $b = b'$.

   (c) *Close to $\mathcal{F}$:* For $b = 0$, the functions are identical. For $b = 1$ they are Gaussians shifted by $\mathbf{e}$, and by Lemma 3.2, the distance between them is negligible by our choice constraints in Section 6.1:

$$H^2(f_{k,1}(\mathbf{x}), f'_{k,1}(\mathbf{x})) \leq 1 - e^{\frac{-2\pi m B_V}{B_P}} \leq \frac{2\pi m B_V}{B_P}$$

   (d) *Efficient Sampling:* From [Reg24] Lemma 3.12, we use the result that the following state can be efficiently prepared:

$$|\psi_0\rangle = \sum_{\mathbf{e}_0 \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m,B_P}(\mathbf{e}_0)}\,|\mathbf{e}_0\rangle$$

   We obtain the required state by these transformations:

$$\begin{aligned}
|\psi_0\rangle &\xrightarrow{\text{Add an auxiliary registers}} \sum_{\mathbf{e}_0 \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m,B_P}(\mathbf{e}_0)}\,|\mathbf{e}_0\rangle\,|0\rangle\,|0\rangle \\
&\xrightarrow{\text{Compute uniform superposition over second register}} \frac{1}{\sqrt{q^n}} \sum_{\mathbf{x} \in \mathbb{Z}_q^n \mathbf{e}_0 \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m,B_P}(\mathbf{e}_0)}\,|\mathbf{e}_0\rangle\,|\mathbf{x}\rangle\,|0\rangle \\
&\xrightarrow{\text{Compute third register}} \frac{1}{\sqrt{q^n}} \sum_{\mathbf{x} \in \mathbb{Z}_q^n \mathbf{e}_0 \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m,B_P}(\mathbf{e}_0)}\,|\mathbf{e}_0\rangle\,|\mathbf{x}\rangle\,|\mathbf{e}_0 - \mathbf{A}\mathbf{x} - b(\mathbf{A}\mathbf{s} + \mathbf{e})\rangle \\
&\xrightarrow{\text{Uncompute and discard first register}} \frac{1}{\sqrt{q^n}} \sum_{\mathbf{x} \in \mathbb{Z}_q^n \mathbf{e}_0 \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m,B_P}(\mathbf{e}_0)}\,|\mathbf{x}\rangle\,|\mathbf{e}_0 + \mathbf{A}\mathbf{x} + b(\mathbf{A}\mathbf{s} + \mathbf{e})\rangle \\
&= \frac{1}{\sqrt{q^n}} \sum_{\mathbf{x} \in \mathbb{Z}_q^n, \mathbf{y} \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m,B_P}(\mathbf{y} - \mathbf{A}\mathbf{x} - b(\mathbf{A}\mathbf{s} + \mathbf{e}))}\,|\mathbf{x}\rangle\,|\mathbf{y}\rangle = \frac{1}{\sqrt{q^n}} \sum_{\mathbf{x} \in \mathbb{Z}_q^n, \mathbf{y} \in \mathbb{Z}_q^m} \sqrt{(f'_{k,b}(x))(y)}\,|\mathbf{x}\rangle\,|\mathbf{y}\rangle
\end{aligned}$$

   Which is the required superposition.

4. **Adaptive Hardcore Bit 1:**

   (a) Let $J : \mathcal{X}^n \to \{0,1\}^w$ be the binary representation of $\mathbf{x} \in \mathcal{X}$. Define $I_{b,\mathbf{x}}(d) \in \{0,1\}^n$ to be the vector whose each coordinate is obtained by taking the inner product mod 2 of the corresponding block of

$\lceil \log q \rceil$ coordinates of d and of $J(\mathbf{x}) \oplus J(\mathbf{x} - (-1)^b \mathbf{1})$ where $\mathbf{1}$ is the vector with all entries equal to 1. For $k = (\mathbf{A}, \mathbf{As} + \mathbf{e})$ and $\mathbf{x} \in \mathcal{X}$, we define the set

$$G_{k,0,\mathbf{x}} := \left\{ d \in \{0,1\}^w \middle| \exists i \in \left\{0, \dots, \frac{n}{2}\right\} : (I_{b,\mathbf{x}}(d))_i = 1 \right\}$$
$$G_{k,1,\mathbf{x}} := \left\{ d \in \{0,1\}^w \middle| \exists i \in \left\{\frac{n}{2}, \dots, n\right\} : (I_{b,\mathbf{x}}(d))_i = 1 \right\}$$

Observation 1: For all $b, \mathbf{x}$, if d is sampled uniformly at random, $d \notin G_{k,b,\mathbf{x}}$ with negligible probability. This is because J is injective, implying that each of the n components of $J(\mathbf{x}) \oplus J(\mathbf{x} - (-1)^b \mathbf{1})$ is not zero. Since d is sampled uniformly at random, the probability that the inner product mod 2 of each component of length $\lceil \log q \rceil$ is zero is $\frac{1}{2}$ as exactly half of the bitstrings will be orthogonal to this bitstring, giving a total probability of 2^{-n} which is negligible.

Observation 2: Checking membership in $G_{k,b,\mathbf{x}}$ can be done efficiently using just b, x.

(b) Given $(\mathbf{x}_0, \mathbf{x}_1) \in \mathcal{R}_k$, we know that $\mathbf{x}_1 = \mathbf{x}_0 - \mathbf{s}$. We split $\mathbf{s} = (\mathbf{s}_0, \mathbf{s}_1)$ into two equal halves. Also, introduce the following set, where $y = f_{k,0}(x_0) = f_{k,1}(x_1)$

$$\hat{G}_{s_1,0,x_0} = \hat{G}_{s_0,1,x_1} = G_{k,0,x_0} \cap G_{k,1,x_1}$$

Lemma 6.2. Assume a choice of parameters satisfying conditions 6.1 and the hardness of $\mathrm{LWE}_{l,q,D_{\mathbb{Z}_q},B_L}$. Let $s \in \{0,1\}^n$ and

$$H_s = \{(b, x, d, d \cdot (J(\mathbf{x}) \oplus J(\mathbf{x} - (-1)^b \mathbf{s})))| b \in \{0,1\}, \mathbf{x} \in \mathcal{X}, d \in \hat{G}_{s \oplus 1, b, x}\}$$
$$\bar{H}_s = \{(b, x, d, c \oplus 1)|(b, x, d, c) \in H_s\}$$

Then for any QPT procedure

$$\mathcal{A} : \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m \to \{0,1\} \times \mathcal{X} \times \{0,1\}^w \times \{0,1\}$$

$$\left| \Pr_{(k,\mathrm{td}) \leftarrow \mathrm{GEN}(1^\lambda)} [\mathcal{A}(k) \in H_s] - \Pr_{(k,\mathrm{td}) \leftarrow \mathrm{GEN}(1^\lambda)} [\mathcal{A}(k) \in \bar{H}_s] \right| \leq \mathrm{negl}(\lambda)$$

[3] We prove this in 3 steps. First consider the following lemma:

Lemma 6.3. Assume a choice of parameters satisfying conditions 6.1 and the hardness of $\mathrm{LWE}_{l,q,D_{\mathbb{Z}_q},B_L}$. Let

$$\mathcal{A} : \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m \to \{0,1\} \times \mathcal{X} \times \{0,1\}^w \times \{0,1\}$$

be a quantum poly-time procedure. Then the distributions

$$D_0 = \Big((\mathbf{A}, \mathbf{As} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ (b, x, d, c) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{As} + \mathbf{e}),\ I_{b,x}(d) \cdot \mathbf{s} \mod 2 \Big)$$

and

$$D_1 = \Big((\mathbf{A}, \mathbf{As} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ (b, x, d, c) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{As} + \mathbf{e}),\ (\delta_{d \in \hat{G}_{s_{b \oplus 1}, b, x}} r) \oplus I_{b,x}(d) \cdot \mathbf{s} \mod 2 \Big)$$

where $r \leftarrow \{0,1\}$ and $\delta_{d \in \hat{G}_{s_{b \oplus 1}, b, x}}$ is 1 if $d \in \hat{G}_{s_{b \oplus 1}, b, x}$ and 0 otherwise, are computationally indistinguishable.

[3] Note that this lemma is equivalent to condition (b) of 4. This is because, for our construction, any $(\mathbf{x}_0, \mathbf{x}_1) \in \mathcal{R}_k$ satisfy the relation $\mathbf{x}_0 = \mathbf{x}_1 + \mathbf{s}$

To prove Lemma 6.2, we will first show that Lemma 6.3 implies 6.2. The proof for this statement can be found in the appendix (Proof A.1).

It is thus sufficient to give a proof for 6.3. Before jumping into the proof, first consider the following lemma that would be useful in the proof.

Lemma 6.4 (Moderate Matrix Lemma)**.** Let q be a prime, $l, n \geq 1$ integers and $\mathbf{C} \in \mathbb{Z}_q^{l \times n}$ a uniformly random matrix. With probability at least $1 - q^l \cdot 2^{-\frac{n}{8}}$ over the choice of $\mathbf{C}$, the following holds: For a fixed $\mathbf{C}$, all $\mathbf{v} \in \mathbb{Z}_q^l$ and $\hat{d} \in \{0,1\}^n \backslash \{0^n\}$, the distribution of $(\hat{d} \cdot \mathbf{s} \mod 2)$, where $\mathbf{s}$ is uniform in $\{0,1\}^n$ conditioned on $\mathbf{Cs} = \mathbf{v}$ is within statistical distance $O(q^{\frac{3l}{2}} \cdot 2^{-\frac{n}{40}})$ of the uniform distribution over $\{0,1\}$.

Proof. The proof for this lemma can be found in the appendix. ■

We now prove our lemma 6.3 through a sequence of hybrid distributions. Let

$$HD^{(1)} = \Big((\tilde{\mathbf{A}}, \tilde{\mathbf{A}}\mathbf{s} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ (b, x, d, c) \leftarrow \mathcal{A}(\tilde{\mathbf{A}}, \tilde{\mathbf{A}}\mathbf{s} + \mathbf{e}),\ I_{b,x}(d) \cdot \mathbf{s} \mod 2 \Big),$$

where $\tilde{\mathbf{A}} = \mathbf{BC} + \mathbf{F} \leftarrow \mathsf{LOSSY}(1^n, 1^m, 1^l, q, D_{\mathbb{Z}_q, B_L})$ is sampled from a lossy sampler. From Theorems 3.3 and 3.4, we know that both $\mathbf{A}$ (generated by GENTRAP) and $\tilde{\mathbf{A}}$ are negligibly far from the uniform distribution. Hence, we can say that $D_0 \approx_c HD^{(1)}$.

Next we remove the term $\mathbf{Fs}$ from the lossy LWE sample $\tilde{\mathbf{A}}\mathbf{s} + \mathbf{e}$ to obtain the distribution,

$$HD^{(2)} = \Big((\mathbf{BC} + \mathbf{F}, \mathbf{BCs} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ (b, x, d, c) \leftarrow \mathcal{A}(\mathbf{BC} + \mathbf{F}, \mathbf{BCs} + \mathbf{e}),\ I_{b,x}(d) \cdot \mathbf{s} \mod 2 \Big).$$

Observe that $||\mathbf{Fs}|| \leq n\sqrt{m}B_L$ [4]. Applying Lemma 3.2, the statistical distance between $HD^{(1)}$ and $HD^{(2)}$ is at most

$$\gamma = \sqrt{2}\Big(1 - e^{\frac{-2\pi mnB_L}{B_V}}\Big)^{1/2},$$

which is negligible due to the condition (A6) in 6.1. Next observe that the distribution $HD^{(2)}$ depends on $\mathbf{s}$ only through the terms $\mathbf{BCs}$ and $I_{b,x}(d) \cdot \mathbf{s}$. Since $\mathbf{B}, \mathbf{C}$ are uniformly random matrices, it follows from Lemma 6.4 that the distribution of $I_{b,x}(d) \cdot \mathbf{s} \mod 2$ is statistically indistinguishable (Provided that $n = \Omega(l \log q + \lambda)$, which is satisfied by condition (A1) in 6.1) from $r \leftarrow \{0,1\}$ as long as not all bits of $I_{b,x}(d)$ are 0. Hence, the distribution $HD^{(2)}$ is statistically indistinguishable from

$$HD^{(3)} = \Big((\mathbf{BC} + \mathbf{F}, \mathbf{BCs} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ (b, x, d, c) \leftarrow \mathcal{A}(\mathbf{BC} + \mathbf{F}, \mathbf{BCs} + \mathbf{e}),\ (\delta_{d \in \hat{G}_{s_{b \oplus 1}, b, x}} r) \oplus I_{b,x}(d) \cdot \mathbf{s} \mod 2 \Big),$$

where $r \leftarrow \{0,1\}$. Next, we reintroduce the $\mathbf{Fs}$ term to obtain

$$HD^{(4)} = \Big((\tilde{\mathbf{A}}, \tilde{\mathbf{A}}\mathbf{s} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ (b, x, d, c) \leftarrow \mathcal{A}(\tilde{\mathbf{A}}, \tilde{\mathbf{A}}\mathbf{s} + \mathbf{e}),\ (\delta_{d \in \hat{G}_{s_{b \oplus 1}, b, x}} r) \oplus I_{b,x}(d) \cdot \mathbf{s} \mod 2 \Big).$$

Statistical indistinguishability between $HD^{(3)}$ and $HD^{(4)}$ follows similarly as between $HD^{(1)}$ and $HD^{(2)}$. Finally, computational indistinguishability between $HD^{(4)}$ and D_1 follows similarly as between D_0 and $HD^{(1)}$. Thus, we obtain

$$D_0 \approx_c HD^{(1)} \approx HD^{(2)} \approx HD^{(3)} \approx HD^{(4)} \approx_c D_1$$

$$\implies D_0 \approx_c D_1$$

5. **Hardcore bit 2:** The proof for Hardcore Bit 2 property follows in a similar manner as AHB. To show that our construction satisfies the HB2 property, we prove the following lemma.

[4] This is because $\mathbf{s}$ is a binary vector and the entries of $\mathbf{F}$ are sampled from a B_L-bounded distribution

Lemma 6.5. Assume a choice of parameters satisfying the conditions 6.1. Assume the hardness assumption $\mathrm{LWE}_{l,q,D_{\mathbb{Z}_q,B_L}}$ holds. Let $\mathbf{s} \in \{0,1\}^n$ and for $d \in \{0,1\}^w$[5] let

$$H'_{s,d} = \{d \cdot (J(\mathbf{x}) \oplus j(\mathbf{x} - \mathbf{s}) | \mathbf{x} \in \mathcal{X}\}.$$

Then for all $\hat{d} \in \{0,1\}^n \backslash \{0^n\}$ and for any quantum polynomial-time procedure

$$\mathcal{A} : \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^n \to \{0,1\}$$

there exists a negligible function $\mu(.)$ such that

$$\left| \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)} [\mathcal{A}(\mathbf{A}, \mathbf{As} + \mathbf{e}) \in H'_{s,J(\hat{d})}] - \frac{1}{2} \right| \leq \mu(1^\lambda).$$

To prove this lemma, First consider a variant of this lemma (Lemma 6.6) which we prove here. We later claim that the following Lemma 6.6 implies Lemma 6.5.

Lemma 6.6. Assume a choice of parameters satisfying conditions 6.1. Under the hardness assumption $\mathrm{LWE}_{l,q,D_{\mathbb{Z}_q,B_L}}$, for all $\hat{d} \in \{0,1\}^n \backslash \{0^n\}$, the distributions

$$D_0 = \Big((\mathbf{A}, \mathbf{As} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ \hat{d} \cdot \mathbf{s} \mod 2 \Big)$$

$$D_1 = \Big((\mathbf{A}, \mathbf{As} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ r \Big),$$

where $r \leftarrow \{0,1\}$, are computationally indistinguishable.

Proof. The proof of this lemma is very similar to the proof of Lemma 6.3. We present a series of Hybrid distributions to show that the above two distributions D_0 and D_1 are computationally indistinguishable, where each of the indistinguishability argument follows in the same way as in the proof of Lemma 6.3

$$\begin{aligned}
D_0 &= \Big((\mathbf{A}, \mathbf{As} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ \hat{d} \cdot \mathbf{s} \mod 2 \Big) \\
\approx HD^{(1)} &= \Big((\tilde{\mathbf{A}}, \tilde{\mathbf{A}}\mathbf{s} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ \hat{d} \cdot \mathbf{s} \mod 2 \Big) \\
\approx HD^{(2)} &= \Big((\mathbf{BC} + \mathbf{F}, \mathbf{BCs} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ \hat{d} \cdot \mathbf{s} \mod 2 \Big) \\
\approx HD^{(3)} &= \Big((\mathbf{BC} + \mathbf{F}, \mathbf{BCs} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ r \Big) \\
\approx HD^{(4)} &= \Big((\tilde{\mathbf{A}}, \tilde{\mathbf{A}}\mathbf{s} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ r \Big) \\
\approx D_1 &= \Big((\mathbf{A}, \mathbf{As} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda),\ r \Big)
\end{aligned}$$

■

Claim 6.7. Lemma 6.6 implies Lemma 6.5.

Proof. The proof is similar to the proof A.1 and can be found in Appendix (Proof A.2) ■

Hence by Claim 6.7 and Lemma 6.6 we can conclude that our construction satisfies the HB2 property.

■

6.2. TIF

We will show that for key $k = (\mathbf{A}, \mathbf{u})$

$$(g_{k,b}(x))(y) = D_{\mathbb{Z}_q^m, B_P}(\mathbf{y} - \mathbf{A}\mathbf{x} - b\mathbf{u})$$

satisfies the definition of trapdoor injective function given above.

1. **Efficient Function Generation:** This is similar to that for TCF family, but we modify the way we obtain the second component of the key.

   - Sample $(\mathbf{A}, t_{\mathbf{A}}) \leftarrow \textsc{GenTrap}(1^n, 1^m, q)$, where $\textsc{GenTrap}$ is as defined in Theorem 3.3.
   - ~~Sample $\mathbf{s} \leftarrow \{0,1\}^n$, and a vector $\mathbf{e} \leftarrow D_{\mathbb{Z}_q^m, B_V}$ $\mathbb{Z}_q^m$~~ ★ Sample $\mathbf{u} \leftarrow \mathbb{Z}_q^m$. Now using the trapdoor apply $(\mathbf{s}, \mathbf{e}) \leftarrow \textsc{Invert}(\mathbf{A}, t_{\mathbf{A}}, \mathbf{u})$ and check if $\mathbf{A}\mathbf{s} + \mathbf{e} = \mathbf{u}$ and $\|\mathbf{e}\| \leq 2B_P\sqrt{m}$ (This tells us that the inversion procedure ran correctly). If so, then discard $\mathbf{u}$ and repeat again.
   - Return $k = (\mathbf{A}, \mathbf{u})$ and td $= t_{\mathbf{A}}$.

   Due to the setting of parameters, the rejection in the modified procedure happens with negligible probability, and hence the distribution is close to uniform. This also causes the inversion procedure of the trapdoor function to not work correctly on $\mathbf{u}$ since there do not exist $\mathbf{s}, \mathbf{e}$ with $\|\mathbf{e}\| \leq 2B_P\sqrt{m}$ such that $\mathbf{u} = \mathbf{A}\mathbf{s} + \mathbf{e}$. Observe that this puts a constraint on the norm of $\mathbf{u}$, namely, $\|\mathbf{u}\| \geq 2B_p\sqrt{m}$, as otherwise we can set $\mathbf{s} = 0$ and $\mathbf{e} = \mathbf{u}$. This is required for the $\textsc{Inv}_{\mathcal{G}_{\text{LWE}}}$ function described below.

2. **Disjoint Trapdoor Injective Pair:**

   (a) *Trapdoor:* Observe that

   $$\textsc{Supp}(g_{k,0}(x)) = \left\{\mathbf{A}\mathbf{x} + \mathbf{e}_0 \middle| \, \|\mathbf{e}_0\| \leq B_p\sqrt{m}\right\}$$
   $$\textsc{Supp}(g_{k,1}(x)) = \left\{\mathbf{A}\mathbf{x} + \mathbf{e}_0 + \mathbf{u} \middle| \, \|\mathbf{e}_0\| \leq B_p\sqrt{m}\right\}$$

   The procedure $\textsc{Inv}_{\mathcal{G}_{\text{LWE}}}$ takes as input $(t_{\mathbf{A}}, \mathbf{y})$ where $\mathbf{y} \in \mathcal{Y}$ and runs the algorithm $\textsc{Invert}$ on $\mathbf{y}$.

   - If it outputs $(\mathbf{s}_0, \mathbf{e}_0)$ such that $y = \mathbf{A}\mathbf{s}_0 + \mathbf{e}_0$ and $\|\mathbf{e}_0\| \leq B_P\sqrt{m}$ then return $(0, \mathbf{s}_0)$.
   - Otherwise, run the algorithm on $\mathbf{y} - \mathbf{u}$ to obtain $(\mathbf{s}_0, \mathbf{e}_0)$ and output $(1, \mathbf{s}_1)$

   (b) *Disjoint Injective Pair:* We have three cases here:

   - For $x \neq x'$, $\textsc{Supp}(g_{k,b(x)}) \cap \textsc{Supp}(g_{k,b}(x')) = \phi$ due to the correctness of the $\textsc{Invert}$ function of the trapdoor function.
   - $\textsc{Supp}(g_{k,0}(x)) \cap \textsc{Supp}(g_{k,1}(x)) = \phi$ due to the discussion about the length of $\mathbf{u}$ in the generation algorithm (see above). Also refer to Fig. 2 for a graphical visualization.
   - For $x \neq x'$, $\textsc{Supp}(g_{k,0}(x)) \cap \textsc{Supp}(g_{k,1}(x')) = \phi$ because of similar reasons as above.

3. **Efficient Range Superposition:** Use the same functions $\textsc{Chk}_{\mathcal{F}_{\text{LWE}}}$ and $\textsc{Samp}_{\mathcal{F}_{\text{LWE}}}$ as in the TCF family.

6.3. Injective Invariance

To show that $\mathcal{F}_{\text{LWE}}$ is injective invariant with respect to $\mathcal{G}_{\text{LWE}}$, we just need to show that for all QPT attackers $\mathcal{A}$, the distributions produced by $\textsc{Gen}_{\mathcal{F}_{\text{LWE}}}$ and $\textsc{Gen}_{\mathcal{G}_{\text{LWE}}}$ are computationally indistinguishable. This turns out to be equivalent to the hardness of LWE! This is because $\mathbf{u}$ is almost uniformly random as noted in the above construction.

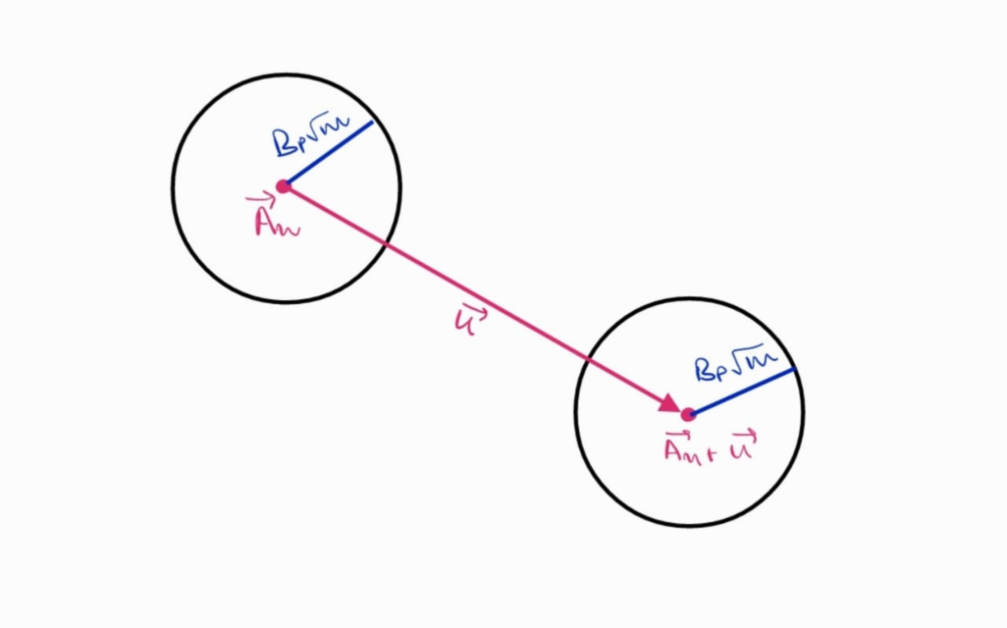


Figure 2: The length of **u** causes both the supports to be disjoint

Lemma 6.8. Assuming the choice of parameters **(A1)-(A6)** and hardness of $\mathrm{LWE}_{l,q,D_{\mathbb{Z}_q,BL}}$, the distributions

$$\mathcal{D}_0 = \{(\mathbf{A}, \mathbf{As} + \mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}_{\mathrm{LWE}}}(1^\lambda)\}$$
$$\mathcal{D}_1 = \{(\mathbf{A}, \mathbf{u}) \leftarrow \mathrm{GEN}_{\mathcal{G}_{\mathrm{LWE}}}(1^\lambda)\}$$

are computationally indistinguishable.

§7. Questions and Open problems

7.1. What are some other applications of TCFs?

- [CGV22] This paper explores **deniable encryption** in the context of quantum computing, aiming to create cryptographic systems where a user can plausibly deny the contents of their encrypted messages, even in the presence of quantum adversaries. Deniable encryption allows a sender to produce alternative "fake" plaintexts in case they are forced to reveal the encryption. The authors extend classical deniable encryption schemes to account for quantum threats, focusing on settings where both the sender and the receiver can manipulate information in a quantum way.

- [HMNY21] This paper addresses **quantum public key encryption (QPKE)** with a feature known as **certified deletion**, where a receiver can prove that they have deleted an encrypted message irreversibly. Certified deletion is significant in contexts where data sensitivity requires verifiable deletion (such as regulatory compliance or privacy mandates). The authors construct QPKE schemes that enable the sender to verify whether the receiver has deleted the decryption key, a functionality not achievable classically. The scheme relies on quantum no-cloning and measurement principles, enabling a novel form of data protection where the receiver has to measure (and therefore lose) certain information to demonstrate deletion. This work has implications for privacy and data management in quantum communication networks.

7.2. Can we construct trapdoor claw free functions without LWE?

There have been attempts to construct TCFs based on conjectured hard problems on isogeny-based group actions [AMR22]. This is the only construction we know of which is not based on lattice problems.

7.3. Can we obtain the adaptive hardcore bit properties from Ring LWE?

In [BKVV20], the authors present a proof-of-quantumness protocol that avoids relying on the adaptive hardcore bit (AHB1) property in the random oracle model. Their construction is based on the Ring Learning with Errors (Ring LWE) problem. However, a proof-of-quantumness protocol is considered *weaker* than qubit certification, which is achieved in [BCM$^+$21] using the AHB1 property. An open problem remains as to whether the AHB1 property can be derived from Ring LWE.

§8. Acknowledgments

We would like to thank Prof. Rajendra Kumar for regular discussions regarding this project. We would also thank Prof. Venkata Koppula and Dr. Mahesh Sreekumar Rajasree for fruitful discussions.

§A. Appendix

A.1. Proving that Lemma 6.3 $\implies$ Lemma 6.2

We prove this by contradiction. Assume that there exists a quantum polynomial-time procedure $\mathcal{A}$ such that

$$\left| \Pr_{(k,\mathrm{td})\leftarrow \mathrm{GEN}(1^\lambda)}[\mathcal{A}(k) \in H_s] - \Pr_{(k,\mathrm{td})\leftarrow \mathrm{GEN}(1^\lambda)}[\mathcal{A}(k) \in \bar{H}_s] \right| = \eta(\lambda) \, ,$$

where $\eta(\lambda)$ is some non-negligible function in the security parameter. We show that there exists a quantum polynomial-time adversary $\mathcal{A}'$ which uses $\mathcal{A}$ to distinguish between the two distributions D_0 and D_1 with some non-negligible probability. First consider the following observation

Claim A.1. For all $b \in \{0,1\}, x \in \mathcal{X}, d \in \{0,1\}^w$ and $\mathbf{s} \in \{0,1\}^m$, the following equality holds:

$$d \cdot (J(\mathbf{x}) \oplus J(\mathbf{x} - (-1)^b \mathbf{s})) = I_{b,x}(d) \cdot \mathbf{s}$$

Proof. Let $\mathbf{x} = (x_1, ..., x_n)$ be the individual entries of the vector $\mathbf{x}$. Similarly, let $J(\mathbf{x}) = (\alpha_1, \alpha_2, ...\alpha_n)$, where $\alpha_i \in \{0,1\}^{\lceil \log q \rceil}$ and let let $J(\mathbf{x} - (-1)^b \mathbf{1}) = (\alpha'_1, \alpha'_2, ...\alpha'_n)$. The i^{th} entry of the vector $I_{b,x}(d)$ is given by $(\alpha_i \oplus \alpha'_i) \cdot d_i$, where $d = (d_1, d_2, ..., d_n)$. Also let $J(\mathbf{x} - (-1)^b \mathbf{s}) = (\beta_1, \beta_2, ...\beta_n)$. The bit on the LHS is obtained by taking the sum $\sum_{i=1}^n (\alpha_i \oplus \beta_i).d_i$ and taking $\mod 2$ over this sum. Let $LHS_i = (\alpha_i \oplus \beta_i).d_i$ and in a similar fashion define $RHS_i = ((\alpha_i \oplus \alpha'_i) \cdot d_i \mod 2) \cdot s_i$ ($\mathbf{s} = (s_1, ..., s_n)$)

- **CASE 1:** $[s_i = 0]$ For this case, both LHS_i and RHS_i evaluate to 0. (This is because if $s_i = 0$, $\beta_i = \alpha_i$).
- **CASE 2:** $[s_i = 1]$ For this case, $\alpha'_i = \beta_i$ and thus again both sides become the same expression.

Hence each individual bit of inner product on both sides of LHS and RHS is the same value and thus taking sum mod 2 gives us the same value for both expressions. ■

Now we show the working of our distinguisher $\mathcal{A}'$. Let $\mathcal{A}'$ consist of two possible distinguishers, $\mathcal{A}'_u$ for $u \in \{0,1\}$, such that given a sample $w = \Big((\mathbf{A}, \mathbf{As} + \mathbf{e}), (b, x, d, c), t \Big)$, $\mathcal{A}'_u$ returns 0 if $c = t \oplus u$ and 1 otherwise. First consider the combined advantage of distinguishers $\mathcal{A}'_0$ and $\mathcal{A}'_1$,

$$\begin{aligned}
&\sum_{u \in \{0,1\}} \left| \Pr_{w \leftarrow D_0}[\mathcal{A}'_u(w) = 0] - \Pr_{w \leftarrow D_1}[\mathcal{A}'_u(w) = 0] \right| \\
&=^{[6]} \sum_{u \leftarrow \{0,1\}} \left| \Pr_{w \leftarrow D_0}[\mathcal{A}'_u(w) = 0 \;\;\&\;\; d \in \hat{G}_{s_{b\oplus 1}, b, x}] - \Pr_{w \leftarrow D_1}[\mathcal{A}'_u(w) = 0 \;\;\&\;\; d \in \hat{G}_{s_{b\oplus 1}, b, x}] \right| \\
&=^{[7]} \left| \Pr_{(\mathbf{A}, \mathbf{As}+\mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(\mathbf{A}, \mathbf{As} + \mathbf{e}) \in H_s] - \frac{1}{2} \Pr_{w \leftarrow D_1}[d \in \hat{G}_{s_{b\oplus 1}, b, x}] \right| \\
&+ \left| \Pr_{(\mathbf{A}, \mathbf{As}+\mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(\mathbf{A}, \mathbf{As} + \mathbf{e}) \in \bar{H}_s] - \frac{1}{2} \Pr_{w \leftarrow D_1}[d \in \hat{G}_{s_{b\oplus 1}, b, x}] \right| \\
&\geq^{[8]} \left| \Pr_{(\mathbf{A}, \mathbf{As}+\mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(\mathbf{A}, \mathbf{As} + \mathbf{e}) \in H_s] - \Pr_{(\mathbf{A}, \mathbf{As}+\mathbf{e}) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(\mathbf{A}, \mathbf{As} + \mathbf{e}) \notin H_s] \right| = \eta(1^\lambda),
\end{aligned}$$

Hence, atleast one of $\mathcal{A}'_0$ or $\mathcal{A}'_1$ must successfully distinguish between D_0 and D_1 with advantage atleast $\eta/2$, which contradicts Lemma 6.3.

[6] the first equality follows from the fact that if $d \notin \hat{G}_{s_{b\oplus 1}, b, x}$, then both distributions will be identical.

[7] Consider the case of $\mathcal{A}'_0$, it outputs 0 in case of $w \leftarrow D_0$ only if $\mathcal{A}$ outputs a tuple $(b, x, d, c) \in H_s$ (Note that the condition that $d \in \hat{G}_{s_{b\oplus 1}, b, x}$ is already enforced in the tuple belonging to H_s). Also, it outputs 0 in case of $w \leftarrow D_1$ with exactly half probability conditioned that $d \in \hat{G}_{s_{b\oplus 1}, b, x}$, since the bit t is completely random in this case. The other part can be similarly explained for $\mathcal{A}'_1$.

[8] Since $|a - b| \leq |a| + |b|$

A.2. Proof of Claim 6.7

We prove this by contradiction. Assume that there exists $\hat{d} \in \{0,1\}^n$ and a quantum polynomial-time procedure $\mathcal{A}$ such that

$$\left| \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e})\leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) \in H'_{s,J(\hat{d})}] - \frac{1}{2} \right| \geq \eta(1^{\lambda}),$$

where $\eta(.)$ is some non-negligible function. We derive a contradiction by showing that for $\hat{d}$, the two distributions D_0 and D_1 in Lemma 6.6 are computationally distinguishable, giving a contradiction. First consider the following claim:

Claim A.2. For all $\mathbf{x} \in \mathcal{X}, \hat{d} \in \{0,1\}^n$, and $\mathbf{s} \in \{0,1\}^n$, the following equality holds:

$$J(\hat{d}) \cdot (J(\mathbf{x}) \oplus J(\mathbf{x}-\mathbf{s}) = \hat{d} \cdot \mathbf{s}.$$

Proof. The proof is same as the proof of Claim A.1 in the case when bit $b = 0$. ■

Let $\mathcal{A}'$ be the adversary that uses $\mathcal{A}$ to distinguish between D_0 and D_1. Suppose that $\mathcal{A}'$ consists of two possible distinguishers $\mathcal{A}'_0$ and $\mathcal{A}'_1$. Given a sample $w = ((\mathbf{A}, \mathbf{As}+\mathbf{e}), t)$, $\mathcal{A}'_u$ computes $c = \mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e})$ and returns 0 if $c = t \oplus u$, and 1 otherwise. The combined advantage of distinguishers $\mathcal{A}'_0$ and $\mathcal{A}'_1$ is:

$$\begin{aligned}
&\sum_{u\in\{0,1\}} \left| \Pr_{((\mathbf{A},\mathbf{As}+\mathbf{e}),\, r)\leftarrow D_0}[\mathcal{A}'_u((\mathbf{A}, \mathbf{As}+\mathbf{e}),\, \hat{d}\cdot\mathbf{s}) = 0] - \Pr_{((\mathbf{A},\mathbf{As}+\mathbf{e}),\, r)\leftarrow D_1}[\mathcal{A}'_u((\mathbf{A}, \mathbf{As}+\mathbf{e}),\, r) = 0] \right| \\
&= \sum_{u\in\{0,1\}} \left| \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e})\leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) = \hat{d}\cdot\mathbf{s}\oplus u] - \Pr_{((\mathbf{A},\mathbf{As}+\mathbf{e}),\, r)\leftarrow D_1}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) = r\oplus u] \right| \\
&= \sum_{u\in\{0,1\}} \left| \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e})\leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) = \hat{d}\cdot\mathbf{s}\oplus u] - \frac{1}{2} \right| \\
&\geq \left| \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e})\leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) = \hat{d}\cdot\mathbf{s}] - \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e})\leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) = \hat{d}\cdot\mathbf{s}\oplus 1] \right| \\
&\geq 2\left| \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e})\leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) = \hat{d}\cdot\mathbf{s}] - \frac{1}{2} \right| \\
&= 2\left| \Pr_{(\mathbf{A},\mathbf{As}+\mathbf{e})\leftarrow \mathrm{GEN}_{\mathcal{F}}(1^{\lambda})}[\mathcal{A}(\mathbf{A}, \mathbf{As}+\mathbf{e}) \in H'_{s,J(\hat{d})}] - \frac{1}{2} \right| \\
&\geq 2\eta(1^{\lambda}).
\end{aligned}$$

Therefore, at least one of $\mathcal{A}'_0$ or $\mathcal{A}'_1$ must successfully distinguish between D_0 and D_1 with advantage atleast η, which completes our contradiction.

References

[ABOE08] Dorit Aharonov, Michael Ben-Or, and Elad Eban. Interactive proofs for quantum computations, 2008. URL: https://arxiv.org/abs/0810.5375, arXiv:0810.5375.

[ABOEM17] Dorit Aharonov, Michael Ben-Or, Elad Eban, and Urmila Mahadev. Interactive proofs for quantum computations, 2017. URL: https://arxiv.org/abs/1704.04487, arXiv:1704.04487.

[AKPW13] Joël Alwen, Stephan Krenn, Krzysztof Pietrzak, and Daniel Wichs. Learning with rounding, revisited. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013*, pages 57–74, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[AMR22] Navid Alamati, Giulio Malavolta, and Ahmadreza Rahimi. Candidate trapdoor claw-free functions from group actions with applications to quantum protocols. Cryptology ePrint Archive, Paper 2022/1775, 2022. URL: https://eprint.iacr.org/2022/1775.

[BCM$^+$21] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device, 2021. URL: https://arxiv.org/abs/1804.00640, arXiv:1804.00640.

[BFK09] Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. Universal blind quantum computation. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, October 2009. URL: http://dx.doi.org/10.1109/FOCS.2009.36, doi:10.1109/focs.2009.36.

[BKVV20] Zvika Brakerski, Venkata Koppula, Umesh Vazirani, and Thomas Vidick. Simpler proofs of quantumness, 2020. URL: https://arxiv.org/abs/2005.04826, arXiv:2005.04826.

[BV97] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997. arXiv:https://doi.org/10.1137/S0097539796300921, doi:10.1137/S0097539796300921.

[CGV22] Andrea Coladangelo, Shafi Goldwasser, and Umesh Vazirani. Deniable encryption in a quantum world, 2022. URL: https://arxiv.org/abs/2112.14988, arXiv:2112.14988.

[FK17] Joseph F. Fitzsimons and Elham Kashefi. Unconditionally verifiable blind quantum computation. *Physical Review A*, 96(1), July 2017. URL: http://dx.doi.org/10.1103/PhysRevA.96.012303, doi:10.1103/physreva.96.012303.

[HMNY21] Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. *Quantum Encryption with Certified Deletion, Revisited: Public Key, Attribute-Based, and Classical Communication*, page 606–636. Springer International Publishing, 2021. URL: http://dx.doi.org/10.1007/978-3-030-92062-3_21, doi:10.1007/978-3-030-92062-3_21.

[Mah23] Urmila Mahadev. Classical verification of quantum computations, 2023. arXiv:1804.01082.

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

[PRSD17] Chris Peikert, Oded Regev, and Noah Stephens-Davidowitz. Pseudorandomness of ring-LWE for any ring and modulus. Cryptology ePrint Archive, Paper 2017/258, 2017. URL: https://eprint.iacr.org/2017/258.

[Reg24] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography, 2024. URL: https://arxiv.org/abs/2401.03703, arXiv:2401.03703.

[RUV12] Ben W. Reichardt, Falk Unger, and Umesh Vazirani. A classical leash for a quantum system: Command of quantum systems via rigidity of chsh games, 2012. URL: https://arxiv.org/abs/1209.0448, arXiv:1209.0448.

[Sha92] Adi Shamir. Ip = pspace. *J. ACM*, 39(4):869–877, October 1992. doi:10.1145/146585.146609.

[Vid20] Thomas Vidick. Interactive proofs with quantum devices, 2020. URL: http://users.cms.caltech.edu/~vidick/teaching/fsmp/.