Quantum Universally Composable Oblivious Linear Evaluation

Manuel B. Santos[1,2], Paulo Mateus[1,2], and Chrysoula Vlachou[1,2]

[1]Instituto de Telecomunicações, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal

[2]Departamento de Matemática, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal

Oblivious linear evaluation is a generalization of oblivious transfer, whereby two distrustful parties obliviously compute a linear function, $f(x) = ax + b$, i.e., each one provides their inputs that remain unknown to the other, in order to compute the output $f(x)$ that only one of them receives. From both a structural and a security point of view, oblivious linear evaluation is fundamental for arithmetic-based secure multi-party computation protocols. In the classical case, oblivious linear evaluation protocols can be generated using oblivious transfer, and their quantum counterparts can, in principle, be constructed as straightforward extensions using quantum oblivious transfer. Here, we present the first, to the best of our knowledge, quantum protocol for oblivious linear evaluation that, furthermore, does not rely on quantum oblivious transfer. We start by presenting a semi-honest protocol and then extend it to the dishonest setting employing a *commit-and-open* strategy. Our protocol uses high-dimensional quantum states to obliviously compute $f(x)$ on Galois Fields of prime and prime-power dimension. These constructions utilize the existence of a complete set of mutually unbiased bases in prime-power dimension Hilbert spaces and their linear behaviour upon the Heisenberg-Weyl operators. We also generalize our protocol to achieve vector oblivious linear evaluation, where several instances of oblivious linear evaluation are generated, thus making the protocol more efficient. We prove the protocols to have static security in the framework of quantum universal composability.

1 Introduction

Oblivious Linear Evaluation (OLE) is a cryptographic task that permits two distrustful parties, say Alice and Bob, to jointly compute the output of a linear function $f(x) = ax+b$ in some finite field, $\mathbb{F}$. Alice provides inputs $a, b \in \mathbb{F}$ and Bob provides $x \in \mathbb{F}$, while the output, $f(x)$, becomes available only to Bob. As the parties are distrustful, a secure OLE protocol should not permit Alice to learn anything about Bob's input, while also Alice's inputs should remain unknown to Bob. OLE can be seen as a generalization of Oblivious Transfer (OT) [66], a basic primitive for secure two-party computation, which, in turn, is a special case of secure multi-party computation [18, 26, 40]. OT has been shown to be

Manuel B. Santos: manuel.batalha.santos@gmail.com

Paulo Mateus: pmat@tecnico.ulisboa.pt

Chrysoula Vlachou: chrysoula.vlachou@tecnico.ulisboa.pt

complete for secure multi-party computation [48], i.e., any such task, including OLE, can be achieved given an OT implementation. A compelling reason to study OLE protocols is that they can serve as building blocks for the secure evaluation of arithmetic circuits [4, 32, 35, 38], just like OT allows the secure evaluation of boolean circuits [41]. Specifically, OLE can be used to generate multiplication triples which are the basic tool for securely computing multiplication gates [32]. Besides that, OLE has applications in more tasks for two-party secure computation [5, 14, 25, 44, 46] and Private Set Intersection [37].

Impagliazzo and Rudich proved that OT protocols require public-key cryptography and cannot just rely on symmetric cryptography [45]. Consequently, OLE cannot rely on symmetric cryptography either, and we need to resort to public-key cryptography. However, Shor's quantum algorithm [70] poses a threat to the currently deployed public-key systems, motivating the search for protocols secure against quantum attacks. Bennet et al. [9] and Crépeau [28] proposed the first protocols for Quantum OT (QOT). The no-go theorems by Lo and Chau [53, 54] and, independently, by Mayer [57], implied that an unconditionally secure QOT without any additional assumption is impossible. A decade later, Damgård et al. proved the security of QOT in the stand-alone model under additional assumptions [29], and Unruh [76] showed that it is statistically secure and universally composable with access only to ideal commitments. As far as quantum OLE (QOLE) is concerned, to the best of our knowledge, no protocol has been proposed as of now. Analogously to the classical case it is expected that one can implement QOLE based on QOT protocols. That said, in this work we propose a protocol for QOLE that, additionally, does not rely on any QOT implementation, and as such, it might provide efficiency advantages for various tasks that can be achieved directly with OLE, e.g. the evaluation of arithmetic circuits.

OLE is commonly generalized to Vector OLE (VOLE). In this setting, Alice defines a set of k linear functions $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{F}^k \times \mathbb{F}^k$ and Bob receives the evaluation of all these functions on a specified element $x \in \mathbb{F}$, i.e. $\boldsymbol{f} := \boldsymbol{a}x + \boldsymbol{b}$. One can think of VOLE as the arithmetic analogue of string OT and show how it can be used in certain Secure Arithmetic Computation and Non-Interactive Zero Knowledge proofs [14]. Ghosh et. al put further in evidence the usefulness of VOLE by showing that it serves as the building block of Oblivious Polynomial Evaluation [38], a primitive which allows more sophisticated applications, such as password authentication, secure list intersection, anonymous complaint boxes [62], anonymous initialization for secure metering of client visits in servers [61], secure Taylor approximation of relevant functions (e.g. logarithm) [52], secure set intersection [43] and distributed generation of RSA keys [39]. We also show how our QOLE protocol can be adapted to achieve secure VOLE.

1.1 Contributions

We present a quantum protocol for OLE with quantum universally composable security (Definition 2.7) in the $\mathcal{F}_{\mathbf{COM}}-$hybrid model, i.e. when assuming the existence of a commitment functionality, $\mathcal{F}_{\mathbf{COM}}$ (Figure 3). To obtain a secure protocol, we take advantage of the properties of Mutually Unbiased Bases (MUBs) in high-dimensional Hilbert spaces with prime and prime-power dimension. Such a choice is motivated by recent theoretical and experimental advances that pave the way for the development and realization of new solutions for quantum cryptography [1, 2, 12, 13, 22, 31, 34, 69, 71, 79]. To the best of our knowledge, our protocol is the first proposal of a QOLE protocol which we prove to be quantum-UC secure. Moreover, it is not based on any QOT implementation, according to the standard approach. To prove its security, the only assumption we make is the existence of a classical commitment functionality. We consider the static corruption adversarial model with both semi-honest and dishonest adversaries (Section 2.3). Finally, we

modify the proposed protocol to generate quantum-UC secure VOLE.

Main tool. The proposed protocol $\pi_{\mathbf{QOLE}}$ (Figure 8) is based on the fact that in a Hilbert space of dimension d (isomorphic to $\mathbb{Z}_d$) there exists a set of MUBs $\{|e_r^x\rangle\}_{x,r\in\mathbb{Z}_d}$, such that, upon the action of a certain operator V_a^b, each basis element r is shifted by some linear factor $ax-b$ inside the same basis x:

$$V_a^b \left|e_r^x\right\rangle = c_{a,b,x,r} \left|e_{ax-b+r}^x\right\rangle, \tag{1}$$

where $a,b,x,r \in \mathbb{Z}_d = \{0,1,\ldots,d-1\}$. If Alice controls the operator V_a^b and Bob controls the quantum state $|e_r^x\rangle$, they are able to compute a linear function $f(x) = ax-b$ where effectively Alice controls the function $f=(a,b)$ and Bob controls its input x. Moreover, since Bob controls x and r, he can receive $f(x)$ by measuring the output element.

Protocol overview. In a nutshell, $\pi_{\mathbf{QOLE}}$ (Figure 8) with inputs $f=(a,b)$ from Alice and x from Bob is divided into two main phases. In the first *quantum phase*, Alice and Bob use high-dimensional quantum states to generate n random weak OLE (RWOLE) instances, where n is the security parameter. In this phase, Alice outputs n random elements $f_i^0 = (a_i^0, b_i^0)$, and Bob outputs n elements $(x_i^0, y^0 = f_i^0(x_i^0))$. These instances are considered to be weaker because Bob is allowed to have some amount of information about the n outputs of Alice (a_i^0, b_i^0). In the second *post-processing phase*, Alice and Bob use classical tools to extract one secure OLE from the aforementioned n instances.

More specifically, in the quantum phase, Bob randomly generates $m=(1+t)n$ quantum states $\left|e_{r_i}^{x_i^0}\right\rangle$ and sends them to Alice. Then, Bob commits to his choice (x_i^0, r_i), $\forall i \in [m]$, where for any $l \in \mathbb{N}$, $[l]$ denotes the set $\{1,\ldots,l\}$, using an ideal commitment functionality, $\mathcal{F}_{\mathbf{COM}}$, and Alice asks to verify a subset T of size tn of these commitments. This intermediate *commit-and-open* step allows Alice to test Bob's behaviour and ensure that he does not deviate *too much* from the protocol, and it is a common method used in security proofs of QOT protocols [29, 76]. If Bob passes all the tests, Alice randomly generates (a_i^0, b_i^0) and applies $V_{a_i^0}^{b_i^0}$ to the remaining n received states $\left|e_{r_i}^{x_i^0}\right\rangle$, for $i \in [m] \setminus T$. For the rest of this section we relabel and denote $[n] = [m] \setminus T$. According to the expression (1), the output states are given by $\left|e_{a_i^0 x_i^0 - b_i^0 + r_i}^{x_i^0}\right\rangle$ and she sends them to Bob, who outputs $y_i^0 = a_i^0 x_i^0 - b_i^0$ by measuring the received states in the corresponding basis x_i^0 and subtracting r_i, $\forall i \in [n]$.

The post-processing phase uses two subprotocols: a derandomization step (Figure 6) and an extraction step (Figure 7). The derandomization step is based on the protocol π_{OLE}^n from [30] and transforms the n RWOLE instances into n weak OLE (WOLE) instances with inputs $(a_i, b_i)_{i\in[n]}$ chosen by Alice and inputs x_i for $i \in [n]$ chosen by Bob. The extraction protocol uses the so-called *Multi-linear Modular Hashing* family, $\mathfrak{G}$, of two-universal hash functions [42] to render Bob's information on Alice's system useless and to extract one secure OLE out of n instances of WOLE. In the extraction phase, Alice samples a two-universal hash function $g_{\boldsymbol{\kappa}}$ from $\mathfrak{G}$ and sends it to Bob. Then, with adequately-crafted vectors $(\boldsymbol{a}, \boldsymbol{b}) = ((a_1,\ldots,a_n),(b_1,\ldots,b_n))$, Alice has $a = g_{\boldsymbol{\kappa}}(\boldsymbol{a})$ and $b = g_{\boldsymbol{\kappa}}(\boldsymbol{b})$, and Bob outputs $y = g_{\boldsymbol{\kappa}}(\boldsymbol{y})$, where $\boldsymbol{y} = \boldsymbol{a}\boldsymbol{x} + \boldsymbol{b}$ after point-wise vector multiplication with the constant vector $\boldsymbol{x} = (x,\ldots,x)$.

Quantum-UC security. Due to the quantum nature of the states $\left|e_{r_i}^{x_i^0}\right\rangle_{i\in[n]}$, a dishonest Alice is not able to distinguish which bases $x_i^0, i \in [n]$ are used by Bob. From her point

of view, Bob's states are maximally mixed and therefore completely hide x_i^0. This is enough to ensure that, in the derandomization step, Alice does not receive any information about Bob's final input x. For a dishonest Bob, to correctly pass all Alice's tests, it means he did not cheat at all rounds with overwhelming probability. This ensures that he has some *bounded* information on Alice's random elements $(a_i^0, b_i^0)_{i\in[n]}$, and using privacy amplification techniques in the extraction step, Alice can guarantee that Bob's information about her final input (a, b) is the same as in the case of an ideal OLE functionality, i.e. the probability distribution of a is close to uniform.

Turning this intuition into a quantum-UC security proof requires some additional insights. First, we need a way to quantify Bob's information on Alice's elements (a_i^0, b_i^0) after the testing phase and the application of the corresponding $V_{a_i^0}^{b_i^0}$ operators, for $i \in [n]$; for this purpose we use the *min-entropy* (Definition 4). We follow the approach of [29] to guarantee that Bob does not significantly deviate from the protocol in all the rounds, and we use Theorem 1 from [33] to compute a concrete lower bound of Bob's min-entropy on Alice elements $(a_i^0, b_i^0)_{i\in[n]}$. Along with Lemma 2.2, we have that $a = g_{\boldsymbol{\kappa}}(\boldsymbol{a})$ is close to uniform, which is sufficient to prove that Bob does not know more about (a, b) than what the output $y = ax + b$ reveals.

In order to show that $\pi_{\mathbf{QOLE}}$ is quantum-UC secure, we need to show that an ideal execution of $\pi_{\mathbf{QOLE}}$ with access to $\mathcal{F}_{\mathbf{OLE}}$ (Figure 1) is indistinguishable from a real execution of the protocol from the point of view of an external entity called the *environment*. To prove this indistinguishability, we have to build a simulator that simulates the execution of the protocol in the ideal setting and generates messages on behalf of the honest simulated parties, while trying to extract the dishonest party's inputs and feed them in $\mathcal{F}_{\mathbf{OLE}}$. In particular, for a dishonest Alice, we have to demonstrate the existence of a simulator, $\mathcal{S}_A$, that generates messages on behalf of honest Bob and extracts Alice's input (a, b) which, in turn, feeds into $\mathcal{F}_{\mathbf{OLE}}$. To this end, we consider that $\mathcal{S}_A$ simulates an attack by Bob at all rounds, i, of the protocol which allows to extract the m values of Alice (a_i^0, b_i^0). However, the commit-and-open scheme described above is designed to catch such an attack, and to work around this issue we substitute the ideal commitment functionality, $\mathcal{F}_{\mathbf{COM}}$, with a fake commitment functionality, $\mathcal{F}_{\mathbf{FakeCOM}}$, that allows $\mathcal{S}_A$ to open the commitments later [76]. From the remaining n values (a_i^0, b_i^0), $\mathcal{S}_A$ computes Alice's input (a, b) and feeds it to $\mathcal{F}_{\mathbf{OLE}}$.

For a dishonest Bob, we have to show the existence of a simulator, $\mathcal{S}_B$, that generates messages on behalf of honest Alice and extracts Bob's input x. We assume that $\mathcal{S}_B$ has full control over $\mathcal{F}_{\mathbf{COM}}$, which means that it has access to Bob's m committed values (x_i^0, r_i); the input x can be easily extracted from these values.

Protocol generalizations. We start by generalizing the main relation (24) to Galois Fields of prime-power dimension, $GF(d^M)$ for $M > 1$. Then, we show how we can obtain a protocol for quantum VOLE. In particular, from n WOLE instances, we are able to generate a VOLE with size proportional to n, and we bound this proportion by the min-entropy value on the WOLE instances.

1.2 Organization

In Section 2, we introduce notation and state results and definitions, that we will use in the rest of the paper. In Section 3, in order to build some intuition, we present a QOLE protocol that is secure only if we consider Bob to be semi-honest; in case Bob is dishonest, its security is compromised. In Section 4, we construct our secure QOLE protocol, $\pi_{\mathbf{QOLE}}$,

and in Section 5, we prove its security in the quantum-UC framework. Finally, in the last Section 6, we show how to generalize the presented QOLE protocol to Galois Fields of prime-power dimensions and we also present a quantum-UC secure protocol achieving VOLE. In the last Section 7, we give an outlook of our results and point out directions of future work.

2 Preliminaries

The elements of a set, $\mathcal{X}$, are denoted by lower case letters, x, while capital letters, X, are used for random variables with probability distribution, P_X, over the set $\mathcal{X}$. For a prime number, d, the set $\mathbb{Z}_d = \{0, ..., d-1\}$ denotes the finite field with dimension d. Vectors $\boldsymbol{v} \in \mathbb{Z}_d^n$, for any $n \in \mathbb{N}$ are denoted in lower case bold letters, while capital bold letters are used for vectors of random variables. Also, we denote the uniform distribution over $\mathcal{X}$ as $\tau_{\mathcal{X}} = \frac{1}{|\mathcal{X}|}\sum_{x\in\mathcal{X}} |x\rangle\langle x|$, where $|\mathcal{X}|$ is the size of $\mathcal{X}$. We write as $x \leftarrow_{\$} \mathcal{X}$ the operation of assigning to x an element uniformly chosen from a set $\mathcal{X}$. Finally, $\mathbb{1}$ is the identity operator.

We consider Hilbert spaces, $\mathcal{H}$, of dimension $d \geq 2$, where d is a prime number. Each Hilbert space has an orthonormal basis $\{|x\rangle : x \in \mathbb{Z}_d\}$, called the computational basis. Given different quantum systems $\mathcal{H}_1, \ldots, \mathcal{H}_n$, we use the tensor product notation to describe the joint system as $\mathcal{H}_1 \otimes \ldots \otimes \mathcal{H}_n$. The vectors in this joint system are denoted by $|\boldsymbol{x}\rangle = |x_1\rangle \otimes \ldots \otimes |x_n\rangle$, where $\boldsymbol{x} \in \mathbb{Z}_d^n$. Quantum states are, in general, represented as positive semi-definite operators with unitary trace acting on a Hilbert space $\mathcal{H}$, and we denote the respective set as $\mathcal{P}(\mathcal{H})$. They are called mixed states, as opposed to pure quantum states that are represented by vectors in $\mathcal{H}$.

Let $\rho, \sigma \in \mathcal{P}(\mathcal{H})$ be two quantum states. Then, the *trace distance* between them is defined as

$$\delta(\rho, \sigma) := \frac{1}{2}||\rho - \sigma||_1, \tag{2}$$

where $||\cdot||_1$ is the $1-$Schatten norm in the space of bounded operators acting on $\mathcal{H}$.

Physically possible quantum operations are described by Completely Positive Trace Preserving (CPTP) maps. By definition, CPTP maps preserve the normalization of quantum states (being Trace Preserving) and map positive operators to positive operators (being Completely Positive), ensuring that density operators are mapped to density operators.

As we mentioned before, we use the min-entropy, $H_{\min}$, to quantify the amount of information that Bob might obtain about Alice's system, A, given some (possibly quantum) side information encoded in a system B', and which he can obtain by means of an attack.

Definition 2.1. Let $\rho_{AB'} \in \mathcal{P}(\mathcal{H}_A \otimes \mathcal{H}_{B'})$ and $\sigma_{B'} \in \mathcal{P}(\mathcal{H}_{B'})$. The *min-entropy* of $\rho_{AB'}$ relative to $\sigma_{B'}$ is given by

$$H_{\min}(A|B')_{\rho|\sigma} = -\log \min\{\lambda : \lambda \cdot \mathrm{id}_A \otimes \sigma_{B'} \geq \rho_{AB'}\}, \tag{3}$$

and

$$H_{\min}(A|B')_{\rho} = \sup_{\sigma_{B'}} H_{\min}(A|B')_{\rho|\sigma}. \tag{4}$$

Throughout our analysis we will also use the so-called *d-ary entropy function*, the generalization of the standard binary entropy function:

Definition 2.2. For $d \geq 2$, the *d-ary entropy function* $h_d : [0, 1] \to \mathbb{R}$ is given by

$$h_d(x) = x \log_d(d-1) - x \log_d x - (1-x)\log_d(1-x). \tag{5}$$

2.1 Mutually Unbiased Bases in $\mathcal{H}^d$

In this section, we present the basics and some properties of MUBs in $\mathcal{H}^d$. This is the main tool that is used in our protocol. For more details about MUBs, see [34].

Definition 2.3. Let $\mathcal{B}_0 = \{|\psi_1\rangle, \ldots, |\psi_d\rangle\}$ and $\mathcal{B}_1 = \{|\phi_1\rangle, \ldots, |\phi_d\rangle\}$ be orthonormal bases in the d-dimensional Hilbert space $\mathcal{H}^d$. They are said to be *mutually unbiased* if $|\langle\psi_i|\phi_j\rangle| = \frac{1}{\sqrt{d}}$ for all $i, j \in \{1, \ldots, d\}$. Furthermore, a set $\{\mathcal{B}_0, \ldots, \mathcal{B}_m\}$ of orthonormal bases on $\mathcal{H}^d$ is said to be a set of MUBs if, for every $i \neq j$, $\mathcal{B}_i$ is mutually unbiased with $\mathcal{B}_j$.

MUBs are extensively used in quantum cryptography because, in some sense, these bases are as far as possible from each other and the overlap between two elements from different bases is constant. Let $\{|0\rangle, \ldots, |d-1\rangle\}$ be the computational basis of $\mathcal{H}^d$, where d is a prime number, and $\{|\tilde{0}\rangle, \ldots, |\widetilde{d-1}\rangle\}$ be the dual basis which is given by the Fourier transform on the computational basis:

$$\left|\tilde{j}\right\rangle = \frac{1}{\sqrt{d}} \sum_{i=0}^{d-1} \omega^{-ij} |i\rangle , \tag{6}$$

where $\omega = e^{\frac{2\pi i}{d}}$. We can easily verify that the computational basis and its dual basis are mutually unbiased, and we will make use of the following two operators, V_a^0 and V_0^b, to encode Alice's functions during the first (quantum) phase of the protocol.

Definition 2.4 (Shift operators)**.** The *shift operator* V_a^0 shifts the computational basis by a elements, i.e.

$$V_a^0 |i\rangle = |i + a\rangle . \tag{7}$$

Similarly, the *dual shift operator* V_0^b shifts the dual basis by b elements, i.e.

$$V_0^b \left|\tilde{j}\right\rangle = \left|\widetilde{j - b}\right\rangle . \tag{8}$$

The operators V_a^0 and V_0^b are diagonal in the dual and computational basis, respectively[1], i.e.

$$V_a^0 = \sum_{j=0}^{d-1} \omega^{aj} \left|\tilde{j}\right\rangle\!\left\langle\tilde{j}\right| \text{ and } V_0^b = \sum_{i=0}^{d-1} \omega^{bi} |i\rangle\langle i| . \tag{9}$$

Furthermore, we can define

$$V_a^b := V_0^b V_a^0 = \sum_{l=0}^{d-1} \omega^{(l+a)b} |l + a\rangle\langle l| , \tag{10}$$

obtaining the so-called *Heisenberg-Weyl operators*. These operators form a group of unitary transformations with d^2 elements; the group has $d + 1$ commuting abelian subgroups of d elements, and for each abelian subgroup, there exists a basis of joint eigenstates of all V_a^b in the subgroup. These $d + 1$ bases are pairwise mutually unbiased. Let $x \in \mathbb{Z}_{d+1}$ label the abelian subgroups, let $l \in \mathbb{Z}_d$ label the elements of each subgroup, and let U_l^x denote

[1]Note that V_a^0 and V_0^b can be seen as a generalization of the Pauli X and Z operators, respectively.

the corresponding subgroup operators. Finally, let the $i-$th basis element associated with the $x-$th subgroup be denoted by $|e_i^x\rangle$. Then, it can be seen that [34],

$$U_l^x = \sum_{i=0}^{d-1} \omega^{il} \, |e_i^x\rangle\langle e_i^x| \,, \tag{11}$$

and

$$|e_i^x\rangle = \frac{1}{\sqrt{d}} \sum_{l=0}^{d-1} \omega^{-xl(d-l)/2-il} \, |l\rangle \,, \tag{12}$$

where

$$U_l^x = \alpha_l^x V_l^{xl} \text{ with } \alpha_l^x = \omega^{-xl(l+d)/2}. \tag{13}$$

One can show that

$$V_a^b \, |e_0^x\rangle = c_{x,a,b} \, |e_{ax-b}^x\rangle \,, \; x \in \mathbb{Z}_d, \tag{14}$$

and

$$V_a^b \left|e_0^d\right\rangle = c_{d,a,b} \left|e_a^d\right\rangle \text{ for } x = d, \tag{15}$$

or more generally

$$V_a^b \, |e_r^x\rangle = c_{a,b,x,r} \, |e_{ax-b+r}^x\rangle \text{ with } c_{a,b,x,r} = \omega^{a(r+x-b)+\frac{a(a-1)}{2}x}. \tag{16}$$

This last property is the main ingredient for the construction of our protocol as it encodes a linear evaluation based on values a, b and $x \in \mathbb{Z}_d$[2]. In our protocol, we take a, b – that determine the operators V_a^b – to be Alice's inputs and x to be Bob's input.

Finally, let us see how the operators V_a^b act on the so-called *generalized Bell states*, since Bob's attack to the protocol is based on that. We start with the definition of the *seed* Bell state

$$|B_{0,0}\rangle = \frac{1}{\sqrt{d}} \sum_i |i^*, i\rangle \,, \tag{17}$$

where the map $|\psi\rangle \to |\psi^*\rangle$ is defined by taking the complex conjugate of the coefficients:

$$|\psi\rangle = \sum_i \beta_i \, |i\rangle \to |\psi^*\rangle = \sum_i \beta_i^* \, |i\rangle \,. \tag{18}$$

Using the properties of the operators V_a^b, we can derive the rest of the generalized Bell states from the seed state, as

$$|B_{a,b}\rangle = (\mathbb{1} \otimes V_a^b) \, |B_{0,0}\rangle = \frac{1}{\sqrt{d}} \sum_{i=0}^{d-1} \omega^{(i+a)b} \, |i^*, i+a\rangle \,, \tag{19}$$

and one can prove that the set $\{|B_{a,b}\rangle\}_{(a,b)\in\mathbb{Z}_d^2}$ constitutes an orthonormal maximally entangled basis in the Hilbert space of two-qudit states [34].

[2]While $x \in \mathbb{Z}_{d+1}$, henceforth we consider $x \in \mathbb{Z}_d$, since we only use d out of the $d+1$ MUBs.

2.2 Two-Universal Functions

During the post-processing phase of the protocol, in order to amplify the privacy of Alice's inputs and outputs against the information leakage to Bob, we will use the so-called *Multi-linear Modular Hashing* family of functions:

Definition 2.5 (Definition 2, [42])**.** Let d be a prime and let n be an integer $n > 0$. We define the family, $\mathfrak{G}$, of functions from $\mathbb{Z}_d^n$ to $\mathbb{Z}_d$, as

$$\mathfrak{G} := \{g_{\boldsymbol{\kappa}} : \mathbb{Z}_d^n \to \mathbb{Z}_d \,|\, \boldsymbol{\kappa} \in \mathbb{Z}_d^n\}, \tag{20}$$

where the functions $g_{\boldsymbol{\kappa}}$ are defined for any $\boldsymbol{\kappa} = (\kappa_1, \ldots, \kappa_n) \in \mathbb{Z}_d^n$ and $x = (x_1, \ldots, x_n)$ as

$$g_{\boldsymbol{\kappa}}(\boldsymbol{x}) = \boldsymbol{\kappa} \cdot \boldsymbol{x} \mod d = \sum_{i=1}^{n} \kappa_i \, x_i \mod d. \tag{21}$$

These functions are linear, since they are based on the modular inner product of vectors, therefore they preserve the structure of the OLE regarding the inputs and outputs, while they amplify the privacy of Alice's elements. For the $\mathfrak{G}$ family one can prove the following theorem [42]:

Theorem 2.1. The family $\mathfrak{G}$ is two-universal (see Definition 2.6)[3].

Definition 2.6 (Two-universal hash family)**.** A family, $\mathfrak{G}$, of functions, g, with domain D and range R is called a *two-universal hash family* if for any two distinct elements $w, w' \in D$ and for g chosen at random from $\mathfrak{G}$, the probability of a *collision* $g(w) = g(w')$ is at most $1/|R|$, where $|R|$ is the size of the range R.

Next, the *Generalized Leftover Hash Lemma* is a relevant ingredient in our proof of security, as it ensures that, after applying a known function g from a two-universal family to a random variable X, the resulting random variable $Z = g(X)$ is close to uniform, conditioned on some (possibly quantum) side information E.

Lemma 2.2 (Generalized Leftover Hash Lemma [74][4])**.** Let X be a random variable, E a quantum system, and $\mathfrak{G}$ a two-universal family of hash functions from X to $\mathbb{Z}_d^l$. Then, on average over the choices of g from $\mathfrak{G}$, the output $Z := g(X)$ is ξ-close to uniform conditioned on E, where

$$\xi = \frac{1}{2}\sqrt{2^{l \log d - H_{\min}(X|E)}}. \tag{22}$$

[3]In fact, Halevi and Krawczyk [42] proved a stronger result, namely that the $\mathfrak{G}$ family is $\Delta-$*universal*, which is more general than two-universal.

[4]This is a high-dimensional version of the Generalized Leftover Hash Lemma, that can be easily deduced by using Lemma 4 from [74] with $d_A = d^l$. Note also that this is a special version, since Tomamichel et al. in [74] prove it in the more general case for $\delta-$almost two-universal hash families.

2.3 Quantum Universal Composability

The Universal Composability (UC) framework was introduced by Canetti [19] in the classical setting and extended to the quantum setting by Unruh, and Ben-Or and Mayers [8, 75] (see also [36, 76]). It renders strong composability guarantees as it ensures that the security of the protocol is independent of any external execution of the same or other protocols. Both classical and quantum frameworks follow the same ideal-real world comparison structure and consider the same interactions between machines. The difference lies on the operations allowed by the quantum-UC framework, where we are also allowed to store, transmit, and process quantum states.

More specifically, the quantum-UC security of some protocol π is drawn by comparing two scenarios. A real scenario where π is executed and an ideal scenario where an ideal functionality, $\mathcal{F}$, that carries out the same task and is defined *a priori*, is executed. The comparison is done by a special machine, called the *environment*, $\mathcal{Z}$, that supervises the execution of both scenarios and has access to any external information (e.g. concurrent executions of the same or any other protocol). In the two-party case that we are considering, the structure of the machines present in both scenarios is as follows: The real scenario has the environment, $\mathcal{Z}$, the adversary, $\mathcal{A}$, and the two parties Alice and Bob, while the ideal scenario has the environment $\mathcal{Z}$, the simulator $\mathcal{S}$, the two parties Alice and Bob and the ideal functionality $\mathcal{F}$. Informally, we say that the protocol π is quantum-UC secure if no environment $\mathcal{Z}$ can distinguish between the execution of π in the real scenario and the execution of the functionality $\mathcal{F}$ in the ideal scenario. Any possible attack of the adversary $\mathcal{A}$ in the execution of π can be simulated by the simulator $\mathcal{S}$ in the ideal-world execution of $\mathcal{F}$, without any noticeable difference form the point-of-view of the environment $\mathcal{Z}$. Since the ideal functionality $\mathcal{F}$ is secure by definition, the real-world adversary is not able to extract any more information than what is allowed by the $\mathcal{F}$.

Let us now see the formal definition. Let π and ρ be the real and ideal two-party protocols. We denote by $\mathrm{Ex}_{\pi^C,\mathcal{A},\mathcal{Z}}$ (analogously $\mathrm{Ex}_{\rho^C,\mathcal{S},\mathcal{Z}}$) the output of the environment $\mathcal{Z}$ at the end of the real (ideal) execution, and by C the corrupted party.

Definition 2.7 (Quantum-UC security [76])**.** Let protocols π and ρ be given. We say that π **statistically** quantum-UC emulates ρ if and only if for every party, C, and for every adversary, $\mathcal{A}$, there exists a simulator, $\mathcal{S}$, such that for every environment $\mathcal{Z}$, and every $z \in \{0,1\}^*$, $n \in \mathbb{N}$

$$\left|\Pr[\mathrm{Ex}_{\pi^C,\mathcal{A},\mathcal{Z}}(n,z)=1] - \Pr[\mathrm{Ex}_{\rho^C,\mathcal{S},\mathcal{Z}}(n,z)=1]\right| \leq \mu(n), \tag{23}$$

where $\mu(n)$ is a negligible function, i.e., for every positive polynomial, $p(\cdot)$, there exists an integer $N_p > 0$ such that for all $n > N_p$ we have $\mu(n) < 1/p(n)$, and n is the security parameter. We furthermore require that if $\mathcal{A}$ is quantum-polynomial-time, so is $\mathcal{S}$. Finally, if we consider quantum-polynomial-time $\mathcal{A}$ and $\mathcal{Z}$ we have **computational** quantum-UC security.

The role of the simulator, $\mathcal{S}$, is to simulate the execution of the protocol π in such a way that the environment $\mathcal{Z}$ is not able to distinguish between the two executions. In particular, since $\mathcal{S}$ does not have access to the inputs/outputs of the actual honest party, it runs a simulated honest party interacting with the environment which is acting as the adversary. Moreover, $\mathcal{S}$ controls the dishonest party, therefore controlling their inputs to the ideal functionality $\mathcal{F}$. It then relies on its ability to extract the inputs that the environment provides to the dishonest party. Then, it uses them along with the ideal functionality outputs in order to successfully generate a simulated execution that cannot be distinguished by the environment.

Finally, note that, in case the real execution of the protocol π makes use of some external functionality $\mathcal{F}_{\mathbf{ext}}$, the simulator $\mathcal{S}$ can control and reprogram $\mathcal{F}_{\mathbf{ext}}$ in whatever way suits best in the ideal world to produce an indistinguishable simulation of the real world.

Regarding the adversarial model, we consider both *semi-honest* and *dishonest* adversaries. Semi-honest adversaries (also called honest-but-curious or passive adversaries) do not deviate from the protocol and only try to passively gain extra information by looking at the exchanged messages. Dishonest adversaries may deviate arbitrarily from the protocol. We also adopt the static corruption adversarial model where the corruption of each party is done just before the execution of the protocol.

Ideal functionalities. The main functionality is the OLE functionality, $\mathcal{F}_{\mathbf{OLE}}$, and it is described in Figure 1.

$\mathcal{F}_{\mathbf{OLE}}$ functionality

Alice's input: $(a, b) \in \mathbb{Z}_d^2$; **Bob's input:** $x \in \mathbb{Z}_d$

- Upon receiving input (a, b) from Alice and x from Bob, the functionality sends $f(x) := ax + b$ to Bob.

Alice's output: $\perp$; **Bob's output:** $f(x)$

Figure 1: OLE functionality definition.

In Figure 2, we present the functionality, $\mathcal{F}^n_{\mathbf{VOLE}}$, which is similar to the former with the difference being that its inputs and outputs are vectors.

$\mathcal{F}^n_{\mathbf{VOLE}}$ functionality

Alice's input: $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{Z}_d^{2n}$; **Bob's input:** $x \in \mathbb{Z}_d$

- Upon receiving input $(\boldsymbol{a}, \boldsymbol{b})$ from Alice and x from Bob, the functionality sends $\boldsymbol{f}(x) := \boldsymbol{a}x + \boldsymbol{b} = (a_1x + b_1, \ldots, a_nx + b_n)$ to Bob.

Alice's output: $\perp$; **Bob's output:** $\boldsymbol{f}(x)$

Figure 2: VOLE functionality definition.

As we mentioned already, we work in the so-called $\mathcal{F}_{\mathbf{COM}}-$ hybrid model, where the real execution of the protocol has access to an ideal commitment functionality, $\mathcal{F}_{\mathbf{COM}}$, defined in Figure 3. Note that our protocol $\pi_{\mathbf{QOLE}}$ makes several calls to $\mathcal{F}_{\mathbf{COM}}$ and only opens a subset of the committed elements, therefore we use an index element i to specify different instance calls. In the commitment phase, Bob sends $(\texttt{commit}, i, M)$ to the functionality and the functionality sends $(\texttt{commit}, i)$ to Alice. In the opening phase, Bob sends $(\texttt{open}, i)$ and the functionality sends $(\texttt{open}, i, M)$ to Alice.

The $\mathcal{F}_{\mathbf{COM}}$ functionality can be replaced by the commitment protocol $\pi_{\mathbf{COM}}$ presented in [20] which is computationally UC-secure in the Common Reference String (CRS) model. As shown in [27] (Theorem 3.), the protocol $\pi_{\mathbf{COM}}$ computationally quantum-UC realizes

$\mathcal{F}_{\mathbf{COM}}$ functionality

Committer message: M

- *Commitment phase.* Upon receiving $(\texttt{commit}, M)$ from Bob, the functionality sends `commit` to Alice.
- *Opening phase.* Upon receiving `open` from Bob, the functionality sends $(\texttt{open}, M)$ to Alice.

Figure 3: Commitment functionality definition.

$\mathcal{F}_{\mathbf{COM}}$ in the CRS model[5]. As a result, the resulting protocol $\pi_{\mathbf{QOLE}}^{\pi_{\mathbf{COM}}}$ is quantum-UC secure.

3 Semi-honest QOLE protocol

In order to build some intuition on the proposed protocol for QOLE, we start by presenting a simpler protocol (Figure 4) that is only secure under the semi-honest adversarial model.

Semi-honest QOLE

Alice's input: $(a, b) \in \mathbb{Z}_d^2$; **Bob's input:** $x \in \mathbb{Z}_d$

1. Bob randomly generates $r \in \mathbb{Z}_d$. He prepares and sends the state $|e_r^x\rangle$ to Alice.
2. Alice prepares the operator V_a^b according to her inputs a and b. She then applies V_a^b to Bob's state: $V_a^b |e_r^x\rangle = c_{x,a,b,r} \left|e_{ax-b+r}^x\right\rangle$. She sends the resulting state back to Bob.
3. Bob measures in the basis x, subtracts r, and outputs the desired result $ax-b =: f(x)$.

Alice's output: $\perp$; **Bob's output:** $f(x)$

Figure 4: Semi-honest QOLE protocol.

This semi-honest version leverages the properties of MUBs explored in Section 2.1 and, in particular, the one presented in (16). As we saw, given the set of MUBs $\{|e_r^x\rangle\}_{r\in\mathbb{Z}_d}$, $\forall x \in \mathbb{Z}_d$, the operators V_a^b simply permute the elements inside the basis x, according to a linear combination of the elements a, b, x and r:

$$V_a^b |e_r^x\rangle = c_{a,b,x,r} |e_{ax-b+r}^x\rangle . \tag{24}$$

Alice and Bob can use the above property to compute together a linear function $f(x) = ax - b$, where Alice chooses the parameters a and b, and Bob chooses the input element

[5]The computational quantum UC-security of $\pi_{\mathbf{COM}}$, shown in [27], follows by lifting its classical computational UC-security (proved in [20]) to the quantum setting, assuming that the pseudorandom number generator in $\pi_{\mathbf{COM}}$, is robust against quantum adversaries.

x. Bob starts by choosing a basis x and an element r therein, and prepares the state $|e_r^x\rangle$: the basis choice x plays the role of the input element x, and the basis element r is used to enhance Bob's security against a potentially dishonest Alice. Then, he sends the state $|e_r^x\rangle$ to Alice, who, in turn, applies on it the operator V_a^b and sends back to Bob the resulting state. According to (24), Bob receives $\left|e^x_{ax-b+r}\right\rangle$, measures it in the x basis, and outputs the linear function evaluation $f(x) = ax - b$ by subtracting r. Thus, the correctness of the protocol is ensured by (24).

As far as the security of this protocol is concerned, we can easily see that it is secure against a dishonest Alice. From her point of view, all the density matrices describing the several possible cases for $x = 0, \ldots, d-1$ are maximally mixed states. Therefore, she cannot know anything about the value of x. If, moreover, Bob is semi-honest the protocol remains secure.

On the other hand, if Bob is dishonest and deviates from the protocol, he is able to find out Alice's inputs a and b with certainty. In (19), we saw that the generalized Bell basis is generated by Alice's operators, V_a^b, i.e. $|B_{a,b}\rangle = (1 \otimes V_a^b)\,|B_{0,0}\rangle$, and Bob can make use of this property in order to extract her inputs a and b. His attack can be described as follows:

1. Bob prepares the state $|B_{0,0}\rangle$ and sends the second qudit to Alice.
2. Alice applies her chosen operator V_a^b.
3. Bob measures both qudits in the generalized Bell basis and outputs a, b.

It becomes clear that the protocol is secure only as long as Bob does not deviate from it; a dishonest Bob can break its security by performing the above attack. Therefore, we have to make sure that Bob sticks to the protocol. To achieve this, we apply a *commit-and-open* scheme [29] that can be briefly described as follows: Bob runs step 1. of the semi-honest QOLE protocol (Figure 4) multiple times, say m in total, for multiple values of x_i, and r_i, for $i \in [m]$ and commits to these values by means of the functionality $\mathcal{F}_{\mathbf{COM}}$ (Figure 3). Then, he sends these states to Alice, who, in turn, asks him to disclose his chosen x_i's and r_i's for some of the m instances that she chooses. $\mathcal{F}_{\mathbf{COM}}$ forwards these committed values to Alice and she measures the corresponding received states in the disclosed bases. She can, thus, verify whether she got the right basis element for all the instances she chose to check. If Bob had used the Bell state $|B_{0,0}\rangle$ in one out of the m instances, then the probability of Alice getting the correct result after measuring the state in the committed basis would be $\frac{1}{d}$. In other words, Bob would get caught with high probability $1 - \frac{1}{d}$. Furthermore, if he chooses to attack all the instances, the probability of Alice getting correctly all the results is negligible, i.e. exponentially small in the number of instances, m.

4 QOLE protocol

Our QOLE protocol is divided in two phases: a quantum phase and a classical post-processing phase. The first phase uses quantum communication to generate several instances of OLE with random inputs. These instances may leak some information to the parties, therefore we refer to them as random weak OLE (RWOLE). The second phase is purely classical. It uses the RWOLE instances and extracts one classical OLE instance. The post-processing phase has two phases. It implements a derandomization procedure

followed by an extraction phase that serves as a privacy amplification method. Before proceeding, it is worth mentioning that we consider that neither dishonest party maliciously aborts the protocol. Indeed, in our setting, such a behaviour does not provide an advantage for learning the other party's input. The only case to abort the protocol is when honest Alice catches Bob cheating during the *commit-and-open* stage.

In what follows, we break down the protocol, show its correctness and retrieve some technical lemmas used for the security proof against static dishonest adversaries.

Notation. During the RWOLE phase, $\mathbf{F}_0 = (F_1^0, F_2^0, \dots, F_n^0)$ is the vector whose components are the random variables associated to Alice's functions. Each F_i^0 ranges over the set of affine functions in $\mathbb{Z}_d$ such that $\Pr(F_i^0(x) = a_i^0 x + b_i^0)$ is uniform for all $i \in [n]$. We do not distinguish the set of affine functions in $\mathbb{Z}_d$ from $\mathbb{Z}_d^2$. The classical values $\mathbf{F}_0$ are saved in the Hilbert space $\mathcal{H}_{\mathbf{F}_0}$. The same holds for the derandomization phase, where $\mathbf{F}$ denotes the random variable for Alice's functions in the protocol $\pi^n_{\mathbf{WOLE}}$. $\mathbf{X}_0$ and $\mathbf{Y}_0$ are the random variables for $\mathbf{x}_0, \mathbf{y}_0 \in \mathbb{Z}_d^n$ in the RWOLE phase. and $\mathbf{X}$ and $\mathbf{Y}$ the corresponding random variables for $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_d^n$ in the post-processing phase. Also, we use A' and B' to denote the system that a dishonest Alice and Bob, respectively, hold at the end of the execution of the protocol.

4.1 RWOLE phase

We now introduce the quantum phase of the proposed QOLE protocol, which we informally call the random weak OLE (RWOLE) phase. We denote by π^n_{RWOLE} the protocol that implements this RWOLE phase and we present it in Figure 5.

If both parties are honest the protocol is correct: if Alice is honest, her functions $\mathbf{F}_0$ are chosen uniformly at random, and if Bob is honest he will obtain $\left| e^{x_i^0}_{a_i^0 x_i^0 - b_i^0 + r_i} \right\rangle_{i \in [n]}$ according to (24).

Security. In the case of a dishonest Alice, it is straightforward to verify that the security property of the semi-honest protocol still holds, following the same reasoning. In the case of dishonest Bob, though, these OLE random instances might leak information on Alice's random functions $\mathbf{F}_0$ to him. To quantify this side information, we must bound $H_{\min}(\mathbf{F}_0|B')_{\rho_{\mathbf{F}_0 B'}}$, where $\rho_{\mathbf{F}_0 B'}$ is the output state of the real execution of $\pi^n_{\mathbf{RWOLE}}$. The following lemma shows that $\rho_{\mathbf{F}_0 B'}$ is at least $\epsilon-$close to an ideal state $\sigma_{\mathbf{F}_0 B'}$ independently of the attack that the dishonest party may perform. This ideal state $\sigma_{\mathbf{F}_0 B'}$ has a bound on $H_{\min}(\mathbf{F}_0|B')_{\sigma_{\mathbf{F}_0 B'}}$ that is proportional to the security parameter.

Lemma 4.1 (Security against dishonest Bob)**.** Let $\rho_{\mathbf{F}_0 B'}$ be the state given by the real execution of the protocol $\pi^n_{\mathbf{RWOLE}}$, where $\mathbf{F}_0$ is the system saving Alice's functions, B' is Bob's (possibly quantum) system. Fix $\zeta \in]0, 1 - \frac{1}{d}]$ and let

$$\epsilon(\zeta, n) = \exp\left(-\frac{2\zeta^2 t^2 n^2}{(nt+1)(t+1)} \right). \tag{25}$$

Then, for any attack of a dishonest Bob, there exists an ideal classical-quantum state $\sigma_{\mathbf{F}_0 B'}$, such that

1. $\sigma_{\mathbf{F}_0 B'} \approx_\epsilon \rho_{\mathbf{F}_0 B'}$,
2. $H_{\min}(\mathbf{F}_0|B')_{\sigma_{\mathbf{F}_0 B'}} \geq \frac{n \log d}{2}(1 - h_d(\zeta))$,

where $h_d(\zeta)$ is given in Definition 2.2.

Protocol $\pi^n_{\mathbf{RWOLE}}$

Parameters: n, number of output qudits; t, proportion of receiver test qudits.

(Initialization Phase:)

1. Bob randomly generates $m = (1+t)n$ different pairs (x_i^0, r_i) and commits to them by sending $(\texttt{commit}, (i, x_i^0, r_i))$ to $\mathcal{F}_{\mathbf{COM}}$. He prepares the states $\left|e_{r_i}^{x_i^0}\right\rangle_{i\in[m]}$ and sends them to Alice.

(Test Phase:)

2. Alice randomly chooses a subset of indices $T \subset [m]$ of size tn and sends it to Bob.

3. Bob sends $(\texttt{open}, i)$, $i \in T$, to $\mathcal{F}_{\mathbf{COM}}$ and $\mathcal{F}_{\mathbf{COM}}$ sends to Alice $(\texttt{open}, (i, x_i^0, r_i))$, $i \in T$.

4. Alice measures the received qudits in the corresponding x_i^0 basis for $i \in T$, and checks whether the received commitments are compatible with her measurements. In case there is no error she proceeds, otherwise she aborts. After the Test Phase, we relabel and identify $[n] = [m] \setminus T$.

(Computation Phase:)

6. Alice randomly generates n pairs (a_i^0, b_i^0) and prepares $V_{a_i^0}^{b_i^0}$ for $i \in [n]$.

7. Alice applies these operators to the received states, i.e. $V_{a_i^0}^{b_i^0}\left|e_{r_i}^{x_i^0}\right\rangle = c_{x_i^0,a_i^0,b_i^0,r_i}\left|e_{a_i^0 x_i^0 - b_i^0 + r_i}^{x_i^0}\right\rangle$, for $i \in [n]$, and sends the resulting states to Bob.

(Measurement Phase:)

8. Bob measures the received states in the basis x_i^0 for $i \in [n]$ and gets the states $\left|e_{a_i^0 x_i^0 - b_i^0 + r_i}^{x_i^0}\right\rangle, i \in [n]$. Finally, he subtracts r_i, for $i \in [n]$ from his results.

Alice's output: (a_i^0, b_i^0), for $i \in [n]$.
Bob's output: (x_i^0, y_i^0), where $y_i^0 = g_i(x_i^0) = a_i^0 x_i^0 - b_i^0$ for $i \in [n]$.

Figure 5: RWOLE protocol.

The proof comprises two parts corresponding to the two conditions of Lemma 4.1: first, we prove that the state just before the *Computation Phase* is close to the ideal state $\sigma_{\mathbf{F}_0 B'}$; and then, we prove that the operators applied by Alice to $\sigma_{\mathbf{F}_0 B'}$ increase the min-entropy by a specific amount that is proportional to the number of output qudits, n. We present the proof in Appendix A.1, where we follow the same reasoning as Damgård et al. in Section 4.3 of [29], and adapt it to our case. We also use certain results from [33] to establish the lower bound of property 2.

4.2 Post-processing phase

The $\pi^n_{\mathbf{RWOLE}}$ protocol (Figure 5) generates several instances of RWOLE, which leak information to Bob about Alice's inputs. In this section, we present the post-processing phase that allows to extract one secure QOLE out of several RWOLE instances. Combining them is sufficient to generate a secure QOLE protocol, because Bob has only a negligible probability of attacking *all* the weak instances without being caught; indeed, if he chooses to attack one of them the probability of Alice not aborting is $\frac{1}{t+1} + \frac{t}{d(1+t)}$, while if he chooses to attack all of them this probability becomes $\frac{1}{d^{tn}}$, which is negligible in n, thus ensuring the asymptotic security of our protocol. The post-processing comprises two subprotocols: the first is a derandomization protocol (Figure 6) that integrates the randomized outputs of RWOLE into a deterministic scheme where Alice and Bob choose their inputs; the second is an extraction protocol (Figure 7) that generates a secure QOLE protocol from these deterministic weak instances by means of a two-universal family of hash functions. Note that the classical post-processing phase does not give any advantage to a potentially dishonest Alice, therefore we only need to prove security against dishonest Bob.

4.2.1 Derandomization

Our derandomization protocol, denoted as $\pi^n_{\mathbf{WOLE}}$ and summarized in Figure 6, transforms the random RWOLE instances into deterministic ones, which we informally call weak OLE (WOLE). The output of $\pi^n_{\mathbf{WOLE}}$ is still a weak version of OLE because Bob is allowed to have knowledge on Alice's inputs. The difference between RWOLE and WOLE is that the parties now choose their inputs. Our derandomization protocol is an adaptation of the protocol in [30]. We denote by $*$ the product of two matrices of the same dimensions, such that the result is also a matrix of the same dimensions whose elements are the product of the respective elements of the operand matrices.

Protocol $\pi^n_{\mathbf{WOLE}}$

Alice's input: $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{Z}_d^{2n}$; **Bob's input:** $\boldsymbol{x} \in \mathbb{Z}_d^n$

1. Alice and Bob run the $\pi^n_{\mathbf{RWOLE}}$ protocol and receive $(\boldsymbol{a}_0, \boldsymbol{b}_0)$ and $(\boldsymbol{x}_0, \boldsymbol{y}_0)$, respectively.
2. Bob computes and sends to Alice $\boldsymbol{c} = \boldsymbol{x} - \boldsymbol{x}_0$.
3. Alice computes and sends to Bob $\boldsymbol{d} = \boldsymbol{a} - \boldsymbol{a}_0$ and $\boldsymbol{s} = \boldsymbol{b}_0 + \boldsymbol{a} * \boldsymbol{c} + \boldsymbol{b}$.
4. Bob computes $\boldsymbol{y} = \boldsymbol{y}_0 + \boldsymbol{x} * \boldsymbol{d} - \boldsymbol{d} * \boldsymbol{c} + \boldsymbol{s}$.

Alice's output: $\perp$; **Bob's output:** $\boldsymbol{y} = \boldsymbol{a} * \boldsymbol{x} + \boldsymbol{b}$

Figure 6: WOLE protocol.

Security. The requirements to prove security against dishonest Bob are summarized in Lemma 4.2, which is very similar in structure to Lemma 4.1. We show that the real output state $\rho_{\mathbf{F}B'}$ of the protocol $\pi^n_{\mathbf{WOLE}}$ is $\epsilon-$close to an ideal state $\sigma_{\mathbf{F}B'}$, which has min-entropy lower-bounded by a fixed value proportional to the security parameter n. Intuitively, this means that Bob's state is indistinguishable from a state where his knowledge on Alice's

inputs is limited.

Lemma 4.2. Let $\rho_{\mathbf{F}B'}$ be the state given by the real execution of the protocol $\pi^n_{\mathbf{WOLE}}$, where $\mathbf{F}$ is the system saving Alice's inputs, B' is Bob's (possibly quantum) system. Fix $\zeta \in]0, 1 - \frac{1}{d}]$ and let

$$\epsilon(\zeta, n) = \exp\left(-\frac{2\zeta^2 t^2 n^2}{(nt+1)(t+1)}\right). \tag{26}$$

Then, for any attack of a dishonest Bob, there exists a classical-quantum state $\sigma_{\mathbf{F}B'}$ such that

1. $\sigma_{\mathbf{F}B'} \approx_\epsilon \rho_{\mathbf{F}B'}$,
2. $H_{\min}(\mathbf{F}|B')_{\sigma_{\mathbf{F}B'}} \geq \frac{n \log d}{2}(1 - h_d(\zeta))$,

where $h_d(\zeta)$ is given in Definition 2.2.

Proof. Alice holds the system $A = \mathbf{F}\mathbf{F}_0\mathbf{CDS}$, where $\mathbf{F} = (\mathbf{F}_{\boldsymbol{a}}, \mathbf{F}_{\boldsymbol{b}})$ refers to her inputs $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{Z}_d^{2n}$, $\mathbf{F}_0 = (\mathbf{F}_{\boldsymbol{a}_0}, \mathbf{F}_{\boldsymbol{b}_0})$ is the subsystem obtained from the RWOLE phase, and $\mathbf{C}, \mathbf{D}$ and $\mathbf{S}$ are classical subsystems used to save the values of $\boldsymbol{c}$, $\boldsymbol{d}$, and $\boldsymbol{s}$ from the protocol, respectively. Bob holds the system $B' = \mathbf{CDS}B'_0$ where $\mathbf{C}, \mathbf{D}$ and $\mathbf{S}$ are the subsystems on Bob's side where the values of $\boldsymbol{c}$, $\boldsymbol{d}$ and $\boldsymbol{s}$ are saved, respectively, and $B'_0 = \mathbf{Y}_0 E_0$ is his (possibly quantum) system generated from the RWOLE phase.

To prove property 1., we will use Lemma 4.1, namely that the state $\rho_{\mathbf{F}_0 B'_0}$ resulting from the RWOLE scheme is $\epsilon-$close to the ideal state $\sigma_{\mathbf{F}_0 B'_0}$. Then, we will show that the operations applied to $\rho_{\mathbf{F}_0 B'_0}$ during the derandomization process can only decrease the distance between the real and the ideal output states of the WOLE protocol, thus keeping them at least $\epsilon-$close. We start by specifying the operators corresponding to the classical operations executed in steps 2 and 3 of $\pi^n_{\mathbf{WOLE}}$. In step 2, a dishonest Bob can send to Alice some value $\boldsymbol{c}$ that depends on his system B'_0. So, he starts by applying a CPTP map $\mathcal{T}_{B'_0 \to \mathbf{C}B'_0} : \mathcal{P}\left(\mathcal{H}_{B'_0}\right) \to \mathcal{P}\left(\mathcal{H}_{B'_0} \otimes \mathcal{H}_{\mathbf{C}}\right)$ to his state and then projects it into the Hilbert space $\mathcal{H}_{\mathbf{C}}$. The operator for step 2 is a CPTP map

$$\mathcal{O}^{(2)} : \mathcal{P}\left(\mathcal{H}_{\mathbf{F}_0} \otimes \mathcal{H}_{B'_0}\right) \to \mathcal{P}\left(\mathcal{H}_{\mathbf{F}_0} \otimes \mathcal{H}_{\mathbf{C}} \otimes \mathcal{H}_{\mathbf{D}} \otimes \mathcal{H}_{\mathbf{S}} \otimes \mathcal{H}_{B'_0}\right) \tag{27}$$

described by his action on some general quantum state ρ, as

$$\mathcal{O}^{(2)}(\rho) = \mathbb{1} \otimes \sum_{\boldsymbol{d},\boldsymbol{s},\boldsymbol{c}} |\boldsymbol{c}\rangle\langle\boldsymbol{c}|_{\mathbf{C}}\, \mathcal{T}_{B'_0 \to \mathbf{C}B'_0}(\rho)\, |\boldsymbol{c}\rangle\langle\boldsymbol{c}|_{\mathbf{C}} \otimes |\boldsymbol{d}\rangle\langle\boldsymbol{d}|_{\mathbf{D}} \otimes |\boldsymbol{s}\rangle\langle\boldsymbol{s}|_{\mathbf{S}}\,. \tag{28}$$

In step 3, Bob takes no action. Since Alice is honest, the operator for this step simply describes her action on subsystems $\mathbf{D}$ and $\mathbf{S}$ according to her choice at subsystem $\mathbf{F}$. This operator is a CPTP map

$$\mathcal{O}^{(3)} : \mathcal{P}\left(\mathcal{H}_{\mathbf{F}_0} \otimes \mathcal{H}_{\mathbf{C}} \otimes \mathcal{H}_{\mathbf{D}} \otimes \mathcal{H}_{\mathbf{S}} \otimes \mathcal{H}_{B'_0}\right) \to \mathcal{P}\left(\mathcal{H}_{\mathbf{F}} \otimes \mathcal{H}_{\mathbf{F}_0} \otimes \mathcal{H}_{\mathbf{C}} \otimes \mathcal{H}_{\mathbf{D}} \otimes \mathcal{H}_{\mathbf{S}} \otimes \mathcal{H}_{B'_0}\right) \tag{29}$$

described by his action on some general quantum state ρ, as

$$\mathcal{O}^{(3)}(\rho) = \frac{1}{d^{2n}} \sum_{\boldsymbol{a},\boldsymbol{b}} \mathcal{P}^{\boldsymbol{a},\boldsymbol{b}}\, \rho\, (\mathcal{P}^{\boldsymbol{a},\boldsymbol{b}})^\dagger, \tag{30}$$

where

$$\mathcal{P}^{\boldsymbol{a},\boldsymbol{b}} = |\boldsymbol{a},\boldsymbol{b}\rangle_{\mathbf{F}} \otimes \sum_{\boldsymbol{a}_0,\boldsymbol{b}_0,\boldsymbol{c}} |\boldsymbol{a}_0,\boldsymbol{b}_0\rangle\langle\boldsymbol{a}_0,\boldsymbol{b}_0|_{\mathbf{F}_0} \otimes |\boldsymbol{c}\rangle\langle\boldsymbol{c}|_{\mathbf{C}} \otimes |\boldsymbol{a}-\boldsymbol{a}_0\rangle\langle\boldsymbol{a}-\boldsymbol{a}_0|_{\mathbf{D}} \otimes |\boldsymbol{b}_0+\boldsymbol{a}\cdot\boldsymbol{c}+\boldsymbol{b}\rangle\langle\boldsymbol{b}_0+\boldsymbol{a}\cdot\boldsymbol{c}+\boldsymbol{b}|_{\mathbf{S}}\,. \tag{31}$$

Note that $\mathcal{O}^{(2)}$ adds subsystems **CDS** and distributes **C** according to Bob's action. The operator $\mathcal{O}^{(3)}$ adds subsystem **F** and projects **DS** according to the information at subsystem $\mathbf{FF}_0$ and the expressions of $\boldsymbol{d}$ and $\boldsymbol{s}$. Regarding the trace distance between the real and ideal states, we have:

$$\begin{aligned}\delta(\rho_{\mathbf{F}_0B'_0},\sigma_{\mathbf{F}_0B'_0}) &\geq \delta\Big(\mathcal{O}^{(2)}(\rho_{\mathbf{F}_0B'_0}),\mathcal{O}^{(2)}(\sigma_{\mathbf{F}_0B'_0})\Big)\\ &\geq \delta\Big(\mathcal{O}^{(3)}\mathcal{O}^{(2)}(\rho_{\mathbf{F}_0B'_0}),\mathcal{O}^{(3)}\mathcal{O}^{(2)}(\sigma_{\mathbf{F}_0B'_0})\Big)\\ &= \delta(\rho_{\mathbf{F}B'},\sigma_{\mathbf{F}B'}).\end{aligned} \tag{32}$$

For the above inequalities, we took into account that $\mathcal{O}^{(2)}$ and $\mathcal{O}^{(3)}$ are CPTP maps, and as such they do not increase the trace distance (see Lemma 7 in [78]). For the last equality, recall that $B' = \boldsymbol{CDS}B'_0$. Now, from Lemma 4.1, we have that $\sigma_{\mathbf{F}_0B'_0} \approx_\epsilon \rho_{\mathbf{F}_0B'_0}$. Hence, we conclude that $\delta(\rho_{\mathbf{F}B'},\sigma_{\mathbf{F}B'}) \leq \epsilon(\zeta,n)$ for $\epsilon(\zeta,n)$ given as (26), i.e. $\sigma_{\mathbf{F}B'} \approx_\epsilon \rho_{\mathbf{F}B'}$.

We move on to prove property 2. Consider the bijective function $g^{\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}} : \mathbb{Z}_d^{2n} \to \mathbb{Z}_d^{2n}$ given by

$$g^{\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}}(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x}+\boldsymbol{d},\boldsymbol{s}-\boldsymbol{y}-(\boldsymbol{x}+\boldsymbol{d})*\boldsymbol{c}), \tag{33}$$

for fixed $\boldsymbol{c},\boldsymbol{d}$ and $\boldsymbol{s}$. Essentially, $g^{\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}}$ describes how the input vector $(\boldsymbol{a},\boldsymbol{b})$ is related to the RWOLE output vector $(\boldsymbol{a}_0,\boldsymbol{b}_0)$:

$$(\boldsymbol{a},\boldsymbol{b}) = g^{\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}}(\boldsymbol{a}_0,\boldsymbol{b}_0) = (\boldsymbol{a}_0+\boldsymbol{d},\boldsymbol{s}-\boldsymbol{b}_0-(\boldsymbol{a}_0+\boldsymbol{d})*\boldsymbol{c}). \tag{34}$$

Intuitively, this means that the subsystem **F** is defined by the subsystems $\mathbf{F}_0\mathbf{CDS}$. We can rewrite the action of the operator $\mathcal{O}^{(3)}$ on some general quantum state ρ as follows:

$$\mathcal{O}^{(3)}(\rho) = \frac{1}{d^{2n}}\sum_{\boldsymbol{d},\boldsymbol{s}} \mathcal{P}^{\boldsymbol{d},\boldsymbol{s}}\,\rho\,(\mathcal{P}^{\boldsymbol{d},\boldsymbol{s}})^\dagger, \tag{35}$$

where

$$\mathcal{P}^{\boldsymbol{d},\boldsymbol{s}} = \sum_{\boldsymbol{a}_0,\boldsymbol{b}_0,\boldsymbol{c}} \Big|g^{\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}}(\boldsymbol{a}_0,\boldsymbol{b}_0)\Big\rangle\Big\langle\boldsymbol{a}_0,\boldsymbol{b}_0\Big|_{\mathbf{F}} \otimes |\boldsymbol{a}_0,\boldsymbol{b}_0,\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}\rangle\langle\boldsymbol{a}_0,\boldsymbol{b}_0,\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}|_{\mathbf{F}_0\boldsymbol{CDS}}\,. \tag{36}$$

Hence, for the min-entropy bound, we have:

$$\begin{aligned}H_{\min}(\mathbf{F}\,|\,B')_{\sigma_{\mathbf{F}B'}} &= H_{\min}(\mathbf{F}_{\boldsymbol{a}},\mathbf{F}_{\boldsymbol{b}}\,|\,B')_{\sigma_{\mathbf{F}B'}}\\ &= H_{\min}(f^{\boldsymbol{C},\boldsymbol{D},\boldsymbol{S}}(\mathbf{F}_{\boldsymbol{a}_0},\mathbf{F}_{\boldsymbol{b}_0})\,|\,\mathbf{CDS}B'_0)_{\mathcal{O}^{(3)}\mathcal{O}^{(2)}\sigma_{\mathbf{F}_0B'_0}}\\ &\geq H_{\min}(\mathbf{F}_{\boldsymbol{a}_0},\mathbf{F}_{\boldsymbol{b}_0}\,|\,\mathbf{CDS}B'_0)_{\mathcal{O}^{(2)}\sigma_{\mathbf{F}_0B'_0}} &(37)\\ &\geq H_{\min}(\mathbf{F}_{\boldsymbol{a}_0},\mathbf{F}_{\boldsymbol{b}_0}\,|\,B'_0)_{\sigma_{\mathbf{F}_0B'_0}} &(38)\\ &\geq \frac{n\log d}{2}(1-h_d(\zeta)). &(39)\end{aligned}$$

The inequality (37) comes from Lemma B.1, as $g^{\boldsymbol{c},\boldsymbol{d},\boldsymbol{s}}$ is bijective. The inequality (38) comes from Theorem 6.19 in [73], as the operator $\mathcal{O}^{(2)}$ takes the form of $\mathcal{O}^{(2)} = \mathbb{1}\otimes\mathcal{M}$, where $\mathcal{M}$ is a CPTP map. The last inequality comes from Lemma 4.1, property 2. □

4.2.2 Extraction

In this section, we present our extraction protocol, $\pi_{\mathbf{EXT}}$, that generates one OLE instance using the derandomization protocol $\pi^n_{\mathbf{WOLE}}$ and the two-universal family of hash functions, $\mathfrak{G}$ (see [42] and Definition 2.5). This family uses the inner product between two vectors in the $\mathbb{Z}_d^n$ space, and since OLE only involves linear operations, we can apply the inner product operation to all vectors $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{y}$ without affecting the overall structure. The protocol $\pi_{\mathbf{EXT}}$ is summarized in Figure 7 and uses the n instances of WOLE in such a way that Bob's knowledge on Alice's inputs decreases exponentially with respect to the security parameter n[6].

Protocol $\pi_{\mathbf{EXT}}$

Alice's input: $(a, b) \in \mathbb{Z}_d^2$; **Bob's input:** $x \in \mathbb{Z}_d$

1. Alice chooses randomly some function $g_{\boldsymbol{\kappa}} \in \mathfrak{G}$ and sends it to Bob.
2. Alice randomly generates $a_2, \ldots, a_n, b_2, \ldots, b_n \leftarrow_{\$} \mathbb{Z}_d$. She computes $a_1 = (a - \sum_{i=2}^n a_i\kappa_i)/\kappa_1$ and $b_1 = (b - \sum_{i=2}^n b_i\kappa_i)/\kappa_1$. We write $\boldsymbol{a} = (a_1, \ldots, a_n)$ and $\boldsymbol{b} = (b_1, \ldots, b_n)$.
3. Alice and Bob run the derandomization protocol $\pi^n_{\mathbf{WOLE}}((\boldsymbol{a}, \boldsymbol{b}), \boldsymbol{x})$. Bob receives $\boldsymbol{y}$ as output.
4. Bob computes $y = g_{\boldsymbol{\kappa}}(\boldsymbol{y})$.

Alice's output: $\perp$; **Bob's output:** y

Figure 7: Extraction protocol.

The correctness of $\pi_{\mathbf{EXT}}$ follows from linearity:

$$\begin{aligned} y = g_{\boldsymbol{\kappa}}(\boldsymbol{y}) &= \boldsymbol{\kappa} \cdot (a_1 x + b_1, \ldots a_n x + b_n) \\ &= \boldsymbol{\kappa} \cdot \left(\frac{a - \sum_{i=2}^n a_i\kappa_i}{\kappa_1} x + \frac{b - \sum_{i=2}^n b_i\kappa_i}{\kappa_1},\ a_2 x + b_2, \ldots, a_n x + b_n \right) \\ &= ax + b. \end{aligned} \tag{40}$$

Security. By definition, the derandomization protocol leaks some information to Bob about Alice's inputs $(\boldsymbol{a}, \boldsymbol{b})$. Since $\boldsymbol{y} = \boldsymbol{a} * \boldsymbol{x} + \boldsymbol{b}$, without loss of generality, any leakage of Alice's inputs can be seen as a leakage on just $\boldsymbol{a}$. In this case, the min-entropy of $\mathbf{F} = (\mathbf{F}_{\boldsymbol{a}}, \mathbf{F}_{\boldsymbol{b}})$ should be the same as the min-entropy of $\mathbf{F}_{\boldsymbol{a}}$. Now, recall the $\mathcal{F}_{\mathbf{OLE}}$ definition (Figure 1), and note that Bob does not possess any knowledge about Alice's input (a, b) other than what can be deduced from his input and output (x, y). Similarly, since $y = ax + b$, Bob has some knowledge on the relation between a and b and – as b is completely determined by (a, x, y) – we only have to guarantee that a looks uniformly random to Bob. The role of the hash functions used in the above protocol $\pi_{\mathbf{EXT}}$ is precisely to extract a uniformly random a from the leaky vector $\boldsymbol{a}$, while preserving the structure of the OLE. This result is summarized in Lemma 4.3 and its proof is based on Lemma 2.2.

[6] This extraction step is similar to the privacy amplification step of QKD protocols.

Lemma 4.3. Let $\rho_{FB'}$ be the state given by the real execution of the protocol $\pi_{\mathbf{EXT}}$, where F is the system saving Alice's inputs (a, b), B' is Bob's (possibly quantum) system. Fix $\zeta \in]0, 1 - \frac{1}{d}]$ and let

$$\epsilon(\zeta, n) = \exp\left(-\frac{2\zeta^2 t^2 n^2}{(nt+1)(t+1)}\right). \tag{41}$$

Then, for any attack of a dishonest Bob, there exists a classical-quantum state $\sigma_{FB'}$, where $F = (F_a, F_b)$, such that

1. $\sigma_{FB'} \approx_\epsilon \rho_{FB'}$, and
2. $\delta(\tau_{\mathbb{Z}_d} \otimes \sigma_{B'}, \sigma_{F_a B'}) \leq K\, 2^{-n\, f_d(\zeta)}$, where $K = \frac{\sqrt{d}}{2}$, $f_d(\zeta) = \frac{\log d}{4}(1 - h_d(\zeta))$, n is the security parameter, and $h_d(\zeta)$ is given in Definition 2.2.

Proof. To prove property 1, we note that the extraction operation applied to the output of $\pi^n_{\mathbf{WOLE}}$ can be described by a projective operator on the space $F = (F_a, F_b)$. Therefore, as in the case of Lemma 4.2, property 1 follows from the fact that CPTP maps do not increase the trace distance (see Lemma 7 in [78]).

Regarding property 2, let us first consider Bob's subsystem E to integrate Bob's inputs $\boldsymbol{x}$, i.e. $E = \mathbf{X}E'$. Then, his full system B' is identified with $\mathbf{Y}E = \mathbf{YX}E'$. We have:

$$\begin{aligned} H_{\min}(\mathbf{F} \,|\, \mathbf{Y}E)_{\sigma_{\mathbf{FY}E}} &= H_{\min}(\mathbf{F}_{\boldsymbol{a}}, \mathbf{F}_{\boldsymbol{b}} \,|\, \mathbf{YX}E')_{\sigma_{\mathbf{FY}E}} && (42)\\ &= H_{\min}(\mathbf{F}_{\boldsymbol{a}}, \mathbf{Y} - \mathbf{F}_{\boldsymbol{a}}\mathbf{X} \,|\, \mathbf{YX}E')_{\sigma_{\mathbf{F}_a\mathbf{Y}E}} && (43)\\ &= H_{\min}(\mathbf{F}_{\boldsymbol{a}} \,|\, \mathbf{YX}E')_{\sigma_{\mathbf{F}_a\mathbf{Y}E}}. && (44) \end{aligned}$$

Therefore,

$$H_{\min}(\mathbf{F}_{\boldsymbol{a}} \,|\, \mathbf{Y}E)_{\sigma_{\mathbf{F}_a\mathbf{Y}E}} \geq \frac{n \log d}{2}(1 - h_d(\zeta)). \tag{45}$$

Now, since $\mathfrak{G}$ is a two-universal family of hash functions, we can directly apply Lemma 2.2 for $l = 1$. It follows that F_a is $\xi-$close to uniform conditioned on $\mathbf{Y}E$, i.e.

$$\delta(\tau_{\mathbb{Z}_d} \otimes \sigma_{\mathbf{Y}E}, \sigma_{F_a\mathbf{Y}E}) \leq \frac{1}{2}\sqrt{2^{\log d - \frac{n \log d}{2}(1 - h_d(\zeta))}} = K\, 2^{-n\, f_d(\zeta)} =: \xi, \tag{46}$$

where $K = \frac{\sqrt{d}}{2}$, $f_d(\zeta) = \frac{\log d}{4}(1 - h_d(\zeta))$ and n is the security parameter. □

Now, we are in position to combine the above $\pi^n_{\mathbf{RWOLE}}$, $\pi^n_{\mathbf{WOLE}}$ and $\pi_{\mathbf{EXT}}$, and present the full protocol $\pi_{\mathbf{QOLE}}$ in Figure 8.

Protocol $\pi_{\mathbf{QOLE}}$

Parameters: n, security parameter; tn, number of test qudits.
Alice's input: $(a, b) \in \mathbb{Z}_d^2$; **Bob's input:** $x \in \mathbb{Z}_d$

(Quantum phase:)

1. Bob randomly generates $m = (1+t)n$ different pairs $(x_i^0, r_i) \in \mathbb{Z}_d^2$ and commits to them by sending $(\texttt{commit}, (i, x_i^0, r_i)_{i\in[m]})$ to $\mathcal{F}_{\mathbf{COM}}$. He also prepares the quantum states $\left|e_{r_i}^{x_i^0}\right\rangle_{i\in[m]}$ and sends them to Alice.

2. Alice randomly chooses a subset of indices $T \subset [m]$ of size tn and sends it to Bob.

3. Bob sends $(\texttt{open}, i)_{i\in T}$ to $\mathcal{F}_{\mathbf{COM}}$ and $\mathcal{F}_{\mathbf{COM}}$ sends to Alice $(\texttt{open}, (i, x_i^0, r_i))_{i\in T}$.

4. Alice measures the received quantum states in the corresponding x_i^0 basis for $i \in T$, and checks whether the received commitments are compatible with her measurements. She proceeds in case there is no error, otherwise she aborts.

5. Alice randomly generates n pairs $(a_i^0, b_i^0) \in \mathbb{Z}_d^2$ and prepares $V_{a_i^0}^{b_i^0}$ for $i \in [m] \setminus T$. We relabel $\boldsymbol{a}_0 = (a_1^0, \ldots, a_n^0)$, $\boldsymbol{b}_0 = (b_1^0, \ldots, b_n^0)$ and $\boldsymbol{x}_0 = (x_1^0, \ldots, x_n^0)$, and from now on identify $[m] \setminus T \equiv [n]$.

6. Alice $\forall i \in [n]$ applies $V_{a_i^0}^{b_i^0}$ to the received state $\left|e_{r_i}^{x_i^0}\right\rangle$, i.e. $V_{a_i^0}^{b_i^0}\left|e_{r_i}^{x_i^0}\right\rangle = c_{x_i^0,a_i^0,b_i^0,r_i}\left|e_{a_i^0 x_i^0 - b_i^0 + r_i}^{x_i^0}\right\rangle$, and sends the resulting states to Bob.

7. Bob $\forall i \in [n]$ measures the received state in the corresponding basis x_i^0, and gets the state $\left|e_{a_i^0 x_i^0 - b_i^0 + r_i}^{x_i^0}\right\rangle$. Finally, $\forall i \in [n]$ he subtracts r_i from his result and gets $y_i^0 = a_i^0 x_i^0 - b_i^0$. We write $\boldsymbol{y}_0 = (y_1^0, \ldots, y_n^0)$.

(Post-processing phase:)

8. Bob defines $\boldsymbol{x} = (x, \ldots, x)$ as the constant vector according to his input x.

9. Alice chooses randomly some function $g_{\boldsymbol{\kappa}} \in \mathfrak{G}$, and she randomly generates $a_2, \ldots, a_n, b_2, \ldots, b_n \leftarrow_{\$} \mathbb{Z}_d$. She computes $a_1 = (a - \sum_{i=2}^n a_i\kappa_i)/\kappa_1$ and $b_1 = (b - \sum_{i=2}^n b_i\kappa_i)/\kappa_1$. We write $\boldsymbol{a} = (a_1, \ldots, a_n)$ and $\boldsymbol{b} = (b_1, \ldots, b_n)$.

10. Bob computes and sends to Alice $\boldsymbol{c} = \boldsymbol{x} - \boldsymbol{x}_0$.

11. Alice computes and sends to Bob $\boldsymbol{d} = \boldsymbol{a} - \boldsymbol{a}_0$ and $\boldsymbol{s} = \boldsymbol{b}_0 + \boldsymbol{a} * \boldsymbol{c} + \boldsymbol{b}$.

12. Bob computes $\boldsymbol{y} = \boldsymbol{y}_0 + \boldsymbol{x} * \boldsymbol{d} - \boldsymbol{d} * \boldsymbol{c} + \boldsymbol{s}$.

13. Finally, Alice sends $\boldsymbol{\kappa}$ to Bob and he computes $y = g_{\boldsymbol{\kappa}}(\boldsymbol{y})$.

Alice's output: $\perp$; **Bob's output:** y

Figure 8: QOLE protocol.

5 UC security

In this section, we will show that $\pi_{\mathbf{QOLE}}$ (Figure 8) is quantum-UC secure. More formally, we will show that $\pi_{\mathbf{QOLE}}$ statistically quantum-UC realizes (Definition 2.7) $\mathcal{F}_{\mathbf{OLE}}$ in the $\mathcal{F}_{\mathbf{COM}}-$hybrid model.

Theorem 5.1 (quantum-UC security of $\pi_{\mathbf{QOLE}}$)**.** The protocol $\pi_{\mathbf{QOLE}}$ from Figure 8 statistically quantum-UC realizes (see Definition 2.7) $\mathcal{F}_{\mathbf{OLE}}$ in the $\mathcal{F}_{\mathbf{COM}}-$hybrid model.

Theorem 5.1 is proved by combining Lemma 5.2 and Lemma 5.4 that we present below. In the former we prove the protocol's security for the case where Alice is dishonest and Bob is honest, while in the latter we prove security in the case where Alice is honest and Bob dishonest.

Lemma 5.2. The protocol $\pi_{\mathbf{QOLE}}$ (Figure 8) statistically quantum-UC realizes (see Definition 2.7) $\mathcal{F}_{\mathbf{OLE}}$ in the $\mathcal{F}_{\mathbf{COM}}-$hybrid model in the case of dishonest Alice and honest Bob.

Proof. We start by presenting the simulator $\mathcal{S}_A$ for the case where Alice is dishonest in Figure 9. Moreover, in Figure 10, we illustrate the corresponding real and ideal networks.

To prove statistical quantum-UC security according to Definition 2.7, we first consider a sequence of hybrid protocols from H_0 to H_4. The first hybrid protocol, H_0, is the real execution of the protocol $\pi_{\mathbf{QOLE}}$, and we gradually change it until obtaining the hybrid H_4 which corresponds to the description of the simulator $\mathcal{S}_A$. By proving indistinguishaility of the hybrids throughout the sequence, we show statistical quantum-UC security for $\pi_{\mathbf{QOLE}}$ in the case of dishonest Alice. Let us know present the hybrids in detail:

Hybrid H_0: This is the real execution of the protocol $\pi_{\mathbf{QOLE}}$.

Hybrid H_1: This hybrid is identical to H_0, except that we replace the $\mathcal{F}_{\mathbf{COM}}$ with a fake commitment functionality, $\mathcal{F}_{\mathbf{FakeCOM}}$, in which Bob, i.e. the honest party, can commit no value. This fake functionality works as follows:

- Commitment phase: expects a `commit` message from Bob instead of (`commit,` x).
- Open phase: expects a message (`open`, x) (instead of open) and sends (`open`, x) to Alice.

Hybrids H_0 and H_1 are perfectly indistinguishable, as the simulator still opens the commitments in the same way.

Hybrid H_2: This hybrid is identical to H_1, except that now $\mathcal{S}_A$ prepares entangled states $|B_{0,0}\rangle_{Q_AQ_S}$ instead of $\left|e_{r_i}^{x_i^0}\right\rangle_{i\in[m]}$, and sends the subsystem Q_A to Alice. Additionally, upon receiving the set of indices, T, from Alice, $\mathcal{S}_A$ measures the corresponding elements of subsystem Q_S using tn randomly chosen bases x_i^0 and provides $(\texttt{open}, (i, x_i^0, r_i))$ to $\mathcal{F}_{\mathbf{FakeCOM}}$, $\forall i \in T$.

Claim 5.3. The hybrids H_1 and H_2 are indistinguishable.

Simulator $\mathcal{S}_\mathbf{A}$

(Quantum phase:)

1. $\mathcal{S}_A$ sends `commit` to $\mathcal{F}_{\mathbf{FakeCOM}}$.
2. $\mathcal{S}_A$ generates $m = (1+t)n$ entangled states $|B_{0,0}\rangle_{Q_A Q_S}$ and sends subsystem Q_A to Alice.
3. Alice asks for a set of indices $T \subset [m]$ of size tn.
4. $\mathcal{S}_A$ measures the corresponding elements of subsystem Q_S using tn randomly chosen bases x_i^0 and provides $(\texttt{open}, (i, x_i^0, r_i))$ to $\mathcal{F}_{\mathbf{FakeCOM}}$, $\forall i \in T$.
5. Upon receiving the processed system $\hat{Q}_A$ from Alice, $\mathcal{S}_A$ measures the joint system $\hat{Q}_A Q_S$ and extracts the measurement outcomes $\mathbf{F} = (\boldsymbol{a}_0, \boldsymbol{b}_0) = ((a_1^0, \ldots, a_n^0), (b_1^0, \ldots, b_n^0))$.

(Post-processing phase:)

6. $\mathcal{S}_A$ randomly generates a vector $\boldsymbol{c}'$ and sends to Alice.
7. Upon receiving $\boldsymbol{d}$ and $\boldsymbol{s}$ from Alice, $\mathcal{S}_A$ extracts $\boldsymbol{a}$ and $\boldsymbol{b}$ based on its knowledge of $(\boldsymbol{a}_0, \boldsymbol{b}_0)$ as follows:

$$\begin{aligned} \boldsymbol{a} &= \boldsymbol{b} + \boldsymbol{a}_0 \\ \boldsymbol{b} &= \boldsymbol{s} - \boldsymbol{b}_0 - \boldsymbol{a} * \boldsymbol{c}'. \end{aligned} \tag{47}$$

8. Upon receiving $\boldsymbol{\kappa}$ from Alice, $\mathcal{S}_A$ extracts her inputs (a, b) as follows:

$$\begin{aligned} a &= \boldsymbol{a} \cdot \boldsymbol{\kappa} \\ b &= \boldsymbol{b} \cdot \boldsymbol{\kappa}. \end{aligned} \tag{48}$$

9. Finally, $\mathcal{S}_A$ sends (a, b) to the ideal functionality $\mathcal{F}_{\mathbf{OLE}}$.

Figure 9: Simulator $\mathcal{S}_A$ against dishonest Alice.

Proof. From Alice's point of view, the state received is exactly the same in both hybrids. In H_1, since the elements r are chosen randomly,

$$\frac{1}{d}\sum_{r=0}^{d-1} \left|e_r^{x^0}\right\rangle\!\left\langle e_r^{x^0}\right| = \frac{\mathbb{1}_A}{d}, \tag{49}$$

for each $x^0 = 0, \ldots, d-1$. In H_2

$$\mathrm{Tr}_{Q_S} |B_{0,0}\rangle\langle B_{0,0}| = \frac{\mathbb{1}_A}{d}. \tag{50}$$

Thus, the environment is not able to distinguish the two scenarios. Furthermore, upon Alice's request of the test set, T, the simulator measures in random bases, x_i^0 for $i \in T$,

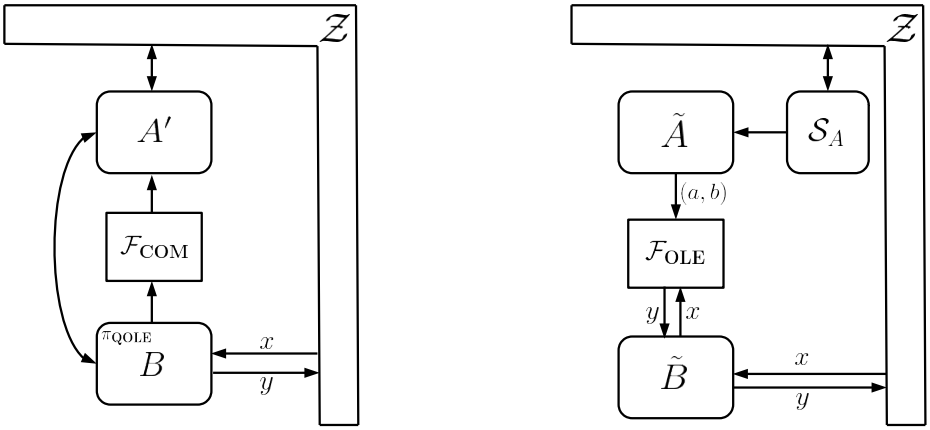

Figure 10: The real and ideal networks in the case of dishonest Alice. In the real network, Bob honestly follows the protocol $\pi_{\mathbf{QOLE}}$, while dishonest Alice A', can be thought as part of the environment $\mathcal{Z}$, since she is controlled by it. In the ideal network, $\tilde{A}$ and $\tilde{B}$ are the dummy parties for Alice and Bob, respectively.

the corresponding qudits of subsystem Q_S. Since both entangled qudits in $Q_A Q_S$ get projected to the some random state, r_i for $i \in T$, $\mathcal{F}_{\mathbf{FakeCOM}}$ provides the correct pair $(x_i^0, r_i)_{i \in T}$ to Alice. Hence, the hybrids H_1 and H_2 are indistinguishable. □

Hybrid H_3: This hybrid is identical to H_2, except that now $\mathcal{S}_A$ extracts Alice's elements $\mathbf{F}_0 = (\boldsymbol{a}_0, \boldsymbol{b}_0)$ by applying a joint measurement on the systems $\hat{Q}_A Q_S$ in the generalized Bell basis.

Hybrids H_2 and H_3 are perfectly indistinguishable, as the simulator only changes the measurement basis for the received state and does not communicate with Alice.

Hybrid H_4: This hybrid is identical to H_3, except that now $\mathcal{S}_A$ generates $\boldsymbol{c}'$ uniformly at random. Additionally, upon receiving $\boldsymbol{d}$, $\boldsymbol{s}$ and $\boldsymbol{\kappa}$, the simulator extracts Alice's vectors $(\boldsymbol{a}, \boldsymbol{b})$ and inputs (a, b) by computing expressions (47) and (48). Finally, $\mathcal{S}_A$ sends (a, b) to the ideal functionality $\mathcal{F}_{\mathbf{OLE}}$. Hybrid H_4 corresponds to the description of the simulator $\mathcal{S}_A$.

Hybrids H_3 and H_4 are perfectly indistinguishable for the following reasons: first, from the proof of Claim 5.3, we have that the vector $\boldsymbol{x}_0$ looks uniformly random to Alice, and consequently, so does $\boldsymbol{c}$. Second, the extraction operations do not require any interaction with Alice. □

We now proceed to the case where Alice is honest and Bob is dishonest.

Lemma 5.4. The protocol $\pi_{\mathbf{QOLE}}$ (Figure 8) statistically quantum-UC realizes (Definition 2.7) $\mathcal{F}_{\mathbf{OLE}}$ in the $\mathcal{F}_{\mathbf{COM}}$−hybrid model in the case of honest Alice and dishonest Bob.

Proof. We start by presenting the simulator $\mathcal{S}_B$ for the case where Bob is dishonest in Figure 11. Moreover, in Figure 12, we illustrate the corresponding real and ideal networks.

Then, just like in the previous case for dishonest Alice, we consider the respective sequence of hybrid protocols, from H_0 (execution of the protocol) to H_2 (simulator $\mathcal{S}_B$), and prove that they are indistinguishable in the case of dishonest Bob.

Simulator $\mathcal{S}_\mathbf{B}$

(Quantum phase:)

1. $\mathcal{S}_B$ receives the qudits from Bob and tests them as in the protocol $\pi_{\mathbf{QOLE}}$.
2. $\mathcal{S}_B$ randomly chooses vectors $\boldsymbol{a}_0$ and $\boldsymbol{b}_0$ and applies $V_{a_i^0}^{b_i^0}$, $i \in [n]$ to the received qudits.
3. $\mathcal{S}_B$ extracts the input element $\boldsymbol{x}_0$ from $\mathcal{F}_{\mathbf{COM}}$.

(Post-processing phase:)

4. Upon receiving $\boldsymbol{c}$ from Bob, $\mathcal{S}_B$ extracts his input x as $\boldsymbol{x} = \boldsymbol{c} + \boldsymbol{x}_0$.
5. $\mathcal{S}_B$ sends x to $\mathcal{F}_{\mathbf{OLE}}$ and receives y.
6. $\mathcal{S}_B$ randomly generates the elements $a' \leftarrow_\$ \mathbb{Z}_d$, $\boldsymbol{\kappa} \leftarrow_\$ \mathbb{Z}_d^n$ and $a_2, \ldots, a_n, b_2, \ldots, b_n \leftarrow_\$ \mathbb{Z}_d$.
7. $\mathcal{S}_B$ computes $b' = a'x - y$, $a_1 = (a' - \sum_{i=2}^n a_i\kappa_i)/\kappa_1$ and $b_1 = (b' - \sum_{i=2}^n b_i\kappa_i)/\kappa_1$.
8. $\mathcal{S}_B$ sends $\boldsymbol{d} = \boldsymbol{a} - \boldsymbol{a}_0$, $\boldsymbol{s} = \boldsymbol{b}_0 + \boldsymbol{a} * \boldsymbol{c} + \boldsymbol{b}$ and $\boldsymbol{\kappa}$ to Bob.

Figure 11: Simulator $\mathcal{S}_B$ against dishonest Bob.

Hybrid H_0: This is the execution of the real protocol $\pi_{\mathbf{QOLE}}$. In this hybrid, $\mathcal{S}_B$ behaves just like honest Alice up to step 6 of $\pi_{\mathbf{QOLE}}$: tests the received qudits (steps 1-4), randomly generates n pairs $(a_i^0, b_i^0)_{i\in[n]}$ (step 5), and applies the respective operators $V_{a_i^0}^{b_i^0}$ for $i \in [n]$ to the received states (step 6).

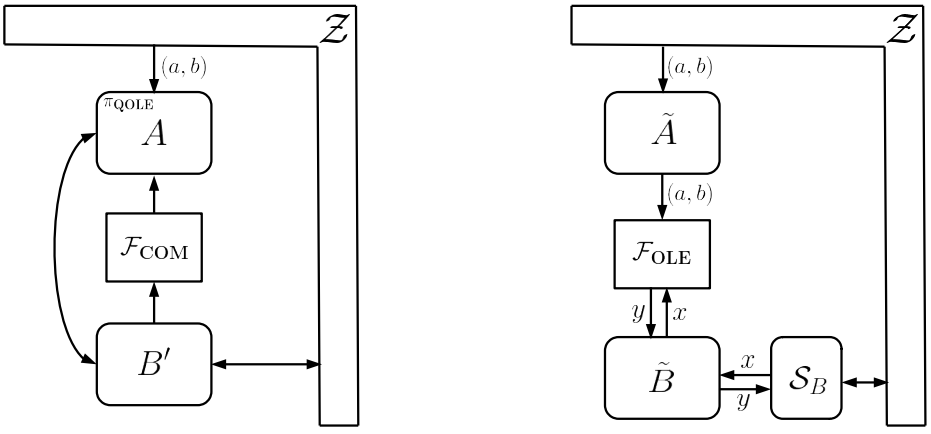

Figure 12: The real and ideal networks in the case of dishonest Bob. In the real network, Alice honestly follows the protocol $\pi_{\mathbf{QOLE}}$, while dishonest Bob B', can be thought as part the environment $\mathcal{Z}$, since he is controlled by it. In the ideal network, $\tilde{A}$ and $\tilde{B}$ are the dummy parties for Alice and Bob, respectively.

Hybrid H_1: This hybrid is identical to H_0, except that now $\mathcal{S}_B$ extracts Bob's random vector $\boldsymbol{x}_0$ from the commitment functionality $\mathcal{F}_{\mathbf{COM}}$. Additionally, upon receiving $\boldsymbol{c}$ from Bob, $\mathcal{S}_B$ extracts Bob's input x by computing $\boldsymbol{c} + \boldsymbol{x}_0$. Then, $\mathcal{S}_B$ sends the extracted

element x to $\mathcal{F}_{\mathbf{OLE}}$ and receives y.

Hybrids H_0 and H_1 are perfectly indistinguishable, because $\mathcal{S}_B$ only interacts with Bob when receiving the element $\boldsymbol{c}$, and this does not change anything from Bob's point of view. The corresponding operations are either carried out locally by $\mathcal{S}_B$ or along with $\mathcal{F}_{\mathbf{COM}}$ which, by definition, is fully controlled by $\mathcal{S}_B$.

Hybrid H_2: This hybrid is identical to H_1, except that now $\mathcal{S}_{\mathcal{B}}$ generates (a, b), $\boldsymbol{d}$ and $\boldsymbol{s}$: it starts by randomly generating $a' \leftarrow_{\$} \mathbb{Z}_d$, $\boldsymbol{\kappa} \leftarrow_{\$} \mathbb{Z}_d^n$ and $a_2, \ldots, a_n, b_2, \ldots, b_n \leftarrow_{\$} \mathbb{Z}_d$. Then, it computes b' according to the generated a', the extracted element x and the output y of $\mathcal{F}_{\mathbf{OLE}}$, as $b' = a'x - y$. It masks a' and b' as

$$a' = \boldsymbol{a} \cdot \boldsymbol{\kappa} \qquad \text{and} \qquad b' = \boldsymbol{b} \cdot \boldsymbol{\kappa}, \tag{51}$$

by setting a_1 and b_1 accordingly, i.e. $a_1 = (a' - \sum_{i=2}^{n} a_i \kappa_i)/\kappa_1$ and $b_1 = (b' - \sum_{i=2}^{n} b_i \kappa_i)/\kappa_1$. Finally, $\mathcal{S}_B$ sends $\boldsymbol{d} = \boldsymbol{a} - \boldsymbol{a}_0$, $\boldsymbol{s} = \boldsymbol{b}_0 + \boldsymbol{a} * \boldsymbol{c} + \boldsymbol{b}$ and $\boldsymbol{\kappa}$ to Bob. This is the last hybrid of the sequence and corresponds to the description of $\mathcal{S}_B$.

Claim 5.5. The hybrids H_1 and H_2 are indistinguishable.

Proof. Since, in its first two steps, $\mathcal{S}_B$ executes a RWOLE scheme, according to Lemma 4.1 we have that $\mathcal{S}_B$ is $\epsilon-$close to a situation where Bob's knowledge on the vectors $(\boldsymbol{a}_0, \boldsymbol{b}_0)$ is lower-bounded by the value

$$\frac{1}{n}\lambda(\zeta) = \frac{\log d}{2}(1 - h_d(\zeta)), \tag{52}$$

for $\zeta \in]0, 1 - \frac{1}{d}]$, n the security parameter and $\epsilon(\zeta, n) = \exp\left(-\frac{2\zeta^2 t^2 n^2}{(nt+1)(t+1)}\right)$. Also, as Bob receives $\boldsymbol{d}$ and $\boldsymbol{s}$, according to Lemma 4.2, his knowledge on $(\boldsymbol{a}, \boldsymbol{b})$ is also lower-bounded by the same $\lambda(\zeta)/n$. Furthermore, since $\mathcal{S}_B$ defines $\boldsymbol{a}$ such that $a' = \boldsymbol{a} \cdot \boldsymbol{\kappa}$, from Lemma 4.3 we can conclude that $\mathcal{S}_B$ is $(\xi + \epsilon)-$close to a scenario where a' is uniformly distributed. This comes from the properties in Lemma 4.3 and the triangle inequality:

$$\begin{aligned} \delta(\tau_{\mathbb{Z}_d} \otimes \sigma_{B'},\, \rho_{F_a B'}) &\leq \delta(\tau_{\mathbb{Z}_d} \otimes \sigma_{B'},\, \sigma_{F_a B'}) + \delta(\sigma_{F_a B'},\, \rho_{F_a B'}) \\ &\leq K\, 2^{-n\, f_d(\zeta)} + e^{-\frac{2\zeta^2 t^2 n^2}{(nt+1)(t+1)}} = \xi + \epsilon, \end{aligned} \tag{53}$$

where $K = \frac{\sqrt{d}}{2}$, $f_d(\zeta) = \frac{\log d}{4}(1 - h_d(\zeta))$. This means that the triple $(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\kappa})$ only gives to the environment a negligible advantage in distinguishing between the real and ideal world executions. □

□

6 Protocol Generalizations

6.1 QOLE in Galois fields of prime-power dimensions

So far, we have been working in Hilbert spaces of prime dimensions; this reflects the fact that, for prime d, $\mathbb{Z}_d$ is a field and, under a well-defined set of MUBs $\{|e_r^x\rangle\}_{r \in \mathbb{Z}_d}$, $\forall x \in \mathbb{Z}_d$, we have the affine relation (24):

$$V_a^b \left|e_r^x\right\rangle = c_{a,b,x,r} \left|e_{ax-b+r}^x\right\rangle. \tag{54}$$

In this section, we generalize our protocol, $\pi_{\mathbf{QOLE}}$, to Hilbert spaces of prime-power dimensions, $N = d^M$ (d prime and $M > 1$), taking advantage of the fact that in a Galois field of dimension d^M, $GF(d^M)$, we can build a complete set of $N + 1$ MUBs [34].

Succinctly, in $GF(d^M)$, we identify the integers $i \in \mathbb{Z}_N$ with their $d-$ary representation, i.e.

$$\mathbb{Z}_N \ni i = \sum_{n=0}^{M-1} i_n d^n \longleftrightarrow (i_0, \ldots, i_{M-1}) \in GF(d^M). \tag{55}$$

In these fields there are two operations, addition and multiplication, which we denote by $\oplus$ and $\odot$, respectively. Addition is straightforward, as it is given by the component-wise addition modulo d of elements, i.e. $i \oplus j = (i_0 + j_0 \mod d, \ldots, i_{M-1} + j_{M-1} \mod d)$. Considering $i = \sum_{n=0}^{M-1} i_n d^n$ as a polynomial of degree $M-1$ given by $i(p) = \sum_{n=0}^{M-1} i_n p^n$, multiplication between two elements i, j, is given by the multiplication between the corresponding polynomials $i(p)$ and $j(p)$ modulo some irreducible polynomial $m(p)$, i.e. $i \odot j = (i(p) \times j(p)) \mod m(p)$.

Analogously to prime-dimension fields, we can write the operators V_a^b in the computational basis, as

$$V_a^b = \sum_{k=0}^{N-1} |k \oplus a\rangle \, \omega^{(k\oplus a)\odot b} \langle k| \,, \tag{56}$$

and the eigenstates for the corresponding $N + 1$ pairwise MUBs, as

$$|e_r^x\rangle = \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} |l\rangle \, \omega^{\ominus(r\odot l)} \alpha_{\ominus l}^{x*}, \tag{57}$$

where $\alpha_{\ominus l}^x$ is a phase factor whose form depends on whether d is even or odd. For details, see Section 2.4.2 in [34].

Given the above, we can derive (Appendix C) the following affine relation similar to (24):

$$V_a^b \left| e_r^i \right\rangle = \omega^{r\odot a} \alpha_a^{i*} \left| e_{i\odot a\ominus b\oplus r}^i \right\rangle. \tag{58}$$

Notice that all the steps in the $\pi_{\mathbf{QOLE}}$ depend on the properties of the field operations and on the fact that (24) holds. Hence, we can use $\pi_{\mathbf{QOLE}}$ adapted for the operations $\oplus$ and $\odot$, in order to quantum-UC-realize $\mathcal{F}_{\mathbf{OLE}}$ in fields of prime-power dimension d^M.

6.2 Quantum Vector OLE

In the proposed protocol $\pi_{\mathbf{QOLE}}$, we extract one instance of OLE out of n instances of WOLE. As far as efficiency is concerned, it would be desirable to generate more instances of OLE out of those n instances of WOLE. Here, we show how to use WOLE as a resource to realize the VOLE functionality, $\mathcal{F}_{\mathbf{VOLE}}^k$, presented in Figure 2. In this case, Alice fixes a k (which is specified later), defines a set of k linear functions $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{F}_q^k \times \mathbb{F}_q^k$ and Bob outputs the evaluation of all these functions on a specified element $x \in \mathbb{F}_q$ that he chooses, i.e. $\boldsymbol{f} := \boldsymbol{a}x + \boldsymbol{b}$. Since $\pi_{\mathbf{QOLE}}$ can be extended to finite fields $\mathbb{F}_q$, where q is a prime or prime-power number (as shown above), the $\mathcal{F}_{\mathbf{VOLE}}^k$ functionality can also be defined in $\mathbb{F}_q$.

In the extraction phase of $\pi_{\mathbf{QOLE}}$, Alice randomly chooses a function $g_{\boldsymbol{\kappa}}$ and applies it to the pair $(\boldsymbol{a}, \boldsymbol{b})$. This procedure suggests that, in order to generate different input

elements (a', b'), Alice can randomly choose another function $g_{\boldsymbol{\kappa'}}$ and set $a' = g_{\boldsymbol{\kappa'}}(\boldsymbol{a})$ and $b' = g_{\boldsymbol{\kappa'}}(\boldsymbol{b})$. This is equivalent to generating a random $2 \times n$ matrix in $\mathbb{F}_q$, i.e.

$$\begin{bmatrix} — & \boldsymbol{\kappa} & — \\ — & \boldsymbol{\kappa'} & — \end{bmatrix} \begin{bmatrix} | & | \\ \boldsymbol{a} & \boldsymbol{b} \\ | & | \end{bmatrix} = \begin{bmatrix} a & b \\ a' & b' \end{bmatrix}. \tag{59}$$

However, in case $\boldsymbol{\kappa}$ and $\boldsymbol{\kappa'}$ are linearly dependent (i.e. $\boldsymbol{\kappa} = c\boldsymbol{\kappa'}$ for some $c \in \mathbb{Z}_d$), Bob would have some extra information about Alice's elements (a, b) and (a', b'), as $(a, b) = c(a', b')$. This leads to a situation beyond the $\mathcal{F}^k_{\mathbf{VOLE}}$ definition. To avoid this issue, let us consider the set of $k \times n$ matrices with rank k over $\mathbb{F}_q$ for $1 \leq k \leq n$, and denote it by $\mathcal{R}_{k\times n}(\mathbb{F}_q)$. For a binary finite field, $\mathcal{R}_{k\times n}(\mathbb{F}_2)$ is a two-universal hash family [21, 64]. Similarly, one can prove that the more general set $\mathcal{R}_{k\times n}(\mathbb{F}_q)$ is also a two-universal hash family from $\mathbb{F}_q^n$ to $\mathbb{F}_q^k$. During the extraction phase of the original $\pi_{\mathbf{QOLE}}$, Alice chooses vectors $(\boldsymbol{a}, \boldsymbol{b})$ according to the random vector $\boldsymbol{\kappa}$ and the desired final elements (a, b) (see step 9 in Figure 8). In that case, since there is only one random vector $\boldsymbol{\kappa}$, there are $n-1$ undefined variables for each vector $\boldsymbol{a}$ and $\boldsymbol{b}$, i.e. $a_2, \ldots, a_n$ and $b_2, \ldots, b_n$ that can be chosen freely. For the VOLE protocol, instead of choosing just one vector $\boldsymbol{\kappa}$, Alice randomly chooses a matrix $\mathcal{K} \in \mathcal{R}_{k\times n}(\mathbb{F}_q)$ of rank k. She then defines vectors $(\boldsymbol{a'}, \boldsymbol{b'}) \in \mathbb{F}_q^n \times \mathbb{F}_q^n$ consistent with the final elements $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{F}_q^k \times \mathbb{F}_q^k$. That is, Alice has the following system:

$$\begin{bmatrix} —— & \boldsymbol{\kappa}_1 & —— \\ & \cdots & \\ —— & \boldsymbol{\kappa}_k & —— \end{bmatrix} \begin{bmatrix} | & | \\ \boldsymbol{a'} & \boldsymbol{b'} \\ | & | \end{bmatrix} = \begin{bmatrix} | & | \\ \boldsymbol{a} & \boldsymbol{b} \\ | & | \end{bmatrix}, \tag{60}$$

that can be solved by means of the Gaussian elimination method. Since $\mathcal{K} \in \mathcal{R}_{k\times n}(\mathbb{F}_q)$, there will be $n - k$ undefined variables in both vectors $\boldsymbol{a'}$ and $\boldsymbol{b'}$. Let U denote the set of undefined indexes in $\boldsymbol{a'}$ and $\boldsymbol{b'}$. Alice randomly chooses a'_i and b'_i for $i \in U$ and solves the above equation system. Then, they proceed similarly to the original $\pi_{\mathbf{QOLE}}$ and execute the derandomization protocol $\pi^n_{\mathbf{WOLE}}((\boldsymbol{a'}, \boldsymbol{b'}), \boldsymbol{x})$. Finally, Bob applies Alice's chosen matrix $\mathcal{K}$ to his output vector $\boldsymbol{y'}$ to get the final element $\boldsymbol{y}$. This vectorized extraction protocol $\pi_{\mathbf{VEXT}}$ is presented in Figure 13.

The correctness of the protocol is drawn immediately from linearity:

$$\boldsymbol{y} = \mathcal{K}\boldsymbol{y'} = \mathcal{K}(\boldsymbol{a'}x + \boldsymbol{b'}) = \boldsymbol{a}x + \boldsymbol{b}. \tag{62}$$

The security of the protocol is constrained by $\xi = \frac{1}{2}\sqrt{2^{k \log q - H_{\min}(X|E)}}$ given by Lemma 2.2, where we consider l to be k and d to be q. As before, $\mathbf{F}_{\boldsymbol{a'}}$ denotes the distribution of the $\pi^n_{\mathbf{WOLE}}$ protocol's input $\boldsymbol{a'}$ from Bob's perspective. From Lemma 2.2, since $\mathcal{R}_{k\times n}(\mathbb{F}_q)$ is a two-universal family of hash functions, we know that $\mathcal{K} \in \mathcal{R}_{k\times n}(\mathbb{F}_q)$ approximates $\mathcal{K}\mathbf{F}_{\boldsymbol{a'}} = \mathbf{F}_{\boldsymbol{a}}$ to uniform conditioned on Bob's side information. However, the closeness parameter ξ, has to be negligible in the security parameter n, thus setting a bound on k (the size of VOLE), i.e. for $\eta > 0$,

$$k \log q - \frac{n \log q}{2}(1 - h_q(\zeta)) < -n\eta \log q \tag{63}$$

$$k < n\Big(\frac{1}{2}(1 - h_q(\zeta)) - \eta\Big). \tag{64}$$

Since $k > 0$, we have that $0 < \eta < \frac{1}{2}(1 - h_q(\zeta))$. This gives a bound on the elements that we can extract from n WOLEs and shows how Alice can fix k in the beginning. Note that

Protocol $\pi_{\mathbf{VEXT}}$

Alice's input: $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{F}_q^k \times \mathbb{F}_q^k$; **Bob's input:** $x \in \mathbb{F}_q$

1. Alice chooses randomly a matrix $\mathcal{K} \in \mathcal{R}_{k\times n}(\mathbb{F}_q)$ and sends it to Bob.
2. Using the Gaussian elimination method, Alice finds one solution of the system:

$$\mathcal{K}\begin{bmatrix} | & | \\ \boldsymbol{a'} & \boldsymbol{b'} \\ | & | \end{bmatrix} = \begin{bmatrix} | & | \\ \boldsymbol{a} & \boldsymbol{b} \\ | & | \end{bmatrix} \tag{61}$$

   (a) Alice finds the set U of undefined indexes in $\boldsymbol{a'}$ and $\boldsymbol{b'}$.
   (b) Alice randomly generates $a'_i, b'_i \leftarrow_{\$} \mathbb{F}_q$ for $i \in U$.
   (c) Alice solves the system for indexes $i \notin U$.

3. Alice and Bob run $\pi^n_{\mathbf{WOLE}}((\boldsymbol{a'}, \boldsymbol{b'}), \boldsymbol{x})$, where $\boldsymbol{x} = (x, \ldots, x)$. Bob outputs $\boldsymbol{y'} \in \mathbb{F}_q^n$.
4. Bob computes $\boldsymbol{y} = \mathcal{K}\boldsymbol{y'}$.

Alice's output: $\perp$; **Bob's output:** $\boldsymbol{y} \in \mathbb{F}_q^k$

Figure 13: Extraction protocol for VOLE.

this bound is not necessarily optimal, and one could try to improve it. We leave this as future work, as it goes beyond the scope of this paper.

Let us denote by $\pi_{\mathbf{QVOLE}}$ the protocol $\pi_{\mathbf{QOLE}}$ with the subprotocol $\pi_{\mathbf{VEXT}}$ instead of $\pi_{\mathbf{EXT}}$. For the security of $\pi_{\mathbf{QVOLE}}$, we have:

Theorem 6.1 (quantum-UC security of $\pi_{\mathbf{QVOLE}}$)**.** The protocol $\pi_{\mathbf{QOVLE}}$ statistically quantum-UC realizes (see Definition 2.7) $\mathcal{F}^k_{\mathbf{VOLE}}$ in the $\mathcal{F}_{\mathbf{COM}}-$hybrid model.

The proof is much the same as the proof of Theorem 5.1, therefore we omit it.

7 Outlook and further work

OLE is an important primitive for secure two-party computation, and while for other primitives such as bit commitment, OT and coin flipping there is a plethora of both theoretical as well as concrete protocol proposals [6, 7, 10, 11, 15, 47, 49, 55, 56, 58, 59, 63, 65, 68], up until now, there was no OLE protocol based on quantum communication. In this work, we present two protocols for QOLE. The first protocol is secure against semi-honest adversaries in the static corruption setting. The second proposed protocol, $\pi_{\mathbf{QOLE}}$, builds upon the semi-honest version and extends it to the dishonest case, following a commit-and-open approach. We prove this second protocol to be secure in the quantum-UC framework when assuming ideal commitments, making it possible to be composed in any arbitrary way. We also constructed two generalizations of our protocol: the first achieves QOLE in Galois fields of prime-power dimensions and the second is a protocol for quantum vector

OLE. Note that our protocol achieves everlasting security, i.e. it remains information-theoretically secure after its execution, even if the dishonest party becomes more powerful in the future.

While the security of $\pi_{\mathbf{QOLE}}$ is thoroughly analyzed, this is done in the noiseless case. Proving security assuming the existence of noise should follow a similar reasoning (see also [29]). Indeed, in the presence of noise, in Step 4 in the quantum phase of $\pi_{\mathbf{QOLE}}$ (Figure 8), Alice should abort the protocol if the errors measured (err) exceed some predetermined value ν, that is assumed to be due to noise. This way, the error parameter is $err = \nu + \zeta'$, where ζ' accounts for the activity of a dishonest Bob. Naturally, this will decrease the bound on the min-entropy of Alice's functions $\mathbf{F}$ given Bob's side information, i.e.

$$H_{\min}(\mathbf{F}|\mathbf{Y}E)_{\sigma_{\mathbf{FY}E}} \geq \frac{n \log d}{2} \left(1 - h_d(\nu + \zeta' + \zeta)\right). \tag{65}$$

Although it is possible to generalize the security results to the setting of noisy quantum communication, it is not guaranteed that the proposed protocol retains correctness. Therefore, as future work, it would be useful to propose specific implementations where noise is taken into account and its effect on the correctness and security is studied. Furthermore, the protocol's resilience to practical attacks [16] can be investigated.

Following a different assumption model, the $\pi_{\mathbf{QOLE}}$ protocol can be easily adapted to remain secure in the bounded-quantum-storage model. In this adapted version, the test phase of $\pi^n_{\mathbf{RWOLE}}$ is simply substituted by a waiting time Δt. This ensures that Bob is only able to store a noisy or limited amount of qudits. It would be interesting to explore how different noisy channels affect the security properties. Also, to guarantee the composability of the protocol, an analysis in the bounded-quantum-storage-UC model as put forth by Unruh [77] can be performed.

Our protocol is a two-way protocol, i.e. Bob prepares and sends a quantum state, Alice applies some operation to it, and sends it back to Bob who measures the final state. For QOT there exist several proposals for two-way protocols [3, 23, 24, 50], and, in particular, the one presented by Amiri et al. [3] also demonstrates their experimental feasibility. This is further motivation to work on developing realistic practical implementations of our protocol. Furthermore, one could increase the security standards by making our protocol device-independent. The proposal of Kundu et al. [50], who extended the work of Chailloux et al. [23], could serve as an inspiration. And while the aforementioned works focus on two-way QOT protocols, recently, non-interactive or one-way protocols have also been proposed for device-independent [17] and XOR QOT [72].

Finally, based on our results, one could construct quantum protocols for oblivious polynomial evaluation, which – as mentioned in Section 1 – is another important primitive facilitating various applications.

Acknowlegments

This work was funded by Fundação para a Ciência e a Tecnologia (FCT) through National Funds under Award SFRH/BD/144806/2019, Award UIDB/50008/2020, and Award UIDP/50008/2020; in part by the Regional Operational Program of Lisbon; by FCT, I.P. and, when eligible, by COMPETE 2020 FEDER funds, through the Competitiveness and Internationalization Operational Programme (COMPETE 2020), under the project QuantumPrime PTDC/EEI-TEL/8017/2020, and under the Scientific Employment Stimulus - Individual Call (CEEC Individual) - 2020.03274.CEECIND/CP1621/CT0003.

References

[1] Antonio Acin, Nicolas Gisin, and Valerio Scarani. Security bounds in quantum cryptography using d-level systems. *Quantum Information and Computation*, 3(6), 2003. DOI: 10.26421/QIC3.6-1.

[2] Irfan Ali-Khan, Curtis J. Broadbent, and John C. Howell. Large-alphabet quantum key distribution using energy-time entangled bipartite states. *Phys. Rev. Lett.*, 98: 060503, 2007. DOI: 10.1103/PhysRevLett.98.060503.

[3] Ryan Amiri, Robert Stárek, David Reichmuth, Ittoop V. Puthoor, Michal Mičuda, Jr. Ladislav Mišta, Miloslav Dušek, Petros Wallden, and Erika Andersson. Imperfect 1-out-of-2 quantum oblivious transfer: Bounds, a protocol, and its experimental implementation. *PRX Quantum*, 2(1), 2021. DOI: 10.1103/prxquantum.2.010335.

[4] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. How to garble arithmetic circuits. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 120–129, 2011. DOI: 10.1109/FOCS.2011.40.

[5] Benny Applebaum, Ivan Damgård, Yuval Ishai, Michael Nielsen, and Lior Zichron. Secure arithmetic computation with constant computational overhead. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017*, pages 223–254, Cham, 2017. Springer International Publishing. DOI: 10.1007/978-3-319-63688-7_8.

[6] Atul Singh Arora, Jérémie Roland, and Stephan Weis. Quantum weak coin flipping. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 205–216, New York, NY, USA, 2019. Association for Computing Machinery. DOI: 10.1145/3313276.3316306.

[7] Atul Singh Arora, Jérémie Roland, and Chrysoula Vlachou. Analytic quantum weak coin flipping protocols with arbitrarily small bias. In *Proceedings of the Thirty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '21, page 919–938. Society for Industrial and Applied Mathematics, 2021. DOI: 10.1137/1.9781611976465.58.

[8] Michael Ben-Or and Dominic Mayers. General security definition and composability for quantum and classical protocols. *arXiv*, 0409062, 2004. DOI: 10.48550/arXiv.quant-ph/0409062.

[9] Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélène Skubiszewska. Practical quantum oblivious transfer. In Joan Feigenbaum, editor, *Advances in Cryptology — CRYPTO '91*, pages 351–366, Berlin, Heidelberg, 1992. Springer Berlin Heidelberg. DOI: 10.1007/3-540-46766-1_29.

[10] Guido Berlín, Gilles Brassard, Félix Bussières, and Nicolas Godbout. Fair loss-tolerant quantum coin flipping. *Phys. Rev. A*, 80:062321, 2009. DOI: 10.1103/PhysRevA.80.062321.

[11] Guido Berlín, Gilles Brassard, Félix Bussières, Nicolas Godbout, Joshua A. Slater, and Wolfgang Tittel. Experimental loss-tolerant quantum coin flipping. *Nature Communications*, 2(1):561, 2011. DOI: 10.1038/ncomms1572.

[12] Frédéric Bouchard, Khabat Heshami, Duncan England, Robert Fickler, Robert W. Boyd, Berthold-Georg Englert, Luis L. Sánchez-Soto, and Ebrahim Karimi. Experimental investigation of high-dimensional quantum key distribution protocols with twisted photons. *Quantum*, 2:111, 2018. DOI: 10.22331/q-2018-12-04-111.

[13] Frédéric Bouchard, Natalia Herrera Valencia, Florian Brandt, Robert Fickler, Marcus Huber, and Mehul Malik. Measuring azimuthal and radial modes of photons. *Opt. Express*, 26(24):31925–31941, 2018. DOI: 10.1364/OE.26.031925.

[14] Elette Boyle, Geoffroy Couteau, Niv Gilboa, and Yuval Ishai. Compressing vector ole. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2018. DOI: 10.1145/3243734.3243868.

[15] Mathieu Bozzio, Ulysse Chabaud, Iordanis Kerenidis, and Eleni Diamanti. Quantum weak coin flipping with a single photon. *Phys. Rev. A*, 102:022414, 2020. DOI: 10.1103/PhysRevA.102.022414.

[16] Mathieu Bozzio, Adrien Cavaillès, Eleni Diamanti, Adrian Kent, and Damián Pitalúa-García. Multiphoton and side-channel attacks in mistrustful quantum cryptography. *PRX Quantum*, 2:030338, 2021. DOI: 10.1103/PRXQuantum.2.030338.

[17] Anne Broadbent and Peter Yuen. Device-independent oblivious transfer from the bounded-quantum-storage-model and computational assumptions. *New Journal of Physics*, 25(5):053019, 2023. DOI: 10.1088/1367-2630/accf32.

[18] Ran Canetti. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology*, 13(1):143–202, 2000. DOI: 10.1007/s001459910006.

[19] Ran Canetti. Universally composable security. *Journal of the ACM*, 67(5):1–94, 2020. DOI: 10.1145/3402457.

[20] Ran Canetti and Marc Fischlin. Universally composable commitments. In *Advances in Cryptology CRYPTO 2001*, pages 19–40. Springer Berlin Heidelberg, 2001. DOI: 10.1007/3-540-44647-8_2.

[21] J. Lawrence Carter and Mark N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18(2):143–154, 1979. DOI: 10.1016/0022-0000(79)90044-8.

[22] Nicolas J. Cerf, Mohamed Bourennane, Anders Karlsson, and Nicolas Gisin. Security of quantum key distribution using d-level systems. *Phys. Rev. Lett.*, 88:127902, 2002. DOI: 10.1103/PhysRevLett.88.127902.

[23] André Chailloux, Iordanis Kerenidis, and Jamie Sikora. Lower bounds for Quantum Oblivious Transfer. In Kamal Lodaya and Meena Mahajan, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2010)*, volume 8 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 157–168, Dagstuhl, Germany, 2010. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. DOI: 10.4230/LIPIcs.FSTTCS.2010.157.

[24] André Chailloux, Gus Gutoski, and Jamie Sikora. Optimal bounds for semi-honest quantum oblivious transfer. *Chicago Journal of Theoretical Computer Science*, (13): 1–17, 2016. DOI: 10.4086/cjtcs.2016.013.

[25] Melissa Chase, Yevgeniy Dodis, Yuval Ishai, Daniel Kraschewski, Tianren Liu, Rafail Ostrovsky, and Vinod Vaikuntanathan. Reusable non-interactive secure computation. *Advances in Cryptology – CRYPTO 2019*, 11694:462–488, 2019. DOI: 10.1007/978-3-030-26954-8_15.

[26] David Chaum, Claude Crépeau, and Ivan Damgard. Multiparty unconditionally secure protocols. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 11–19, New York, NY, USA, 1988. Association for Computing Machinery. DOI: 10.1145/62212.62214.

[27] Bruno Costa, Pedro Branco, Manuel Goulão, Mariano Lemus, and Paulo Mateus. Randomized oblivious transfer for secure multiparty computation in the quantum setting. *Entropy*, 23(8):1001, 2021. DOI: 10.3390/e23081001.

[28] Claude Crépeau. Quantum oblivious transfer. *Journal of Modern Optics*, 41(12): 2445–2454, 1994. DOI: 10.1080/09500349414552291.

[29] Ivan Damgård, Serge Fehr, Carolin Lunemann, Louis Salvail, and Christian Schaffner. Improving the security of quantum protocols via commit-and-open. In *Advances in*

Cryptology - CRYPTO 2009, pages 408–427, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. DOI: 10.1007/978-3-642-03356-8_24.

[30] Ivan Damgård, Helene Haagh, Michael Nielsen, and Claudio Orlandi. Commodity-based 2pc for arithmetic circuits. In Martin Albrecht, editor, *Cryptography and Coding*, pages 154–177, Cham, 2019. Springer International Publishing. DOI: 10.1007/978-3-030-35199-1_8.

[31] Mirdit Doda, Marcus Huber, Gláucia Murta, Matej Pivoluska, Martin Plesch, and Chrysoula Vlachou. Quantum key distribution overcoming extreme noise: Simultaneous subspace coding using high-dimensional entanglement. *Phys. Rev. Applied*, 15: 034003, 2021. DOI: 10.1103/PhysRevApplied.15.034003.

[32] Nico Döttling, Satrajit Ghosh, Jesper Buus Nielsen, Tobias Nilges, and Roberto Trifiletti. Tinyole: Efficient actively secure two-party computation from oblivious linear function evaluation. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, page 2263–2276, New York, NY, USA, 2017. Association for Computing Machinery. DOI: 10.1145/3133956.3134024.

[33] Frederic Dupuis, Omar Fawzi, and Stephanie Wehner. Entanglement sampling and applications. *IEEE Transactions on Information Theory*, 61(2):1093–1112, 2015. DOI: 10.1109/tit.2014.2371464.

[34] Thomas Durt, Berthold-Georg Englert, Ingemar Bengtsson, and Karol Zyczkowski. On mutually unbiased bases. *International Journal of Quantum Information*, 08(04): 535–640, 2010. DOI: 10.1142/S0219749910006502.

[35] Nico Döttling, Daniel Kraschewski, and Jörn Müller-Quade. David and goliath oblivious affine function evaluation - asymptotically optimal building blocks for universally composable two-party computation from a single untrusted stateful tamper-proof hardware token. Cryptology ePrint Archive, Report 2012/135, 2012. URL `https://eprint.iacr.org/2012/135`.

[36] Serge Fehr and Christian Schaffner. Composing quantum protocols in a classical environment. *Theory of Cryptography Conference (TCC 09)*, 5444:350–367, 2009. DOI: 10.48550/arXiv.0804.1059.

[37] Satrajit Ghosh and Tobias Nilges. An algebraic approach to maliciously secure private set intersection. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019*, pages 154–185, Cham, 2019. Springer International Publishing. DOI: 10.1007/978-3-030-17659-4_6.

[38] Satrajit Ghosh, Jesper Buus Nielsen, and Tobias Nilges. Maliciously secure oblivious linear function evaluation with constant overhead. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017*, pages 629–659, Cham, 2017. Springer International Publishing. DOI: 10.1007/978-3-319-70694-8_22.

[39] Niv Gilboa. Two party RSA key generation. In *Advances in Cryptology — CRYPTO' 99*, pages 116–129. Springer Berlin Heidelberg, 1999. DOI: 10.1007/3-540-48405-1_8.

[40] Oded Goldreich. *Foundations of Cryptography*. Cambridge University Press, 2004. DOI: 10.1017/cbo9780511721656.

[41] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, page 218–229, New York, NY, USA, 1987. Association for Computing Machinery. DOI: 10.1145/28395.28420.

[42] Shai Halevi and Hugo Krawczyk. MMH: Software message authentication in the gbit/second rates. In *Fast Software Encryption*, pages 172–189. Springer Berlin Heidelberg, 1997. DOI: 10.1007/bfb0052345.

[43] Carmit Hazay. Oblivious polynomial evaluation and secure set-intersection from algebraic prfs. *J. Cryptol.*, 31(2):537–586, 2018. DOI: 10.1007/s00145-017-9263-y.

[44] Carmit Hazay, Yuval Ishai, Antonio Marcedone, and Muthuramakrishnan Venkitasubramaniam. Leviosa: Lightweight secure arithmetic computation. CCS '19, page 327–344. Association for Computing Machinery, 2019. DOI: 10.1145/3319535.3354258.

[45] Russel Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, page 44–61, New York, NY, USA, 1989. Association for Computing Machinery. ISBN 0897913078. DOI: 10.1145/73007.73012.

[46] Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. Secure arithmetic computation with no honest majority. In Omer Reingold, editor, *Theory of Cryptography*, pages 294–314, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. DOI: 10.1007/978-3-642-00457-5_18.

[47] I. Kerenidis and A. Chailloux. Optimal bounds for quantum bit commitment. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 354–362, Los Alamitos, CA, USA, oct 2011. IEEE Computer Society. DOI: 10.1109/FOCS.2011.42.

[48] Joe Kilian. Founding cryptography on oblivious transfer. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 20–31, New York, NY, USA, 1988. Association for Computing Machinery. DOI: 10.1145/62212.62215.

[49] Robert Konig, Stephanie Wehner, and Jürg Wullschleger. Unconditional security from noisy quantum storage. *IEEE Transactions on Information Theory*, 58(3):1962–1984, 2012. DOI: 10.1109/TIT.2011.2177772.

[50] Srijita Kundu, Jamie Sikora, and Ernest Y. Z. Tan. A device-independent protocol for xor oblivious transfer. *Quantum*, 6:735, 2022. DOI: 10.22331/q-2022-05-30-725.

[51] Charles Ci Wen Lim, Christopher Portmann, Marco Tomamichel, Renato Renner, and Nicolas Gisin. Device-independent quantum key distribution with local bell test. *Physical Review X*, 3(3), 2013. DOI: 10.1103/physrevx.3.031006.

[52] Lindell and Pinkas. Privacy preserving data mining. *Journal of Cryptology*, 15(3): 177–206, 2002. DOI: 10.1007/s00145-001-0019-2.

[53] Hoi-Kwong Lo and H. F. Chau. Is quantum bit commitment really possible? *Phys. Rev. Lett.*, 78:3410–3413, Apr 1997. DOI: 10.1103/PhysRevLett.78.3410.

[54] Hoi-Kwong Lo and H.F. Chau. Why quantum bit commitment and ideal quantum coin tossing are impossible. *Physica D: Nonlinear Phenomena*, 120(1):177–187, 1998. DOI: 10.1016/S0167-2789(98)00053-0.

[55] Ricardo Loura, Álvaro J. Almeida, Paulo S. André, Armando N. Pinto, Paulo Mateus, and Nikola Paunković. Noise and measurement errors in a practical two-state quantum bit commitment protocol. *Phys. Rev. A*, 89:052336, 2014. DOI: 10.1103/PhysRevA.89.052336.

[56] Ricardo Loura, Du šan Arsenović, Nikola Paunković, Duška B. Popović, and Slobodan Prvanović. Security of two-state and four-state practical quantum bit-commitment protocols. *Phys. Rev. A*, 94:062335, 2016. DOI: 10.1103/PhysRevA.94.062335.

[57] Dominic Mayers. Unconditionally secure quantum bit commitment is impossible. *Phys. Rev. Lett.*, 78:3414–3417, 1997. DOI: 10.1103/PhysRevLett.78.3414.

[58] Carlos Mochon. Large family of quantum weak coin-flipping protocols. *Phys. Rev. A*, 72:022341, 2005. DOI: 10.1103/PhysRevA.72.022341.

[59] G. Molina-Terriza, A. Vaziri, R. Ursin, and A. Zeilinger. Experimental quantum coin tossing. *Phys. Rev. Lett.*, 94:040501, 2005. DOI: 10.1103/PhysRevLett.94.040501.

[60] Martin Müller-Lennert, Frédéric Dupuis, Oleg Szehr, Serge Fehr, and Marco Tomamichel. On quantum rényi entropies: A new generalization and some properties. *Journal of Mathematical Physics*, 54(12):122203, 2013. DOI: 10.1063/1.4838856.

[61] Moni Naor and Benny Pinkas. Oblivious transfer and polynomial evaluation. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing - STOC '99*. ACM Press. DOI: 10.1145/301250.301312.

[62] Moni Naor and Benny Pinkas. Oblivious polynomial evaluation. *SIAM Journal on Computing*, 35(5):1254–1281, 2006. DOI: 10.1137/s0097539704383633.

[63] Nelly Huei Ying Ng, Siddarth K. Joshi, Chia Chen Ming, Christian Kurtsiefer, and Stephanie Wehner. Experimental implementation of bit commitment in the noisy-storage model. *Nature Communications*, 3(1):1326, 2012. DOI: 10.1038/ncomms2268.

[64] Dimiter Ostrev. QKD parameter estimation by two-universal hashing. *Quantum*, 7: 894, 2023. DOI: 10.22331/q-2023-01-13-894.

[65] Anna Pappa, Paul Jouguet, Thomas Lawson, André Chailloux, Matthieu Legré, Patrick Trinkler, Iordanis Kerenidis, and Eleni Diamanti. Experimental plug and play quantum coin flipping. *Nature Communications*, 5(1):3717, 2014. DOI: 10.1038/ncomms4717.

[66] Michael O. Rabin. How to exchange secrets with oblivious transfer. Technical Report TR-81, Aiken Computation Laboratory, Harvard University, 1981. URL `https://eprint.iacr.org/2005/187`.

[67] Renato Renner. Security of quantum key distribution. *International Journal of Quantum Information*, 06(01):1–127, 2008. DOI: 10.1142/S0219749908003256.

[68] Manuel B. Santos, Paulo Mateus, and Armando N. Pinto. Quantum oblivious transfer: A short review. *Entropy*, 24(7), 2022. DOI: 10.3390/e24070945.

[69] Lana Sheridan and Valerio Scarani. Security proof for quantum key distribution using qudit systems. *Phys. Rev. A*, 82:030301, 2010. DOI: 10.1103/PhysRevA.82.030301.

[70] Peter W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994. DOI: 10.1109/SFCS.1994.365700.

[71] Alicia Sit, Frédéric Bouchard, Robert Fickler, Jérémie Gagnon-Bischoff, Hugo Larocque, Khabat Heshami, Dominique Elser, Christian Peuntinger, Kevin Günthner, Bettina Heim, Christoph Marquardt, Gerd Leuchs, Robert W. Boyd, and Ebrahim Karimi. High-dimensional intracity quantum cryptography with structured photons. *Optica*, 4(9):1006–1010, 2017. DOI: 10.1364/OPTICA.4.001006.

[72] Lara Stroh, Nikola Horová, Robert Stárek, Ittoop V. Puthoor, Michal Mičuda, Miloslav Dušek, and Erika Andersson. Noninteractive xor quantum oblivious transfer: Optimal protocols and their experimental implementations. *PRX Quantum*, 4:020320, 2023. DOI: 10.1103/PRXQuantum.4.020320.

[73] Marco Tomamichel. *Quantum Information Processing with Finite Resources*. Springer International Publishing, 2016. DOI: 10.1007/978-3-319-21891-5.

[74] Marco Tomamichel, Christian Schaffner, Adam Smith, and Renato Renner. Leftover hashing against quantum side information. *IEEE Transactions on Information Theory*, 57(8):5524–5535, 2011. DOI: 10.1109/tit.2011.2158473.

[75] Dominique Unruh. Simulatable security for quantum protocols. *arXiv*, 0409125, 2004. DOI: 10.48550/arXiv.quant-ph/0409125.

[76] Dominique Unruh. Universally composable quantum multi-party computation. In *Advances in Cryptology – EUROCRYPT 2010*, pages 486–505, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. DOI: 0.1007/978-3-642-13190-5_25.

[77] Dominique Unruh. Concurrent composition in the bounded quantum storage model. In *Advances in Cryptology – EUROCRYPT 2011*, pages 467–486. Springer Berlin Heidelberg, 2011. DOI: 10.1007/978-3-642-20465-4_26.

[78] Dominique Unruh. Lecture notes in quantum cryptography, 2022. URL https://kodu.ut.ee/~unruh/courses/qc/2022/notes.pdf.

[79] Tian Zhong, Hongchao Zhou, Robert D Horansky, Catherine Lee, Varun B Verma, Adriana E Lita, Alessandro Restelli, Joshua C Bienfang, Richard P Mirin, Thomas Gerrits, Sae Woo Nam, Francesco Marsili, Matthew D Shaw, Zheshen Zhang, Ligong Wang, Dirk Englund, Gregory W Wornell, Jeffrey H Shapiro, and Franco N C Wong. Photon-efficient quantum key distribution using time–energy entanglement with high-dimensional encoding. *New Journal of Physics*, 17(2):022002, 2015. DOI: 10.1088/1367-2630/17/2/022002.

A Security of the RWOLE protocol – Proofs

A.1 Proof of Lemma 4.1 (Dishonest Bob)

As we mentioned in the main text, this proof is a combination and adaptation to our case of results from [29] and [33].

To simplify the notation, in this proof we drop the subscript 0 that refers to the RWOLE phase, e.g. we write Alice's function vector $\mathbf{F}_0$, simply as $\mathbf{F}$.

Let the values that Bob commits be fixed as $(x_i, r_i)\,\forall i \in [m]$, where $m = (1+t)n$ and tn the number of qudits $\left|e_{r_i}^{x_i}\right\rangle$ used in the *Test Phase* to check whether he is honest or not. Throughout the proof we denote by $\boldsymbol{x} = (x_1, \ldots, x_m)$ and $\boldsymbol{r} = (r_1, \ldots, r_m)$ the vectors in $\mathbb{Z}_d^m$ whose components contain Bob's commitments $\forall i \in [m]$, and by $X^m = (X_1, \ldots, X_m)$ the vector of the random variables associated to $x_i, i \in [m]$. For each pair (x_i, r_i), the corresponding qudit $\left|e_{r_i}^{x_i}\right\rangle$ belongs in the Hilbert space $\mathcal{H}_{X_i}$, and the quantum system including all the qudits is in $\mathcal{H}_{X^m} = \bigotimes_{i\in[m]} \mathcal{H}_{X_i}$. For simplicity, we refer to the quantum systems in terms of the corresponding random variables X_i, instead of the Hilbert spaces $\mathcal{H}_{X_i}$.

Recall that the set $T \subset [m]$ contains the tn indices of the test qudits, and by $\bar{T}$ we denote its complement $[m]\backslash T$. For $i \in T$, $\boldsymbol{x}_{|T}$ is the vector whose components are the bases x_i in which Alice will measure the test qudits, and $\boldsymbol{r'}_{|T}$ is the vector whose components are Alice's measurement results. The corresponding quantum system is in the Hilbert space $\bigotimes_{i\in T} \mathcal{H}_{X_i}$, which for simplicity we denote as $X_{|T}^m$ in terms of the associated random variables. Finally, $r_H(\cdot,\cdot) = d_H(\cdot,\cdot)/n$ is the relative Hamming distance between two vectors of size n, with $d_H(\cdot,\cdot)$ being their Hamming distance.

Proof. Let us start by proving property 1. of Lemma 4.1. After the first step of π_{RWOLE}^n (Figure 5), the generated state is $\rho_{X^m E}$, where E is an auxiliary quantum system that Bob holds. Without loss of generality we assume that $\rho_{X^m E} = |\phi_{X^m E}\rangle\langle\phi_{X^m E}|$, i.e, it is a pure state[7]. If Bob is honest, we have that $|\phi_{X^m E}\rangle = |e_{\boldsymbol{r}}^{\boldsymbol{x}}\rangle \otimes |\psi_E\rangle$, i.e., Bob's auxiliary quantum system E is not entangled to the states that he sends to Alice.

The *Test Phase* of the protocol is used to guarantee that the real state is close to an ideal state that satisfies the properties 1. and 2. of Lemma 4.1. Let $\boldsymbol{r'}$ be the vector whose components are Alice's outcomes when measuring the state of X^m in the committed bases

[7] Otherwise, we purify it and carry the purification system along with E.

$\boldsymbol{x}$, and let $\mathcal{T}$ be the random variable associated to the set of indexes T of size tn. We can consider the state:

$$\rho_{\mathcal{T}X^mE} = \rho_{\mathcal{T}} \otimes |\phi_{X^mE}\rangle\langle\phi_{X^mE}| = \sum_T P_{\mathcal{T}}(T)\,|T\rangle\langle T| \otimes |\phi_{X^mE}\rangle\langle\phi_{X^mE}|\,, \tag{66}$$

to be the state resulting from the real execution of the protocol, and prove that it is close to some state, $\sigma_{\mathcal{T}X^mE}$, that fulfils the following property: for any choice of T and for any outcome $\boldsymbol{r'}_{|T}$ when measuring the state of ${X^m}_{|T}$ in the bases $\boldsymbol{x}_{|T}$, the relative error $r_H(\boldsymbol{r'}_{|T}, \boldsymbol{r}_{|T})$ is an upper bound on the relative error $r_H(\boldsymbol{r'}_{|\bar{T}}, \boldsymbol{r}_{|\bar{T}})$, which one would obtain by measuring the remaining subsystems ${X^m}_{|\bar{T}}$ in the bases $\boldsymbol{x}_{|\bar{T}}$. This state, $\sigma_{\mathcal{T}X^mE}$, can be written as:

$$\sigma_{\mathcal{T}X^mE} = \sum_T P_{\mathcal{T}}(T)\,|T\rangle\langle T| \otimes \left|\tilde{\phi}^T_{X^mE}\right\rangle\!\left\langle\tilde{\phi}^T_{X^mE}\right|, \tag{67}$$

where $\forall T$,

$$\left|\tilde{\phi}^T_{X^mE}\right\rangle = \sum_{\boldsymbol{r'}\in\mathcal{B}_T} \alpha^T_{\boldsymbol{r'}}\,|e^{\boldsymbol{x}}_{\boldsymbol{r'}}\rangle \otimes \left|\psi^{\boldsymbol{r'}}_E\right\rangle, \tag{68}$$

for $\mathcal{B}_T = \{\boldsymbol{r'} \in \mathbb{Z}_d^m : r_H(\boldsymbol{r'}_{|\bar{T}}, \boldsymbol{r}_{|\bar{T}}) \leq r_H(\boldsymbol{r'}_{|T}, \boldsymbol{r}_{|T}) + \zeta\}$ for $\zeta > 0$ and arbitrary coefficients $\alpha^T_{\boldsymbol{r'}}$. The state $\left|\psi^{\boldsymbol{r'}}_E\right\rangle$ is an arbitrary state on E subsystem that possibly depends on $\boldsymbol{r'}$. Then we have:

Lemma A.1. Let the quantum states $\rho_{\mathcal{T}X^mE}$ and $\sigma_{\mathcal{T}X^mE}$ be given by (66) and (67), respectively. Then, $\forall\zeta > 0$ and fixed strings $\boldsymbol{x}, \boldsymbol{r} \in \mathbb{Z}_d^m$, we have that

$$\rho_{\mathcal{T}X^mE} \approx_\epsilon \sigma_{\mathcal{T}X^mE}, \tag{69}$$

where $\epsilon(\zeta, n) = \epsilon(\zeta, n) = \exp\left(-\frac{2\zeta^2t^2n^2}{(nt+1)(t+1)}\right)$. That is, the real state $\rho_{\mathcal{T}X^mE}$ is exponentially close, with respect to n, to the ideal state $\sigma_{\mathcal{T}X^mE}$.

Proof. This proof is an adaptation of the proof of Lemma 4.3 from [29] to our case.

For any T, let $\left|\tilde{\phi}^T_{X^mE}\right\rangle$ be the renormalized projection of $|\phi_{X^mE}\rangle$ into the subspace

$$\text{Span}\,\{|e^{\boldsymbol{x}}_{\boldsymbol{r'}}\rangle : \boldsymbol{r'} \in \mathcal{B}_T\} \otimes \mathcal{H}_E,$$

and let $\left|\tilde{\phi}^{T^\perp}_{X^mE}\right\rangle$ be the renormalized projection of $|\phi_{X^mE}\rangle$ into its orthogonal complement.

We can, then, write

$$|\phi_{X^mE}\rangle = \epsilon_T \left|\tilde{\phi}^T_{X^mE}\right\rangle + \epsilon^\perp_T \left|\tilde{\phi}^{T^\perp}_{X^mE}\right\rangle, \tag{70}$$

with $\epsilon_T = \left\langle\tilde{\phi}^T_{X^mE}\middle|\phi_{X^mE}\right\rangle$ and $\epsilon^\perp_T = \left\langle\tilde{\phi}^{T^\perp}_{X^mE}\middle|\phi_{X^mE}\right\rangle$. By construction, this state satisfies (68). Furthermore, we can calculate the distance:

$$\delta(\,|\phi_{X^mE}\rangle\langle\phi_{X^mE}|\,, \left|\tilde{\phi}^T_{X^mE}\right\rangle\!\left\langle\tilde{\phi}^T_{X^mE}\right|) = \sqrt{1 - |\left\langle\tilde{\phi}^T_{X^mE}\middle|\phi_{X^mE}\right\rangle|^2} = \sqrt{1 - |\epsilon_T|^2} = |\epsilon^\perp_T|, \tag{71}$$

where, given T, $|\epsilon_T^{\perp}|$ is the probability amplitude for getting outcome $\boldsymbol{r'} \notin \mathcal{B}_T$ when measuring the state of X^m in bases $\boldsymbol{x}$. We continue to derive an upper bound on the distance between the real and the ideal state:

$$\delta(\rho_{\mathcal{T}X^mE}, \sigma_{\mathcal{T}X^mE}) = \left(\sum_T P_{\mathcal{T}}(T) \delta\left(|\phi_{X^mE}\rangle\langle\phi_{X^mE}| , \left|\tilde{\phi}^T_{X^mE}\right\rangle\left\langle\tilde{\phi}^T_{X^mE}\right| \right) \right)^2 \leq \sum_T P_{\mathcal{T}}(T) |\epsilon_T^{\perp}|^2, \tag{72}$$

where we used Jensen's inequality and properties of the trace norm. The last term is the probability that, when choosing T according to $P_{\mathcal{T}}$ and measuring the state of X^m in bases $\boldsymbol{x}$ we get an outcome $\boldsymbol{r'} \notin \mathcal{B}_T$. We write

$$\sum_T P_{\mathcal{T}}(T) |\epsilon_T^{\perp}|^2 = \Pr_{\mathcal{T}}[\boldsymbol{r'} \notin \mathcal{B}_T] = \Pr_{\mathcal{T}}[r_H(\boldsymbol{r'}_{|\bar{T}}, \boldsymbol{r}_{|\bar{T}}) - r_H(\boldsymbol{r'}_{|T}, \boldsymbol{r}_{|T}) > \zeta]. \tag{73}$$

Then, we can use Corollary 4 from [51], which states that the above probability is negligible in n and gives us an upper bound for $\delta(\rho_{\mathcal{T}X^mE}, \sigma_{\mathcal{T}X^mE})$. In particular, given the set $[m]$ with $(1+t)n$ elements, we apply the aforementioned corollary for a random subset $\mathcal{T}$ of size tn and its complement $\bar{\mathcal{T}}$ of size n. Denoting by $\mu_{\mathcal{T}}$ and $\mu_{\bar{\mathcal{T}}}$, respectively, the averages of these subsets, we obtain

$$\Pr[\mu_{\bar{\mathcal{T}}} - \mu_{\mathcal{T}} \geq \zeta] \leq \exp\left(-\frac{2\zeta^2 t^2 n^2}{(nt+1)(t+1)} \right). \tag{74}$$

Hence, we have:

$$\delta(\rho_{\mathcal{T}X^mE}, \sigma_{\mathcal{T}X^mE}) \leq \sum_T P_{\mathcal{T}}(T) |\epsilon_T^{\perp}|^2 \leq \exp\left(-\frac{2\zeta^2 t^2 n^2}{(nt+1)(t+1)} \right) =: \epsilon, \tag{75}$$

concluding the proof of Lemma A.1, i.e. that the real state (66) generated by the protocol until the *Computation Phase* is $\epsilon-$close to the ideal state (67). □

It is now straightforward to complete the proof of property 1. of Lemma 4.1.

To obtain the states $\sigma_{\mathbf{F}B'}$ and $\rho_{\mathbf{F}B'}$ from the states $\sigma_{\mathcal{T}X^mE}$ and $\rho_{\mathcal{T}X^mE}$, respectively, we need to apply the operator $V_{\boldsymbol{a}}^{\boldsymbol{b}} \operatorname{Tr}_{X^m_{|T}E_{|T}}[\,\cdot\,] V_{\boldsymbol{a}}^{\boldsymbol{b}\dagger}$. This operator is a CPTP map, being the composition of the two CPTP maps, $V_{\boldsymbol{a}}^{\boldsymbol{b}}$ and Tr. Since the trace distance between two density matrices does not increase under CPTP maps (see Lemma 7 in [78]), the final states indeed satisfy property 1. of Lemma 4.1, namely

$$\sigma_{\mathbf{F}B'} \approx_{\epsilon} \rho_{\mathbf{F}B'}. \tag{76}$$

For the rest of this proof dishonest Bob's system B' is identified with $\mathbf{Y}\,E$, where $\mathbf{Y}$ corresponds to the classical information leaking to him through the output of the WROLE and E is, in general, a quantum auxiliary system that he might also hold. Consequently, from now on we write $\sigma_{\mathbf{F}B'}$ as $\sigma_{\mathbf{FY}E}$. Now, we have to prove property 2. of Lemma 4.1, i.e., obtain the corresponding lower bound on min-entropy with respect to $\sigma_{\mathbf{FY}E}$. We start with the following Lemma A.2:

Lemma A.2 (Corollary 4.4 in [29])**.** Let $err := r_H(\boldsymbol{r'}_{|T}, \boldsymbol{r}_{|T}) \leq 1 - \frac{1}{d}$ be the error measured by Alice while measuring the state of $X^m_{|T}$ according to her choice of T, and let $\sigma_{\mathbf{X}E} := |\psi\rangle\langle\psi|_{\mathbf{X}E}$ be the state to which the ideal state $\sigma_{\mathcal{T}X^mE}$ collapses after this measurement.

Following (67) and (68), we write $|\psi\rangle_{\mathbf{X}E} = \sum_{\boldsymbol{z}\in\mathcal{B}} \alpha_{\boldsymbol{z}} |e_{\boldsymbol{z}}^{\boldsymbol{x}}\rangle |\psi_E^{\boldsymbol{z}}\rangle$ for some $|\psi_E^{\boldsymbol{z}}\rangle$ and $\mathcal{B} = \{\boldsymbol{z} \in \mathbb{Z}_d^n : r_H(\boldsymbol{z}, \boldsymbol{r}_{|\bar{T}}) \leq err + \zeta\}$ with $\zeta > 0$. Then, we have:

$$H_{\min}(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}} \geq -h_d(err + \zeta) n \log d, \tag{77}$$

where $h_d(x)$ is given in Definition 2.2.

Proof. We start by defining the state $\tilde{\sigma}_{\mathbf{X}E} := \sum_{\boldsymbol{z}\in\mathcal{B}} |\alpha_{\boldsymbol{z}}|^2 |e_{\boldsymbol{z}}^{\boldsymbol{x}}\rangle\langle e_{\boldsymbol{z}}^{\boldsymbol{x}}| \otimes |\psi_E^{\boldsymbol{z}}\rangle\langle\psi_E^{\boldsymbol{z}}|$. Then, by applying Lemma 3.1.13 from [67], we obtain

$$H_{\min}(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}} \geq H_{\min}(\mathbf{X}|E)_{\tilde{\sigma}_{\mathbf{X}E}} - \log|\mathcal{B}|. \tag{78}$$

Since $\tilde{\sigma}_{\mathbf{X}E}$ is a classical-quantum state, its min-entropy cannot be negative, that is $H_{\min}(\mathbf{X}|E)_{\tilde{\sigma}_{\mathbf{X}E}} \geq 0$, thus $H_{\min}(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}} \geq -\log|\mathcal{B}|$. The lower bound shown in (77) follows directly from Definition 2.2 considering the Hamming ball around $\boldsymbol{r}_{|\bar{T}}$ with radius $n(err + \zeta)$. □

To complete the proof of property 2. of Lemma 4.1 and find a lower bound on $H_{\min}(\mathbf{F}|\mathbf{Y}\,E)\sigma_{\mathbf{F}\mathbf{Y}E}$ we need to relate it with $H_{\min}(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}$, for which we just derived a lower bound. In what follows, we adapt the notation from [33]:

- $\Phi_{\mathbf{X}\bar{\mathbf{X}}} = |\Phi\rangle\langle\Phi|_{\mathbf{X}\bar{\mathbf{X}}}$, where $|\Phi\rangle_{\mathbf{X}\bar{\mathbf{X}}} = \sum_{\boldsymbol{s}} |e_{\boldsymbol{s}}^{\boldsymbol{x}}\rangle_{\mathbf{X}} \otimes |e_{\boldsymbol{s}}^{\boldsymbol{x}}\rangle_{\bar{\mathbf{X}}}$ for all basis choices $\boldsymbol{x} \in \mathbb{Z}_d^n$, and
- $\Phi_{(\boldsymbol{a},\boldsymbol{b})} = \left|\Phi_{(\boldsymbol{a},\boldsymbol{b})}\right\rangle\!\left\langle\Phi_{(\boldsymbol{a},\boldsymbol{b})}\right| = \sum_{\boldsymbol{s}\boldsymbol{s}'} V_{\boldsymbol{a}}^{\boldsymbol{b}} |e_{\boldsymbol{s}}^{\boldsymbol{x}}\rangle\langle e_{\boldsymbol{s}'}^{\boldsymbol{x}}| V_{\boldsymbol{a}}^{\boldsymbol{b}\dagger} \otimes |e_{\boldsymbol{s}}^{\boldsymbol{x}}\rangle\langle e_{\boldsymbol{s}'}^{\boldsymbol{x}}|$, with $\left|\Phi_{(\boldsymbol{a},\boldsymbol{b})}\right\rangle = (V_{\boldsymbol{a}}^{\boldsymbol{b}} \otimes \mathbb{1}) |\Phi\rangle_{\mathbf{X}\bar{\mathbf{X}}}$, for $(\boldsymbol{a},\boldsymbol{b}) \in \mathbb{Z}_d^2$

and we use the following properties of the operators V_a^b, which can be derived from (16):

(a) $V_a^b |e_i^x\rangle\langle e_i^x| V_a^{b\dagger} = \left|e_{ax-b+i}^x\right\rangle\!\left\langle e_{ax-b+i}^x\right|$,

(b) $V_a^b = \sum_j c_{a,b,j,x} \left|e_{ax-b+j}^x\right\rangle\!\left\langle e_j^x\right|$,

(c) $V_a^{b\dagger} = \sum_j c_{a,b,j,x}^* \left|e_j^x\right\rangle\!\left\langle e_{ax-b+j}^x\right|$.

The following lemma, which is an adaptation of Theorem 12 in [33] to our case, provides a lower bound for $H_{\min}(\mathbf{F}|\mathbf{Y}\,E)_{\sigma_{\mathbf{F}\mathbf{Y}E}}$ in terms of $H_{\min}(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}$.

Lemma A.3. Let $\mathbf{X}$ denote our n-qudit system and $\sigma_{\mathbf{X}E}$ be the ideal quantum state to which the system collapsed after Alice's test measurements, as introduced before. Then, we have:

$$\mathrm{H}_{\min}(\mathbf{F}|\mathbf{Y}\,E)_{\sigma_{\mathbf{F}\mathbf{Y}E}} \geq \frac{1}{2}(n \log d + \mathrm{H}_{\min}(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}), \tag{79}$$

where

$$\sigma_{\mathbf{F}\mathbf{Y}E} = \frac{1}{d^{2n}} \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathbb{Z}_d^{2n}} |e_{\boldsymbol{a}}^{\boldsymbol{x}}, e_{\boldsymbol{b}}^{\boldsymbol{x}}\rangle\langle e_{\boldsymbol{a}}^{\boldsymbol{x}}, e_{\boldsymbol{b}}^{\boldsymbol{x}}| \otimes V_{\boldsymbol{a}}^{\boldsymbol{b}}\, \sigma_{\mathbf{X}E}\, V_{\boldsymbol{a}}^{\boldsymbol{b}\dagger}, \tag{80}$$

is the state obtained when $V_{\boldsymbol{a}}^{\boldsymbol{b}}$ is applied to the system $\mathbf{X}$ according to $\mathbf{F}$.

Proof. This proof is an adaptation of the proof of Theorem 12 in [33] to our case. Let us fix $x \in \mathbb{Z}_d$ and write $|i\rangle = |e_i^x\rangle$ for short. V_a^b is a CPTP map, and it is known that the min-entropy does not decrease whenever a CPTP map is applied. However, this is not enough to prove the security of the protocol and determine a lower bound. We need a more refined expression relating $H_{\min}(\mathcal{M}(\mathbf{X})|E)$ and $H_{\min}(\mathbf{X}|E)$ for some CPTP map $\mathcal{M}$. This is given by the following Lemma A.4, which is an adaptation of Theorem 1 in [33] to our case:

Lemma A.4 (Theorem 1, [33])**.** Let $\mathbf{X}$ denote our system of n qudits, and $\mathcal{M}_{\mathbf{X}\to\mathbf{FY}}$ be a CP map such that $((\mathcal{M}^\dagger \circ \mathcal{M})_{\mathbf{X}} \otimes \mathrm{id}_{\bar{\mathbf{X}}})(\Phi_{\mathbf{X}\bar{\mathbf{X}}}) = \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathbb{Z}_d^{2n}} \lambda_{(\boldsymbol{a},\boldsymbol{b})}\Phi_{(\boldsymbol{a},\boldsymbol{b})}$. As above, let $\sigma_{\mathbf{X}E}$ be the ideal quantum state to which the system collapsed after Alice's test measurements. Then, for any partition of $\mathbb{Z}_d^{2n} = \mathfrak{S}_+ \cup \mathfrak{S}_-$ into subsets $\mathfrak{S}_+$ and $\mathfrak{S}_-$, we have

$$2^{-\mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}|\sigma_{\mathbf{X}E}}} \leq \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathfrak{S}_+} \lambda_{(\boldsymbol{a},\boldsymbol{b})} 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}} + (\max_{(\boldsymbol{a},\boldsymbol{b})\in\mathfrak{S}_-} \lambda_{(\boldsymbol{a},\boldsymbol{b})}) d^n, \tag{81}$$

where, in general, for a (not necessarily normalized) quantum state $\rho_{AB} \in \mathcal{P}(\mathcal{H}_A \otimes \mathcal{H}_B)$, $\mathrm{H}_2(A|B)$ is the so-called *collision entropy* [67], given as

$$\mathrm{H}_2(A|B)_{\rho_{AB}} = -\log\left(\mathrm{Tr}\left\{\left(\rho_B^{-1/4}\rho_{AB}\rho_B^{-1/4}\right)^2\right\}\right). \tag{82}$$

If we further condition on a general quantum state $\sigma_B \in \mathcal{P}(\mathcal{H}_B)$, we have

$$\mathrm{H}_2(A|B)_{\rho_{AB}|\sigma_B} = -\log\left(\mathrm{Tr}\left\{\left(\sigma_B^{-1/4}\rho_{AB}\sigma_B^{-1/4}\right)^2\right\}\right). \tag{83}$$

We can apply the above Lemma A.4 with $\mathcal{M}_{\mathbf{X}\to\mathbf{FY}} = \mathcal{N}^{\otimes n}$, where

$$\mathcal{N}(\sigma_X) = \frac{1}{d^2}\sum_{(a,b)\in\mathbb{Z}_d^2} \mathcal{N}_{a,b}\sigma_X\mathcal{N}_{a,b}^\dagger = \frac{1}{d^2}\sum_{(a,b)\in\mathbb{Z}_d^2} \Big(|a,b\rangle \otimes V_a^b\Big)\sigma_X\Big(\langle a,b| \otimes V_a^{b\dagger}\Big). \tag{84}$$

The operator $\mathcal{N}$ applies the operator V_a^b to the single system X and saves its choice (a,b) to a new record in the F space. To continue, we want to write $((\mathcal{M}^\dagger \circ \mathcal{M})_{\mathbf{X}} \otimes \mathrm{id}_{\bar{\mathbf{X}}})(\Phi_{\mathbf{X}\bar{\mathbf{X}}})$ as a linear combination of $\Phi_{(\boldsymbol{a},\boldsymbol{b})}$, i.e.

$$((\mathcal{M}^\dagger \circ \mathcal{M})_{\mathbf{X}} \otimes \mathrm{id}_{\bar{\mathbf{X}}})(\Phi_{\mathbf{X}\bar{\mathbf{X}}}) = \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathbb{Z}^{2n}} \lambda_{(\boldsymbol{a},\boldsymbol{b})}\Phi_{(\boldsymbol{a},\boldsymbol{b})} \tag{85}$$

First, note that

$$\begin{aligned}\mathcal{N}(|i\rangle\langle j|) &= \frac{1}{d^2}\sum_{(a,b)\in\mathbb{Z}_d^2} \Big(|a,b\rangle \otimes V_a^b\Big)|i\rangle\langle j|\Big(\langle a,b| \otimes V_a^{b\dagger}\Big) \\ &\overset{(a)}{=} \frac{1}{d^2}\sum_{(a,b)\in\mathbb{Z}_d^2} |a,b\rangle\langle a,b| \otimes |ax-b+i\rangle\langle ax-b+j|\,,\end{aligned} \tag{86}$$

where we used the property (a). We proceed to compute $\mathcal{N}^\dagger \circ \mathcal{N}\,|i\rangle\langle j|$:

$$\begin{aligned}\mathcal{N}^\dagger \circ \mathcal{N}\,|i\rangle\langle j| &= \frac{1}{d^2}\sum_{(a',b')\in\mathbb{Z}_d^2} \mathcal{N}_{a'b'}^\dagger(\mathcal{N}\,|i\rangle\langle j|))\mathcal{N}_{a'b'} \\ &\overset{(86)}{=} \frac{1}{d^4}\sum_{(a',b'),(a,b)\in\mathbb{Z}_d^2} \Big(\langle a',b'| \otimes V_{a'}^{b'\dagger}\Big)|a,b\rangle\langle a,b| \otimes |ax-b+i\rangle\langle ax-b+j|\Big(|a',b'\rangle \otimes V_{a'}^{b'}\Big) \\ &\overset{(b),(c)}{=} \frac{1}{d^4}\sum_{(a,b)\in\mathbb{Z}_d^2} V_a^{b\dagger}\,|ax-b+i\rangle\langle ax-b+j|\,V_a^b = \frac{1}{d^4}\sum_{(a,b)\in\mathbb{Z}_d^2} |i\rangle\langle j| = \frac{1}{d^2}\,|i\rangle\langle j|\,,\end{aligned} \tag{87}$$

and

$$\begin{aligned}((\mathcal{N}^\dagger \circ \mathcal{N})_X \otimes \mathrm{id}_{\bar{X}})(\Phi_{X\bar{X}}) &= \sum_{i,j} \mathcal{N}^\dagger \circ \mathcal{N}(|i\rangle\langle j|) \otimes |i\rangle\langle j| \\ &\overset{\text{eq.(87)}}{=} \sum_{i,j} \Big(\frac{1}{d^2}\,|i\rangle\langle j|\Big) \otimes |i\rangle\langle j| \\ &= \frac{1}{d^2}\sum_{i,j} |i\rangle\langle j| \otimes |i\rangle\langle j| = \frac{1}{d^2}\Phi_{X\bar{X}}. \end{aligned} \tag{88}$$

Therefore, we have

$$((\mathcal{N}^\dagger \circ \mathcal{N})_X \otimes \mathrm{id}_{\bar{X}})(\Phi_{X\bar{X}}) = \frac{1}{d^2}\Phi_{(0,0)}, \tag{89}$$

from which we easily see

$$((\mathcal{M}^\dagger \circ \mathcal{M})_{\mathbf{X}} \otimes \mathrm{id}_{\bar{\mathbf{X}}})(\Phi_{\mathbf{X}\bar{\mathbf{X}}}) = \frac{1}{d^{2n}}\Phi_{(\mathbf{0},\mathbf{0})}. \tag{90}$$

Consequently,

$$\lambda_{(\boldsymbol{a},\boldsymbol{b})} = \begin{cases} \frac{1}{d^{2n}} & \text{if } (\boldsymbol{a},\boldsymbol{b}) = (\mathbf{0},\mathbf{0}) \\ 0 & \text{otherwise.} \end{cases} \tag{91}$$

Now, we want to choose the partition that gives us the best lower bound on the collision entropy, i.e. decreases the r.h.s of the following relation:

$$2^{-\mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}}} \leq \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathfrak{S}_+} \lambda_{(\boldsymbol{a},\boldsymbol{b})} 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}} + \big(\max_{(\boldsymbol{a},\boldsymbol{b})\in\mathfrak{S}_-} \lambda_{(\boldsymbol{a},\boldsymbol{b})}\big) d^n. \tag{92}$$

Note that we dropped the conditioning on the state $\sigma_{\mathbf{X}E}$ at $\mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}}$. This is because the map $\mathcal{M}$ is trace-preserving (for a detailed explanation see [33] below Theorem 1). For our case there are just two types of partitions: the case where $0 \in \mathfrak{S}_+$ and the case where $0 \in \mathfrak{S}_-$. If $0 \in \mathfrak{S}_+$:

$$\text{r.h.s} = \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathfrak{S}_+} \lambda_s 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}} = \frac{1}{d^{2n}} 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}} \leq \frac{1}{d^n}. \tag{93}$$

The last inequality holds because $-n\log d \leq \mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}} \leq n\log d$. In fact,

$$\frac{1}{d^{2n}} 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}} \leq \frac{1}{d^n} \iff 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}} \leq d^n \iff \mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}} \geq -n\log d. \tag{94}$$

If $0 \in \mathfrak{S}_-$, r.h.s $= \frac{1}{d^n}$ which, as we have just seen by the previous inequality, does not provide a better lower bound on the collision entropy whenever $\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}} \neq -n\log d$. So, choosing any partition such that $0 \in \mathfrak{S}_+$, we get

$$\begin{aligned} 2^{-\mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}}} &\leq \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathfrak{S}_+} \lambda_{(\boldsymbol{a},\boldsymbol{b})} 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}} + \big(\max_{(\boldsymbol{a},\boldsymbol{b})\in\mathfrak{S}_-} \lambda_{(\boldsymbol{a},\boldsymbol{b})}\big) d^n \\ &= \frac{1}{d^{2n}} 2^{-\mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}}, \end{aligned} \tag{95}$$

from which we conclude

$$\mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}} \geq 2n\log d + \mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}. \tag{96}$$

In order to relate $\mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}}$ with $\mathrm{H}_2(\mathbf{F}|\mathbf{Y}E)_{\sigma_{\mathbf{FY}E}}$, we use the chain rule proved in Proposition 8 of [60]:

$$\mathrm{H}_2(\mathbf{F}|\mathbf{Y}E)_{\sigma_{\mathbf{FY}E}} \geq \mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}} - \log \mathrm{rank}(\sigma_{\mathbf{Y}}) \geq \mathrm{H}_2(\mathbf{FY}|E)_{\sigma_{\mathbf{FY}E}} - n \log d, \tag{97}$$

since $\log \mathrm{rank}(\sigma_{\mathbf{Y}}) \leq n \log d$. Combining (96) and (97), we get the desired result:

$$\mathrm{H}_2(\mathbf{F}|\mathbf{Y}\,E)_{\sigma_{\mathbf{FY}E}} \geq n \log d + \mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}. \tag{98}$$

To express the above relation in terms of the min-entropy instead of the collision entropy, we start by noticing that the state given in (80) can be written as

$$\sigma_{\mathbf{FY}E} = \frac{1}{d^{2n}} \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathbb{Z}_d^{2n}} |\boldsymbol{a},\boldsymbol{b}\rangle\langle\boldsymbol{a},\boldsymbol{b}| \otimes V_{\boldsymbol{a}}^{\boldsymbol{b}}\, \sigma_{\mathbf{X}E}\, V_{\boldsymbol{a}}^{\boldsymbol{b}\dagger} = \frac{1}{d^{2n}} \sum_{(\boldsymbol{a},\boldsymbol{b})\in\mathbb{Z}_d^{2n}} |\boldsymbol{a},\boldsymbol{b}\rangle\langle\boldsymbol{a},\boldsymbol{b}| \otimes \sigma_{\mathbf{X}E}^{\boldsymbol{a},\boldsymbol{b}}, \tag{99}$$

which is a classical-quantum state. Therefore, we can use Lemma 18 from [33] to obtain

$$\mathrm{H}_{\min}(\mathbf{F}|\mathbf{Y}\,E)_{\sigma_{\mathbf{FY}E}} \geq \frac{1}{2}\left(n \log d + \mathrm{H}_2(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}\right). \tag{100}$$

Furthermore, $\sigma_{\mathbf{X}E}$ is a general quantum state, and from Lemma 17 in [33] we have

$$\mathrm{H}_{\min}(\mathbf{F}|\mathbf{Y}\,E)_{\sigma_{\mathbf{FY}E}} \geq \frac{1}{2}\left(n \log d + \mathrm{H}_{\min}(\mathbf{X}|E)_{\sigma_{\mathbf{X}E}}\right). \tag{101}$$

□

To complete the proof of property 2. of Lemma 4.1, we combine Lemma A.2 and Lemma A.3, and obtain:

$$H_{\min}(\mathbf{F}|\mathbf{Y}E)_{\sigma_{\mathbf{FY}E}} \geq \frac{1}{2}(n \log d - h_d(\zeta) n \log d) = \frac{n \log d}{2}(1 - h_d(\zeta)), \tag{102}$$

for $err = 0$. □

B Lemma B.1

Lemma B.1. Let $\rho_{XB} \in \mathcal{P}(\mathcal{H}_X \otimes \mathcal{H}_B)$ be a cq-state and let $f : \mathcal{X} \to \mathcal{X}$ be a fixed bijective function. Then,

$$H_{\min}(X|E)_\rho \leq H_{\min}(f(X)|B)_\rho. \tag{103}$$

Proof. Consider the unitary operator, $U = \sum_x |f(x)\rangle\langle x|$. Checking that U is indeed unitary:

$$UU^\dagger = \left(\sum_x |f(x)\rangle\langle x|\right)\left(\sum_{x'} |x'\rangle\langle f(x')|\right) = \sum_x |f(x)\rangle\langle f(x)| = I, \tag{104}$$

where in the last step we used the fact that the function f is a bijection. The same holds for $U^\dagger U = I$. Now, observe that

$$\begin{aligned} H_{\min}(f(X)|B) &= -\log \max_{\{M_x\}_x} \sum_x p_x \,\mathrm{tr}\left[M_x \rho_{f(x)}^B\right] \\ &= -\log \max_{\{M_x\}_x} \sum_x p_x \,\mathrm{tr}\left[M_x U \rho_x^B U^\dagger\right] \\ &= -\log \max_{\{M_x\}_x} \sum_x p_x \,\mathrm{tr}\left[U^\dagger M_x U \rho_x^B\right]. \end{aligned} \tag{105}$$

Note that $\{N_x\}_x = \left\{U^\dagger M_x U\right\}_x$ is also a POVM: they are all positive semidefinite operators and it sums up to unity. Therefore, we have that $\left\{U^\dagger M_x U\right\}_x$ can only decrease the space of possible POVMs, i.e

$$\max_{\{M_x\}_x} \sum_x p_x \operatorname{tr}\left[U^\dagger M_x U \rho_x^B\right] \leq \max_{\{M_x\}_x} \sum_x p_x \operatorname{tr}\left[M_x \rho_x^B\right]. \tag{106}$$

This means

$$H_{\min}(f(X)|B) \geq -\log \max_{\{M_x\}_x} \sum_x p_x \operatorname{tr}\left[M_x \rho_x^B\right] = H_{\min}(X|B). \tag{107}$$

□

C Derivation of (58)

The relation (58) can be easily deduced by considering the following property from [34] (Equation (2.56) in section 2.4.2):

$$\alpha_k^i \alpha_l^i = \alpha_{k\oplus l}^i \omega^{i\odot k\odot l}. \tag{108}$$

We have

$$V_a^b \left|e_r^i\right\rangle = \frac{1}{\sqrt{N}} \sum_{k,l=0}^{N-1} |k\oplus a\rangle\, \omega^{(k\oplus a)\odot b} \omega^{\ominus r\odot l} \langle k|l\rangle\, \alpha_{\ominus l}^{i*} \tag{109}$$

$$= \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} |l\rangle\, \omega^{l\odot b} \omega^{\ominus r\odot (l\ominus a)} \alpha_{\ominus(l\ominus a)}^{i*} \tag{110}$$

$$= \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} |l\rangle\, \omega^{l\odot b\ominus r\odot (l\ominus a)} \left(\omega^{\ominus(i\odot a\odot(\ominus l))} \alpha_a^i \alpha_{\ominus l}^i\right)^* \tag{111}$$

$$= \omega^{r\odot a} \alpha_a^{i*} \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} |l\rangle\, \omega^{l\odot b\ominus r\odot l} \omega^{\ominus(i\odot a\odot l)} \alpha_{\ominus l}^{i*} \tag{112}$$

$$= \omega^{r\odot a} \alpha_a^{i*} \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} |l\rangle\, \omega^{\ominus(i\odot a\ominus b\oplus r)\odot l} \alpha_{\ominus l}^{i*} \tag{113}$$

$$= \omega^{r\odot a} \alpha_a^{i*} \left|e_{i\odot a\ominus b\oplus r}^i\right\rangle. \tag{114}$$