Oblivious Transfer from Quantum One-way Functions

MSc Thesis *(Afstudeerscriptie)*

written by

Yilun Wang

under the supervision of **Dr. Léo Colisson** and **Prof. Dr. Ronald de Wolf**, and submitted to the Examinations Board in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:	**Members of the Thesis Committee:**
27 September 2024	Dr. Maria Aloni
	Dr. Léo Colisson
	Prof. Dr. Ronald de Wolf
	Prof. Dr. Christian Schaffner
	Dr. Florian Speelman

Institute for Logic, Language and Computation

Abstract

Secure Multi-party Computation (MPC), which allows multiple parties to jointly compute a function over their inputs while keeping the inputs private, is one of the important research directions in cryptography, and plays a vital role in fields like auctions and electronic votes.

Oblivious Transfer (OT) protocols are sufficient to construct MPC protocols. We provide a construction turning any (classical) Zero-Knowledge (ZK) protocol into a composable quantum Oblivious Transfer (OT) protocol, using weaker assumptions compared to previous works while keeping a protocol optimal in communication.

In particular, this construction only requires collision-resistant quantum one-way functions, instead of collision-resistant hiding hash functions, to build a 2-message quantum OT protocol in the random oracle model.

Internally, we rely on a quantum version of the Goldreich-Levin theorem that we generalize to arbitrary length-preserving one-way functions instead of one-way permutations. This theorem provides a way to generate a quantum hard-predicate that is used in the protocol to hide one bit of information without relying on the hiding property.

Acknowledgment

I would like to express my deepest gratitude to everyone who has contributed to the successful completion of this thesis.

First and foremost, I would like to thank my supervisors, Dr. Léo Colisson and Prof. Dr. Ronald de Wolf, for their invaluable guidance and support throughout this project. Their insights and expertise have been crucial in shaping this thesis, and their encouragement kept me motivated during the most challenging times.

I am also grateful to the members of my thesis committee, Dr. Maria Aloni, Prof. Dr. Christian Schaffner and Dr. Florian Speelman, for their thoughtful feedback, guidance, and constructive criticism, which greatly improved the quality of this work.

I would like to extend my heartfelt thanks to my colleagues and fellow researchers at ILLC for creating an inspiring and collaborative environment.

To my family and friends, your constant support, encouragement, and understanding throughout this challenging process have been my greatest source of strength.

Thank you all for being part of this journey.

Contents

Chapter 1

Introduction

Multi-Party Computation (MPC) protocols allow multiple parties to jointly compute a function over their input while keeping the inputs private. MPC protocols can play a vital role in fields like auctions, electronic voting, and more. A typical example of MPC is the Millionaires' problem [Yao82], where a set of parties gets to know the identity of the richest person, in such a way that the fortune of each party is never leaked to others.

MPC protocols are usually quite complicated to study directly. However, it is proven that there is a simpler primitive called Oblivious Transfer (OT) which is sufficient for constructing multi-party computation (MPC) [Kil88]. An OT protocol is a protocol in which a sender holds two messages: the receiver can choose to learn one of these messages, without revealing their choice to the sender.

Until now, classical OT protocols rely on trapdoor functions. Trapdoor functions are functions that are easy to compute and hard to invert, but easy to invert with the knowledge of a secret trapdoor. One can construct trapdoor functions assuming the hardness of certain highly structured problems such as factoring [EGL85], or by relying on the problem of finding the shortest vector in a lattice [PVW08, BD18, Qua20]. In other words, classically, OT lives in Cryptomania [Imp95], the possible world where public-key cryptography is possible. On the other hand, information hidden in quantum states can only be revealed using some "correct" operations, and will be destroyed by "incorrect" ones. This feature makes it possible for OT to exist under weaker assumptions. As a result, quantumly, OT lives in MiniQcrypt [GLS+21, BCK+21], the possible world where one-way functions exist but where public-key cryptography is not known to be possible. The article [CMS23] provides the

first OT construction in MiniQcrypt requiring the exchange of only 2 messages (which is optimal), assuming the existence of a collision-resistant hiding hash function, which is a function that is hard to invert, for which it is hard to find 2 preimages mapping to the same image, and such that it is hard to find the second bit of the preimage given the image.

It is still unclear which possible world of complexity we live in and specific assumptions might be broken in the future. For example, the SIDH protocol was proven to be insecure in a breakthrough result [CD23], and there have been challenges to the hardness of LWE, and a recent attempt is [Che24], though this paper was found to contain a subtle bug. As a result, we want to construct OT protocols using even **weaker assumptions**. When considering classical functions, one of the most fundamental property is the notion of one-wayness, denoting the fact that the function is hard to invert. It is natural to ask if OT protocols can be obtained from one-way functions, but [CMS23] left this fundamental question unanswered:

*Can we construct OT using collision-resistant quantum **one-way functions**, i.e. without relying on the hiding bit property?*

The following theorem we prove in this thesis answers the question positively.

Theorem 1.1 (Informal)**.** *Assuming the existence of collision-resistant length-preserving quantum one-way functions, and the existence of any n-message ZK proof (or argument) of knowledge, we can obtain an $n+1$-message OT (in the CRS model), or an $n+2$-message OT (in the plain model).*

Corollary 1.2 (Informal)**.** *Assuming the existence of collision-resistant length-preserving quantum one-way functions, there exists a 2-message OT composably secure in the random oracle model.*

Moreover, we obtain the following result:

Theorem 1.3 (Informal)**.** *Assuming the existence of quantum one-way permutation, and the existence of any n-message ZK proof (or argument) of knowledge, we can obtain an $n+1$-message OT without additional assumptions.*

To obtain these protocols we rely on the quantum Goldreich-Levin theorem, which indicates that any length-preserving quantum one-way function can be converted into another quantum one-way function with a quantum hard-predicate. This quantum hard-predicate, informally speaking, is used to hide

one bit of information without assuming the hiding property of the function. The article [AC02] provides a proof of a quantum Goldreich-Levin theorem that applies to quantum one-way permutations, and we prove that the result can be generalized to any length-preserving quantum one-way function.

Organization. In Chapter 2, we introduce the notations and definitions we use in the thesis. In Chapter 3, we provide an informal overview of the construction of the protocol. In Chapter 4, we generalize the Quantum Goldreich-Levin Theorem to quantum one-way functions. In Chapter 5, we give the formal version of the protocol, as well as its correctness proof and security proof. And in Chapter 6, we conclude the contents of the thesis and leave some open questions for future work.

Chapter 2

Preliminaries

2.1 Basic notations

For any bit strings x and y, x_i is the i-th element of x, starting from 1. $x||y$ denotes the string in which y follows x. For 2 bit strings x, y that have the same length n, $x \oplus y$ denotes the string z of length n such that $z_i = x_i \oplus y_i$. The dot product between two strings is defined as $\langle y, x \rangle \coloneqq \oplus_{i=1}^{n} y_i x_i$. For a matrix M, we denote as $M^\dagger \coloneqq \overline{M^T}$ the conjugate transpose of the original matrix. We write $\Pr_{x \xleftarrow{\$} S}[\cdot]$ to denote the probability when x is sampled uniformly at random from the domain S; we also write $\Pr_x[\cdot]$ as a shortcut when the domain is clear in the context.

2.2 Quantum Computing

Here we introduce some basic quantum computing notions that we will use in the thesis. More information about quantum computing can be found in the book by Nielsen and Chuang [NC10].

2.2.1 Hilbert space and Dirac notation

To every quantum system we associate a Hilbert space $\mathscr{H} = \mathbb{C}^d$, and we use the so-called Dirac notation to represent vectors in this Hilbert space:

$$|\psi\rangle \coloneqq \begin{pmatrix} \psi_1 \\ \vdots \\ \psi_d \end{pmatrix}, \langle\psi| \coloneqq \begin{pmatrix} \bar{\psi_1} & \cdots & \bar{\psi_d} \end{pmatrix} \tag{2.1}$$

This way, dot products can be represented naturally:

$$\langle\phi|\psi\rangle = \begin{pmatrix}\bar{\phi_1} & \cdots & \bar{\phi_d}\end{pmatrix}\begin{pmatrix}\psi_1\\ \vdots\\ \psi_d\end{pmatrix}, |\psi\rangle\langle\phi| = \begin{pmatrix}\psi_1\\ \vdots\\ \psi_d\end{pmatrix}\begin{pmatrix}\bar{\phi_1} & \cdots & \bar{\phi_d}\end{pmatrix} \tag{2.2}$$

2.2.2 Qubit

A qubit is the most fundamental object in quantum computing, which can be seen as the quantum generalization of a bit.

A (pure) qubit can be represented as a vector $|\psi\rangle \in \mathbb{C}^2$ such that:

$$|\psi\rangle = \begin{pmatrix}\alpha\\ \beta\end{pmatrix} \tag{2.3}$$

where $|\alpha|^2 + |\beta|^2 = 1$, which means that $\langle\psi|\psi\rangle = 1$.

Moreover, we define the standard basis, also known as the computational basis $\{|0\rangle, |1\rangle\}$ as

$$|0\rangle \coloneqq \begin{pmatrix}1\\ 0\end{pmatrix}, |1\rangle = \begin{pmatrix}0\\ 1\end{pmatrix} \tag{2.4}$$

and the Hadamard basis $\{|+\rangle, |-\rangle\}$ as

$$|+\rangle \coloneqq \frac{1}{\sqrt{2}}\begin{pmatrix}1\\ 1\end{pmatrix}, |-\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\ -1\end{pmatrix} \tag{2.5}$$

Like in the classical setting, we need to handle cases with multiple qubits. Tensor product provides a way to describe multiple vector spaces by using one larger vector space.

Definition 2.1 (Tensor product)**.** *For matrix A that is $m \times n$ and B that is $p \times q$, the tensor product $A \otimes B$ is defined as the following $mp \times nq$ matrix:*

$$A \otimes B = \begin{pmatrix}A_{1,1}B & \dots & A_{1,n}B\\ \vdots & & \vdots\\ A_{m,1}B & \dots & A_{m,n}B\end{pmatrix} \tag{2.6}$$

For vector spaces $\mathscr{A}$ and $\mathscr{B}$, with basis $V = \{v_i\}$ and $W = \{w_j\}$ respectively, the tensor product $\mathscr{A} \otimes \mathscr{B}$ is defined as the vector space that is spanned by $\{v_i \otimes w_j\}$.

We define multiple qubits as follows:

A (pure) state of n qubits can be represented as a vector $|\psi\rangle$ in the vector space $(\mathbb{C}^2)^{\otimes n} = \mathbb{C}^d$ where $d = 2^n$ such that

$$|\psi\rangle = \begin{pmatrix} \psi_1 \\ \vdots \\ \psi_d \end{pmatrix} \tag{2.7}$$

and $\sum_{i=1}^{d} |\psi_i|^2 = 1$, that is, $\langle\psi|\psi\rangle = 1$.

For a state $|\phi\rangle$ of m qubits and a $|\psi\rangle$ of n qubits, we naturally define the composition of these 2 states as $|\phi\rangle \otimes |\psi\rangle$, which is a state of $m + n$ qubits, i.e. of dimension 2^{m+n}.

The standard basis $\{|x\rangle\}_{x\in\{0,1\}^n}$ is defined as

$$|x\rangle = \otimes_{i=1}^{n} |x_i\rangle = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{pmatrix} \tag{2.8}$$

where $\alpha_i = 1$ if x is the binary representation of $i - 1$, otherwise $\alpha_i = 0$. For example, in the 2-qubits space, the standard basis consists of the following states:

$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = |0\rangle \otimes |0\rangle, \qquad |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = |0\rangle \otimes |1\rangle \tag{2.9}$$

$$|10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = |1\rangle \otimes |0\rangle, \qquad |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = |1\rangle \otimes |1\rangle \tag{2.10}$$

2.2.3 Unitary operation

Just like with classical bits, we can perform operations on qubits. One can do two kinds of operations on a quantum state: unitaries and measurements. Unitary operations map quantum states to quantum states in a reversible way.

Definition 2.2 (Unitary operation)**.** *An operation U is unitary operation on a vector of dimension d iff U is a $d \times d$ matrix such that $U^\dagger U = UU^\dagger = I$.*

We also list some unitaries that we use in the thesis.

1. The Hadamard gate H is defined as

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{2.11}$$

This unitary transforms the computational basis into Hadamard basis, that is,

$$H|0\rangle = |+\rangle, H|1\rangle = |-\rangle, H|+\rangle = |0\rangle, H|-\rangle = |1\rangle. \tag{2.12}$$

2. The bit-flip gate X is defined as

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{2.13}$$

This unitary does a bit flip in the standard basis, and does a phase flip on state $|-\rangle$, that is,

$$X|0\rangle = |1\rangle, X|1\rangle = |0\rangle, X|+\rangle = |+\rangle, X|-\rangle = -|-\rangle. \tag{2.14}$$

3. The phase-flip gate Z is defined as

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.15}$$

This unitary does a bit flip in the Hadamard basis, and does a phase flip on state $|1\rangle$, that is,

$$Z|0\rangle = |0\rangle, Z|1\rangle = -|1\rangle, Z|+\rangle = |-\rangle, Z|-\rangle = |+\rangle. \tag{2.16}$$

4. The controlled phase-flip gate CZ is defined as

$$CZ = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{2.17}$$

This unitary operates on 2 qubits and only does a phase flip on the basis state $|11\rangle$, that is

$$CZ|00\rangle = |00\rangle, CZ|01\rangle = |01\rangle, CZ|10\rangle = |10\rangle, CZ|11\rangle = -|11\rangle. \tag{2.18}$$

5. For any function $f\colon \{0,1\}^n \to \{0,1\}^m$, there is a unitary $U_f\colon \mathbb{C}^{2^n} \otimes \mathbb{C}^{2^m} \to \mathbb{C}^{2^n} \otimes \mathbb{C}^{2^m}$ such that for every x in the domain of f and $b \in \{0,1\}^m$:

$$U_f|x\rangle|b\rangle = |x\rangle|b \oplus f(x)\rangle. \tag{2.19}$$

2.2.4 Density matrices

In practice, we may face a situation that a state is a probability distribution over different (pure) states. To describe this kind of situation, we use density matrices to generalize the notion of quantum state. We first give some mathematical definitions.

Definition 2.3 (Hermitian matrix)**.** *A linear operator $M : \mathbb{C}^d \to \mathbb{C}^d$ is Hermitian if $M^\dagger = M$.*

By the spectral theorem, a Hermitian matrix M can always be represented in the form $M = \sum_i \lambda_i |\psi_i\rangle \langle\psi_i|$ where λ_i's are real eigenvalues of M, and $|\psi_i\rangle$'s are orthonormal vectors, that is, $\langle\psi_i|\psi_i\rangle = 1$, and $\langle\psi_i|\psi_j\rangle = 0$ if $i \neq j$.

Definition 2.4 (Positive semidefinite matrix)**.** *A Hermitian matrix M is positive semidefinite, denoted by $M \geq 0$, if all its eigenvalues $\{\lambda_i\}_i$ are non-negative, i.e. $\lambda_i \geq 0$.*

Now we define the notion of density matrix:

Definition 2.5 (Density matrix)**.** *Consider a quantum system with state space $\mathbb{C}^d$. A density matrix ρ is a linear operator $\rho : \mathbb{C}^d \to \mathbb{C}^d$ such that*

1. *$\rho \geq 0$,*
2. *$\operatorname{Tr} \rho = 1$.*

The terms density matrix and density operator are often used interchangeably.

For a pure state $|\psi\rangle$, its density matrix is defined as $|\psi\rangle \langle\psi|$. Also, by the spectral theorem, a density matrix ρ can be written in the form $\rho = \sum_i p_i |\psi_i\rangle \langle\psi_i|$ where $p_i \geq 0$, $\sum_i p_i = 1$, and $|\psi_i\rangle$'s are orthonormal. This expression shows that a density matrix can be seen as an ensemble of pure states, associated with some probability distribution.

We naturally define the result of performing a unitary U on the density matrix ρ to be $U\rho U^\dagger$.

2.2.5 Measurement

A quantum state cannot be read directly. Instead, to retrieve information from it, we need to perform a measurement. A general definition of measurement is the positive operator-valued measurement (POVM). For convenience, we use the Kraus operator representation of POVM here.

Definition 2.6 (Kraus operator representation of POVM)**.** *A measurement on $\mathbb{C}^d$ is a set of linear operators $N_x : \mathbb{C}^d \to \mathbb{C}^d$ such that $\sum_x N_x^\dagger N_x = I$*

For a density operator ρ, if we measure it with a measurement $\{N_x\}$, the probability p_x that we get the outcome x is given by

$$p_x = \mathrm{Tr}[N_x \rho N_x^\dagger] \tag{2.20}$$

and the post-measurement state (conditioned on the outcome x) is

$$\rho_{|x} = \frac{N_x \rho N_x^\dagger}{\mathrm{Tr}[N_x \rho N_x^\dagger]} \tag{2.21}$$

Instead of measuring the whole state, we can also perform measurement on a subsystem.

For a density operator ρ_{AB} on the Hilbert space $\mathscr{A} \otimes \mathscr{B}$, if we measure it with a measurement $\{N_x\}$ on $\mathscr{A}$, the probability p_x that we get the outcome x is given by

$$p_x = \mathrm{Tr}[(N_x \otimes I)\rho_{AB}(N_x^\dagger \otimes I)] \tag{2.22}$$

and the post-measurement state (conditioned on the outcome x) is

$$\rho_{AB|x} = \frac{(N_x \otimes I)\rho_{AB}(N_x^\dagger \otimes I)}{\mathrm{Tr}[(N_x \otimes I)\rho_{AB}(N_x^\dagger \otimes I)]} \tag{2.23}$$

Especially, if $\{|\psi_x\rangle\}_x$ is an orthonormal basis, and $\{N_x\}_x = \{|\psi_x\rangle \langle\psi_x|\}_x$, then we say the system is measured in the corresponding basis.

The above equations show that two states are perfectly indistinguishable if they share the same density matrix. One corollary is that, for pure states, the global phase cannot be observed: since for pure states $|\psi\rangle$ and $|\phi\rangle = e^{i\theta}|\psi\rangle$, we have

$$\rho_\psi = |\psi\rangle \langle\psi| = e^{i\theta}|\psi\rangle \langle\psi| e^{-i\theta} = |\phi\rangle \langle\phi| = \rho_\phi \tag{2.24}$$

which means that these 2 states share the same density matrix.

2.2.6 Trace distance

A natural question is the distinguishability between 2 states. The notion of trace distance provides a way to describe such distinguishability.

We first introduce some linear algebra notions. For any Hermitian matrix A, the trace norm of A is defined as $\|A\|_1 \coloneqq \mathrm{Tr}(\sqrt{A^\dagger A}) = \sum_i |\lambda_i|$ where λ_i's are the eigenvalues of A.

With the fact that if ρ and σ are Hermitian, then $\rho - \sigma$ is also Hermitian, we define the trace distance as follows:

Definition 2.7. *For quantum states ρ, σ, the trace distance is defined as*

$$TD(\rho, \sigma) = \frac{1}{2}||\rho - \sigma||_1. \tag{2.25}$$

Note that the following equality holds, which connects the trace distance and the distinguishability between 2 states:

Theorem 2.8 ([NC10])**.**

$$TD(\rho, \sigma) = \max_N \frac{1}{2} \sum_x |\,\mathrm{Tr}[N_x \rho N_x^\dagger] - \mathrm{Tr}[N_x \sigma N_x^\dagger]| \tag{2.26}$$

where the maximum is taken over all possible POVMs.

2.2.7 Quantum circuit and Quantum algorithm

A quantum algorithm is an algorithm that runs on a model of quantum computation, and the most commonly used model is the quantum circuit model. A quantum circuit is a model for quantum computation equipped with a specific (universal) set of quantum gates, in which computation is a sequence of initializations of qubits, quantum gates, and measurements. Quantum gates are unitaries operating on a small number of qubits. A quantum circuit can perform a unitary by using a combination of quantum gates. A set of gates is said to be universal for quantum computation if any unitary operation can be approximated to arbitrary accuracy by a quantum circuit involving only those gates. For a quantum circuit which has input size n, and the total number of operations is upper bounded by $p(n)$, we say that the circuit runs in $p(n)$ time, and has size $p(n)$. If p is a polynomial, we say that the circuit runs in polynomial time, and has polynomial size.

2.3 Cryptography

Here we introduce some basic cryptography notions that we will use in the thesis.

2.3.1 Negligible Functions

An important concept in cryptography is negligible functions. Intuitively, the notion of negligible denotes the fact that the quantity (often a probability or a difference in probability) is so small that it cannot be told apart from 0. We formally define the notion as follows:

Definition 2.9 (Negligible functions)**.** *A function f is negligible if for every polynomial p, we have* $\lim_{n\to\infty} p(n)f(n) = 0$

So, if there is an algorithm that succeeds with negligible function of probability, then the success probability is close to 0 and repeating the algorithm for a polynomial number of times cannot significantly improve the success probability.

2.3.2 Quantum Stand-alone Security Model

[HSS11] defined a quantum stand-alone security model that we use to prove the security of our protocol. We quickly summarize it here.

Quantum Machine Model.

We use the notion of quantum interactive machine to formalize the parties that are involved in the protocol and security proof.

A quantum interactive machine (QIM) $\mathbf{A}$ is an ensemble of interactive quantum circuits $\{A_n\}_{n\in\mathbb{N}}$ working on 3 registers: an input register for input and workspace, an output register for output, and a networking register for the communication with other machines. We say that a machine $\mathbf{A}$ is QPT if there is some polynomial p such that for every n, A_n runs in $p(n)$ time, i.e. if the number of gates of each circuit is bounded by $p(n)$.

A (two-party) protocol $\Pi = (\mathbf{A}, \mathbf{B})$ is a couple of QIMs. By $\mathbf{A} \leftrightsquigarrow \mathbf{B}$ we denote the sequence of quantum maps (indexed by $n \in \mathbb{N}$) representing the interaction between A_n and B_n. More specifically, each map in this sequence takes the inputs to provide to A_n and B_n, forwards these inputs to A_n and B_n, lets them interact until they stop, and outputs the final outputs of A_n and B_n after the interaction. A protocol is said to be *poly-time* if all the parties run in polynomial time. A functionality is a QIM interacting with all parties: for two QIMs $\mathbf{A}$ and $\mathbf{B}$, we similarly denote as $\mathbf{A} \overset{\mathcal{F}}{\leftrightsquigarrow} \mathbf{B}$ the sequence of quantum maps where each map takes as input the inputs to provide to A_n and B_n, provides

these inputs to, respectively, A_n and B_n, lets A_n and B_n only interact with the functionality $\mathcal{F}$, and outputs the final outputs of A_n and B_n after the interaction.

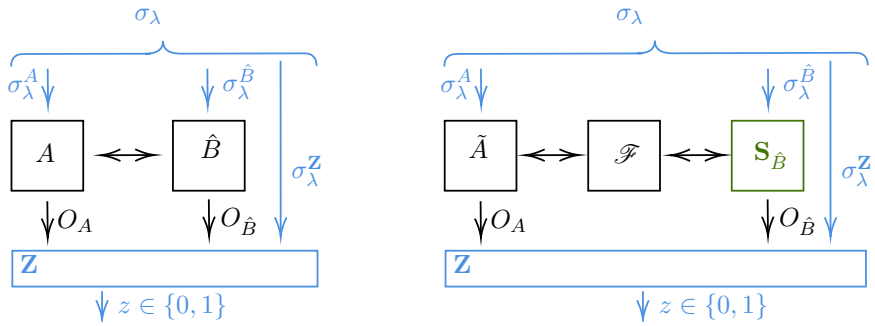
An adversary $\mathcal{A}$ is a QIM able to corrupt parties (in the two-party setting, corrupting a party can be simply thought of as $\mathcal{A}$ replacing the corrupted party). We only consider static adversaries here, which means that the adversary chooses which party to corrupt before the beginning of the protocol. We denote by $\hat{\mathbf{A}}$ the adversary that corrupts $\mathbf{A}$ and $\hat{\mathbf{B}}$ the adversary that corrupts $\mathbf{B}$. We define $\Pi \leftrightsquigarrow \mathcal{A}$ as the quantum map obtained when the protocol Π is run in the presence of the adversary $\mathcal{A}$.

Ideal-world Protocol and Secure Realization of a Functionality.

The security proof relies on the simulation paradigm involving real worlds and ideal worlds. A real world is a run of the protocol where some of the parties might be corrupted. The ideal world is an idealized protocol where the honest party is replaced by an idealized version and the corrupted party is replaced by a simulated version such that both parties are only allowed to interact with some idealized functionality. Take OT as the example, the idealized functionality of OT would basically accept b from Alice and two messages m_0 and m_1 from Bob, and after that outputs m_b to Alice. The functionality characterizes the security properties of the protocol by playing the role of a trusted third party that keeps perfect security. Still take OT as the example, Alice only receives one message m_b from the functionality so Alice cannot learn the information of m_{1-b}; Bob receives nothing so it is impossible for Bob to learn any information of b.

Informally speaking, if the real world and the ideal world are indistinguishable, then the protocol is secure since every attack made to the real world would also apply to the ideal world, which is secure by the definition of the ideal functionality. To achieve this, we use a QIM $\mathbf{Z}$ called a *distinguisher*, that receives the outputs of the parties, as well as some possible auxiliary qubits, and outputs one bit based on whether it thinks it is interacting with the real world or the ideal world, to distinguish these two worlds.

For the ideal world, in order to simulate the real world, we replace any honest party $\mathbf{A}$ by an idealized party $\tilde{\mathbf{A}}$ that honestly interacts with $\mathcal{F}$, where it honestly forwards the inputs/outputs to the functionality. We write $\tilde{\Pi} \coloneqq (\tilde{A}, \tilde{B})$ to denote this dummy protocol. For the corrupted parties, based on $\mathcal{A}$, we try



(a) Real world. (b) Ideal world.

Figure 2.1: Real world and ideal world executions when Bob is malicious.

to construct an adversary $\mathbf{S}_\mathcal{A}$ that corrupts the same party as $\mathcal{A}$. We call such adversaries *simulators* and their goal is to simulate the output of $\mathcal{A}$.

Now we formalize the notion of ideal and real worlds.

Definition 2.10. *Let $\Pi = (\mathbf{A}, \mathbf{B})$ be a two-party protocol, $\mathcal{A}$ be a static adversary as defined above, $\mathbf{S}_\mathcal{A}$ be a simulator, S_A and S_B denote the input registers of $\mathbf{A}$ and $\mathbf{B}$ respectively, $\sigma = \{\sigma_n \in S_A(n) \otimes S_B(n) \otimes \mathcal{W}(n)\}_{n\in\mathbb{N}}$ be a sequence of quantum states and $\mathbf{Z}$ be a QIM called* environment *outputting a single classical bit. We denote by $\mathsf{REAL}^\sigma_{\Pi,\mathcal{A},\mathbf{Z}} \coloneqq \mathbf{Z}((\Pi \leftrightsquigarrow \mathcal{A}) \otimes I)\sigma$ the (sequence of) binary random variables outputted by the environment $\mathbf{Z}$ at the end of an interaction where the adversary $\mathcal{A}$ corrupts some parties in Π. More specifically, σ consists of 3 parts: the inputs of $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{Z}$. Then, the $((\Pi \leftrightsquigarrow \mathcal{A}) \otimes I)\sigma$ denotes the output of the protocol Π, with corrupted party $\mathcal{A}$, runs with inputs of $\mathbf{A}$, $\mathbf{B}$, together with the unchanged input of $\mathbf{Z}$. So $\mathbf{Z}((\Pi \leftrightsquigarrow \mathcal{A}) \otimes I)\sigma$ denotes the final outcome of $\mathbf{Z}$ when receiving the output of the protocol and inputs of $\mathbf{Z}$. Similarly, we denote $\mathsf{IDEAL}^{\sigma,\mathcal{F}}_{\tilde{\Pi},\mathbf{S}_\mathcal{A},\mathbf{Z}} \coloneqq \mathbf{Z}((\tilde{\Pi} \overset{\mathcal{F}}{\leftrightsquigarrow} \mathbf{S}_\mathcal{A}) \otimes I)\sigma$ as the (sequence of) binary random variables output by the environment $\mathbf{Z}$ at the end of an interaction where the simulator can corrupt some dummy parties interacting with the ideal functionality $\mathcal{F}$.*

A protocol realizing a functionality informally means that the protocol is secure when considering the corresponding functionality. Before we formally define the realization of a functionality, we need to define formally the notion of indistinguishable quantum maps.

Definition 2.11 (Indistinguishable random variables)**.** *Two sequences of random variables $\mathbf{X} = \{X_n\}_{n\in\mathbb{N}}$ and $\mathbf{Y} = \{Y_n\}_{n\in\mathbb{N}}$ are said to be ε-*indistinguishable*, denoted $\mathbf{X} \approx_\varepsilon \mathbf{Y}$, if $|\Pr[X_n = 1] - \Pr[Y_n = 1]| \leq \varepsilon(n)$. In particular, if $\varepsilon = \mathsf{negl}(n)$, $\mathbf{X}$ and $\mathbf{Y}$ are said to be indistinguishable, denoted $\mathbf{X} \approx \mathbf{Y}$.*

Definition 2.12 (Computationally indistinguishable quantum maps)**.** *Two sequences of quantum maps* $\mathbf{X} = \{X_n\}_{n\in\mathbb{N}}$ *and* $\mathbf{Y} = \{Y_n\}_{n\in\mathbb{N}}$ *are said to be* computationally indistinguishable*, denoted* $\mathbf{X} \approx_c \mathbf{Y}$*, if for any poly-time* $\mathbf{Z} = \{\mathbf{Z}_n\}_{n\in\mathbb{N}}$ *and any sequence of bipartite advices* $\sigma = \{\sigma_n\}_{n\in\mathbb{N}}$*,* $\mathbf{Z}(\mathbf{X}\otimes I)\sigma \approx \mathbf{Z}(\mathbf{Y}\otimes I)\sigma$*.*

Definition 2.13 (Statistically indistinguishable quantum maps)**.** *Two sequences of quantum maps* $\mathbf{X} = \{X_n\}_{n\in\mathbb{N}}$ *and* $\mathbf{Y} = \{Y_n\}_{n\in\mathbb{N}}$ *are said to be* statistically indistinguishable*, denoted* $\mathbf{X} \approx_s \mathbf{Y}$*, if for any unbounded* $\mathbf{Z} = \{\mathbf{Z}_n\}_{n\in\mathbb{N}}$ *and any sequence of bipartite advices* $\sigma = \{\sigma_n\}_{n\in\mathbb{N}}$*,* $\mathbf{Z}(\mathbf{X}\otimes I)\sigma \approx \mathbf{Z}(\mathbf{Y}\otimes I)\sigma$*.*

With the above definition, we can define the realization of a functionality as follows:

Definition 2.14 (Quantum stand-alone (C-QSA) realization of a functionality [HSS11])**.** *Let* $\mathcal{F}$ *be a poly-time two-party functionality and* Π *be a poly-time two-party protocol. We say that* Π computationally quantum-stand-alone *(C-QSA) realizes* $\mathcal{F}$ *if for any poly-time adversary* $\mathcal{A}$ *there is a poly-time (in the time taken by* $\mathcal{A}$*) simulator* $\mathbf{S}_\mathcal{A}$ *such that for any poly-time environment* $\mathbf{Z}$ *and family of states* $\sigma = \{\sigma_n\}_{n\in\mathbb{N}}$*,* $\mathsf{REAL}^{\sigma}_{\Pi,\mathcal{A},\mathbf{Z}} \approx \mathsf{IDEAL}^{\sigma,\mathcal{F}}_{\tilde{\Pi},\mathbf{S}_\mathcal{A},\mathbf{Z}}$*.*

Definition 2.15 (Quantum stand-alone (S-QSA) realization of a functionality [HSS11])**.** *Let* $\mathcal{F}$ *be a poly-time two-party functionality and* Π *be a poly-time two-party protocol. We say that* Π statistically quantum-stand-alone *(S-QSA) realizes* $\mathcal{F}$ *if for any unbounded adversary* $\mathcal{A}$ *there is a poly-time (in the time taken by* $\mathcal{A}$*) simulator* $\mathbf{S}_\mathcal{A}$ *such that for any unbounded environment* $\mathbf{Z}$ *and family of states* $\sigma = \{\sigma_n\}_{n\in\mathbb{N}}$*,* $\mathsf{REAL}^{\sigma}_{\Pi,\mathcal{A},\mathbf{Z}} \approx \mathsf{IDEAL}^{\sigma,\mathcal{F}}_{\tilde{\Pi},\mathbf{S}_\mathcal{A},\mathbf{Z}}$*.*

Protocols in Hybrid Models.

In practice, it is always handy to analyze modularized protocols, that is, some protocol realizing a functionality $\mathcal{F}$ assuming there exists some protocol realizing a (more primitive) functionality $\mathcal{G}$. We say that we are in $\mathcal{G}-$*hybrid* model if such assumption is made. We call such a protocol a $\mathcal{G}-$*hybrid* protocol, and denote it $\Pi^\mathcal{G}$. If no such an assumption is made, then we say that we are in the plain model.

Also, the following composition theorem shows that sub-protocols can be combined to realize a functionality:

Theorem 2.16. *Let* $\mathcal{F}$ *and* $\mathcal{G}$ *be poly-time functionalities. Let* $\Pi^\mathcal{G}$ *be a* $\mathcal{G}-$*hybrid protocol that C-QSA (resp. S-QSA) realizes* $\mathcal{F}$*, and* Γ *be a protocol*

that `C-QSA` *(resp.* `S-QSA`*) realizes* $\mathcal{G}$*, then* $\Pi^{\mathcal{G}/\Gamma}$*, denoting the protocol deriving from* $\Pi^{\mathcal{G}}$ *that uses* Γ *to replace* $\mathcal{G}$ *in* $\Pi^{\mathcal{G}}$*,* `C-QSA` *(resp.* `S-QSA`*) realizes* $\mathcal{F}$*.*

Ideal Functionalities.

Here we define the ideal functionalities we use in the thesis.

We start with the Oblivious Transfer functionality.

Definition 2.17 (Functionality for Oblivious Transfer)**.** *We define the ideal functionality* $\mathcal{F}_{OT}$ *for oblivious transfer as follows:*

- *it receives one (classical) message from Bob's interface.*
- *it receives one (classical) message from Alice's interface.*
- *if the message from Bob is a binary pair* (m_0, m_1) *and the message from Alice is a bit* b*, it sends* m_b *to Alice, otherwise it sends* $\perp$*.*

We define trivially the idealized parties interacting with $\mathcal{F}_{OT}$*, that is, the parties that honestly forward their input/output to the functionality.*

We then proceed to the zero-knowledge functionality. Informally speaking, in (classical) Zero-knowledge proofs, the prover can prove to the verifier that some given statement is true without revealing any information except the fact that the statement is true.

Definition 2.18 (Functionality for Zero-knowledge)**.** *We define the ideal functionality* $\mathcal{F}_{ZK}^{\mathcal{R}}$ *for zero-knowledge as follows, where* $\mathcal{R}$ *is a relation describing a given language* $\mathcal{L}$ *such that* $x \in \mathcal{L} \Leftrightarrow \exists w, x\mathcal{R}w$ *:*

- *it receives one message from the prover's interface.*
- *if the message from the prover is a binary pair* (x, w) *such that* $x\mathcal{R}w$*, then the verifier receives* x*, otherwise the verifier receives* $\perp$*.*

We define trivially the idealized parties that interacts with $\mathcal{F}_{ZK}^{\mathcal{R}}$*.*

In our protocol, the properties of the main protocol are strongly related to the properties of the ZK scheme. Depending on different assumptions, we obtain different ZK protocols, and thus get different results of the main protocol.

[HSS11] defines a protocol realizing $\mathcal{F}_{ZK}^{\mathcal{R}}$ in the plain model. A non-interactive protocol in the random-oracle model, which is the model where parties have access to an oracle that responds to every unique query with a random response chosen uniformly from its output domain, is shown in [Unr15], and [CMS23] shows that it realizes $\mathcal{F}_{ZK}^{\mathcal{R}}$ in the framework of quantum stand-alone model.

Sampling functions is another important part in our protocol. This procedure prevents trivial attacks of finding collision in a single function. Like the ZK part, there are multiple choices for the sampling procedure. One way to do this is to use a Common Reference String (CRS) assumption. CRS assumes that all parties can get access to the same string that is honestly sampled by some fixed procedure thus can save one message of communication compared to the method we use in the plain model.

Now we model CRS as the following ideal functionality:

Definition 2.19 (Functionality for CRS)**.** *We define the ideal functionality $\mathcal{F}_{CRS}^{\mathtt{Gen}}$ for CRS as follows, where $\mathtt{Gen}$ is a PPT sampling procedure,*

- *it samples $x \leftarrow \mathtt{Gen}(1^n)$ and outputs x to all parties.*

We define trivially the idealized parties that interacts with $\mathcal{F}_{CRS}^{\mathtt{Gen}}$.

2.3.3 Quantum One-way Function

Informally speaking, a quantum one-way function is a function that can be efficiently computed but that is hard to reverse. In this thesis, we only consider length-preserving functions. So we obtain the following formal definition:

Definition 2.20 (Quantum One-Way Function)**.** *A (sequence of) function $f_n \colon \{0,1\}^n \to \{0,1\}^n$ is a (sequence of) quantum One-Way Function if f_n can be computed by a circuit of size $n^{O(1)}$ and there is no quantum circuit C of size $n^{O(1)}$ such that $\Pr_a[f_n(C(f_n(a))) = f_n(a)] \geq 1/n^{O(1)}$.*

2.3.4 Collision-resistance

We also define the notion of collision-resistance.

Definition 2.21 (Collision-resistance)**.** *A family of functions $\{f_k \colon \{0,1\}^n \to \{0,1\}^n\}_{k \in \mathcal{K}}$ is said to be (computationally) collision-resistant if there exists a polynomial generation algorithm $k \leftarrow \mathtt{Gen}_f(1^n)$ such that for any $k \in \mathcal{K}$, f_k*

can be evaluated in polynomial time, and for any QPT *adversary* $\mathcal{A}$ *and advice* $\{\sigma_n\}_{n\in\mathbb{N}}$*:*

$$\Pr\left[\, x \neq x' \wedge f_k(x) = f_k(x') \mid k \leftarrow \mathtt{Gen}_f(1^n), (x, x') \leftarrow \mathcal{A}(k, \sigma_n) \,\right] \leq \mathsf{negl}(n) \tag{2.27}$$

Obviously, if a function f is injective, then $\{f\}$ is collision-resistant. However, when the functions in $\{f_k\}$ are not necessarily injective, then we need to sample the function f_k after the beginning of the protocol. The reason is, if f_k is chosen before the beginning of the protocol, then an adversary taking advice that contains a collision can easily find a collision. So, the procedure of sampling f_k is contained in the protocol, and there are multiple ways to sample f_k, depending on different settings:

- If we assume the existence of an injective function f_0, then we let $\{f_0\}$ be the family of collision-resistant function, then there is a trivial 0-message protocol that both parties output 0.
- If we assume we run the whole protocol in the CRS model, then there is a trivial 0-message protocol that both parties output the number generated by $\mathcal{F}_{CRS}^{\mathtt{Gen}}$.
- There is a 1-message protocol that Bob samples $x \leftarrow \mathtt{Gen}(1^n)$ and sends x to Alice, and Alice outputs x only if $x \in \mathcal{K}$.

In practice, we can use a fixed well known function which is believed to be secure against collision attacks. This plays in a sense the role of a CRS.

In the previous paragraph, we see that there are different ways to sample f_k given different assumptions. In order to avoid the security proof relying on specific settings, we define an idealized functionality of sampling f_k abstracting the procedure of sampling f_k:

Definition 2.22 (Ideal functionality $\mathcal{F}_F^{\mathtt{Gen}}$)**.** *Let* $\{f_k \colon \{0,1\}^n \to \{0,1\}^n\}_{k\in\mathcal{K}}$ *be a family of collision-resistant functions generated by* Gen*. we define the ideal functionality* $\mathcal{F}_F^{\mathtt{Gen}}$ *as follows.* $\mathcal{F}_F^{\mathtt{Gen}}$ *receives an input* c *from Bob's interface, if* $c = \top$*, the functionality samples* $k \leftarrow \mathtt{Gen}(1^n)$ *and sends* k *to both parties, and if* $c \in \mathcal{K}$*, it forwards* c *to Alice's interface. The ideal party of* $\mathbf{A}$ *just forwards the received* k*, while the ideal party of* $\mathbf{B}$ *sends* $c = \top$ *to the functionality and outputs the received* k*.*

It can be proven that the above functionality can be realized by the protocols described above (in their corresponding settings). More formally, we have the following lemmas:

Lemma 2.23 ($\mathcal{F}_F$ in the Injective Function Setting)**.** *In the Injective Function Setting (where the family of collision-resistant functions contains a single injective function f_0), the trivial* 0*-message protocol that both parties output* 0 *realizes $\mathcal{F}_F^{\mathtt{Gen}}$.*

Lemma 2.24 ($\mathcal{F}_F$ in the CRS model)**.** *In the CRS model ($\mathcal{F}_{CRS}$-hybrid model), the trivial* 0*-message protocol that both parties output the number generated by $\mathcal{F}_{CRS}^{\mathtt{Gen}}$ realizes $\mathcal{F}_F^{\mathtt{Gen}}$.*

Lemma 2.25 ($\mathcal{F}_F$ in the plain model)**.** *In the plain model, the* 1*-message protocol that Bob samples $x \leftarrow \mathtt{Gen}(1^n)$ and sends x to Alice, and Alice outputs x only if $x \in \mathcal{K}$ realizes $\mathcal{F}_F^{\mathtt{Gen}}$.*

Proof of Lemma 2.24 and Lemma 2.25 is shown in [CMS23]. We prove Lemma 2.23 as follows:

Proof. We prove in 3 cases that the real world equals the ideal world.

For the case that both parties are honest, then both parties output 0, and in the ideal world, the ideal parties also output 0, so the real world equals the ideal world.

For the case of malicious Alice, we let the simulator be malicious Alice receiving 0 from the functionality. In this setting the real world equals the ideal world.

For the case of malicious Bob, we let the simulator be malicious Bob sending $c = \top$ to the functionality and receiving 0 from the functionality. In this setting the real world equals the ideal world.

So in all 3 cases the real world and the ideal world are equal, which ends the proof. □

2.4 A Quantum Goldreich-Levin Theorem

The Goldreich-Levin Theorem in [GL89] shows how every one-way function can be modified to obtain a one-way function that has a hardcore-predicate that can be used to hide information. Generally speaking, this theorem transforms any length-preserving quantum one-way function to quantum one-way function

with a quantum hard-predicate and we can use the converted function to hide information.

A quantum Goldreich-Levin theorem is introduced in [AC02]. Generally speaking, this theorem transforms any quantum one-way permutation to quantum one-way permutation with a quantum hard-predicate and we can use the converted function to hide information.

We first define the new function that is constructed from the old one.

Definition 2.26. *For a function* $f : \{0,1\}^n \to \{0,1\}^n$, $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^n$ *is defined as* $F(y,x) = (f(y),x)$ *for all* $(y,x) \in \{0,1\}^n \times \{0,1\}^n$.

Then we define the quantum hard-predicate.

Definition 2.27 (Quantum hard-predicate)**.** *The quantum hard-predicate* $h(y,x)$ *is defined as* $h(y,x) = \langle y,x\rangle = \oplus_{i=1}^{n} y_i x_i$.

Then we have the quantum Goldreich-Levin theorem. This theorem mainly says that if f itself can be computed efficiently and the hard-predicate can be predicted efficiently with non-negligible probability, then f can be inverted efficiently with non-negligible probability. More formally,

Theorem 2.28 (Quantum Goldreich-Levin Theorem)**.** *For a permutation* $f : \{0,1\}^n \to \{0,1\}^n$, *if* f *can be computed with a quantum circuit of size* $o(T)$, *and there is a quantum circuit* G *that also has size* $o(T)$ *such that* $\Pr_{y,x}[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \varepsilon$, *then there exists a quantum circuit* C *of size* $O(T/\varepsilon)$ *such that* $\Pr_a[C(f(a)) = a] \geq \varepsilon/2$.

This theorem is not enough for our protocol since we use length-preserving quantum one-way functions instead of permutations. However, similar proof strategy can be used to prove a generalized version of this theorem. We prove the generalized version in Chapter 4.

Chapter 3

Overview of the Protocol

In this section, we provide a quick informal overview of the protocol. Informally, the OT functionality is connected to two parties: a sender Bob holds 2 messages m_0 and m_1, and a receiver Alice holds a bit b. Alice forwards b to the functionality and Bob forwards m_0 and m_1 to the functionality; then the functionality outputs m_b to Alice. Since no message is sent to Bob, we expect Bob to be unable to learn any information on b when Alice is honest, and Alice cannot learn both m_0 and m_1 if Bob is honest since only one of them is sent by the functionality.

3.1 An intuition: A naive approach

We start from the following observation. For a qubit $|\phi\rangle$, if it is in the computational basis, then performing a Z gate on it just adds a global phase on it. More specifically, we have $Z|0\rangle = |0\rangle$, which does not change the state at all, and $Z|1\rangle = -|1\rangle$, which only adds a -1 phase to it.When consider their density matrices, we have

$$|0\rangle\langle 0| = Z|0\rangle\langle 0|Z^\dagger \tag{3.1}$$

$$|1\rangle\langle 1| = (-1)|1\rangle\langle 1|(-1) = Z|1\rangle\langle 1|Z^\dagger \tag{3.2}$$

This means that it is physically impossible to distinguish whether a Z gate is performed on a qubit in the computational basis. On the other hand, whether a Z gate is performed on a qubit in the Hadamard basis can be detected by performing a measurement in the Hadamard basis.

Also, we can observe that it is physically impossible to distinguish a random qubit in the computational basis from a random qubit in the Hadamard basis:

let ρ be the density matrix of the qubit in the computational basis and σ the density matrix of the qubit in the Hadamard basis, then we have

$$\begin{aligned}
\rho =& \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1| && (3.3)\\
=& \frac{1}{2}(\frac{1}{2}(|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| + |1\rangle\langle 1|) && (3.4)\\
&+ \frac{1}{2}(|0\rangle\langle 0| - |0\rangle\langle 1| - |1\rangle\langle 0| + |1\rangle\langle 1|)) && (3.5)\\
=& \frac{1}{2}(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)(\langle 0| + \langle 1|)\frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)(\langle 0| - \langle 1|)\frac{1}{\sqrt{2}}) && (3.6)\\
=& \frac{1}{2}(|+\rangle\langle +| + |-\rangle\langle -|) && (3.7)\\
=& \sigma && (3.8)
\end{aligned}$$

This observation gives us the intuition of a first, naive protocol: Alice sends $|\psi^{(b)}\rangle$, randomly chosen from $|+\rangle$ and $|-\rangle$, and $|\psi^{(1-b)}\rangle$, randomly chosen from $|0\rangle$ and $|1\rangle$ to Bob; Bob then performs Z^{m_0} and Z^{m_1} on the received qubits and sends them back to Alice; then Alice measures in the Hadamard basis the b-th qubit, which is equal to $Z^{m_b}|\psi^{(b)}\rangle$ before the measurement, and can successfully obtain m_b by checking the measurement outcome and comparing it with $|\psi^{(b)}\rangle$.

In this naive approach, since $|\psi^{(b)}\rangle$ is randomly chosen from $|+\rangle$ and $|-\rangle$ and $|\psi^{(1-b)}\rangle$ is randomly chosen from $|0\rangle$ and $|1\rangle$, their density matrices are the same, which means that Bob cannot distinguish $|\psi^{(b)}\rangle$ and $|\psi^{(1-b)}\rangle$. This solves one part of the problem, that Bob should not learn b if Alice is honest. Similarly, since rotating a qubit in the computational basis leaves the state unchanged, a semi-honest Alice that actually sends one qubit in the computational basis cannot recover the value of the other message m_{1-b} from the measurements of Bob.

However, Alice can easily obtain both messages by sending both qubits in Hadamard basis.

3.2 A Protocol using ZK and the Quantum Goldreich-Levin Theorem

In order to avoid this cheating strategy, we want to ensure that at least one qubit sent by Alice is in the computational basis, that is, not in some superposition. However, it is physically impossible to check if a qubit is actually

in a computational basis state or not, and we cannot let Alice tell Bob directly what qubits she sent, as this will easily let Bob know b. Instead, we let Alice send some bigger states, containing some extra qubits that allow Bob to check whether there is one state in the computational basis without disturbing the state when it is honestly prepared.

First, Alice samples 4 classical strings, 3 of which are used as the classical description of the quantum states they will send (one is in superposition and one is not), showing which basis these quantum states are allowed to be in, and the last one of them is an invalid string, which is a dummy object that prevents Bob from trivially finding b based on the different number of descriptions of quantum states. More specifically, Alice samples 4 strings $d||v_d^{(c)}||x_d^{(c)}||u_d^{(c)} \in \{0,1\} \times \{0,1\}^{n-1} \times \{0,1\} \times \{0,1\}^{n-1}$ for $c, d \in \{0,1\}$ where 3 of these are valid strings and 1 is invalid. However, to prevent Alice from using the invalid string to construct a superposition, we need to add different marks on valid and invalid strings. However, these marks should not be known by Bob, otherwise Bob can still easily know b. The quantum hard-predicates play the role of the marks here. In practice, for some randomly chosen l, Alice samples the invalid string $v_{(1-l)}^{(1-b)}||x_{(1-l)}^{(1-b)}||u_{(1-l)}^{(1-b)}$ randomly, post-selecting on the fact that $h((1-l)||v_{(1-l)}^{(1-b)}, x_{(1-l)}^{(1-b)}||u_{(1-l)}^{(1-b)}) = 1$. Alice samples the valid strings similarly, post-selecting on the fact that the quantum hard-predicates equal 0.

Alice then obtains a function F with quantum hard-predicate by using the generalized quantum Goldreich-Levin Theorem. Instead of sending directly the strings to Bob, Alice uses F to compute the images of the strings and sends the images to Bob. The quantum hard-predicates prevent Bob from knowing which string is invalid, but the images of the strings are still sufficient for Bob to check by a unitary whether the quantum states are honestly prepared without disturbing the states. More specifically, Alice computes $F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$. For convenience, we write $w_d^{(c)} = v_d^{(c)}||x_d^{(c)}||u_d^{(c)}$. Then, Alice prepares two quantum states $|\psi^{(b)}\rangle \coloneqq |0\rangle |w_0^{(b)}\rangle + (-1)^{r^{(b)}} |1\rangle |w_1^{(b)}\rangle$ and $|\psi^{(1-b)}\rangle \coloneqq |l\rangle |w_l^{(1-b)}\rangle$ and sends the images and states to Bob.

However, hiding the information of the quantum hard-predicate raises another question: how can Bob know that Alice marked one of the strings invalid? A ZK proof can convince Bob that what Bob receives are the images of 4 strings in which one of the strings is invalid. So, after Alice sends the quantum states and $F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$, two parties run a ZK proof such that Bob is convinced that one of the strings is invalid. For now, Bob checks that the classical strings are honestly prepared, then Bob needs to

check whether the quantum states are honestly prepared. To do this, Bob runs a unitary check on the quantum states to see if they are honestly prepared. More specifically, Bob applies the unitary U such that $U\,|i||v||x||u||0\rangle = |i||v||x||u||h(i||v,x||u) \neq 1 \wedge \exists d, F(i||v,x||u) = F_d^{(c)}\rangle$ and measures the last qubit in the computational basis. This unitary checks whether a basis state corresponds to one of the valid strings. If the states pass the check, this means that Bob checks the quantum states are honestly prepared. Then, Bob can trace out the additional qubits, just leaving the first qubit of two states, while keeping the states in superposition and computational basis, respectively. To achieve this, Bob cannot simply ignore the additional qubits, since the entanglement can disturb the remaining state. Instead, Bob measures the second register of the states in Hadamard basis, then performs Z^{m_c} on the remaining qubit, and measures in the Hadamard basis. Bob then sends all the measurement outcomes to Alice and Alice can recover the desired message.

3.3 Security against some common attacks

In this part, we give an informal security check to see why the above protocol is secure against some common attacks. We use the security model described in Section 2.3.2 to formalize the security proof in Section 5.3.

We first consider attacks from Bob. Bob cannot learn b from strings $F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$ since the information of which string is invalid is hidden by the quantum hard-predicate. Bob cannot learn b from the ZK part since the ZK protocol does not leak information of the strings to Bob except that there is an invalid string. The quantum states sent to Bob also give no information about b since Bob does not know l and r, the density matrices of these two states are the same for Bob so Bob cannot learn anything from the states.

Then we consider attacks from Alice. If Alice wants to learn m_{1-b}, then $|\psi^{(1-b)}\rangle$ needs to be in superposition after the unitary check performed by Bob. This unitary check prevents Alice from sending a state that has nonzero amplitude in invalid basis, because if Alice does so, then either the measurement outcome is 0, which means that the test fails and the protocol abort; or the measurement outcome is 1, but the state will collapse to superposition that only contain valid strings due to the measurement. So, if $|\psi^{(1-b)}\rangle$ is still in superposition after the unitary check, then there are 4 strings that are valid. However, the ZK proof ensures that at most 3 of the preimages of $F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$ are valid. This means that it is possible for Alice to find a collision of F, which contradicts the assumption that F is collision-resistant.

The above arguments show that the protocol is secure against some common attacks. However, to rule out all possible attacks, we formalize the proof in Section 5.3.

Chapter 4

Generalized Quantum Goldreich-Levin Theorem

4.1 Preliminaries and Definition

The quantum Goldreich-Levin theorem in [AC02] transforms any quantum one-way permutation to a quantum one-way permutation with a quantum hard-predicate. However, this is not enough for our protocol, since we use quantum one-way functions, instead of quantum one-way permutations. So we prove a generalized version of quantum Goldreich-Levin theorem here, which states that any length-preserving quantum one-way function can be converted to a quantum one-way function with a quantum hard-predicate.

The converted function F and the quantum hard-predicate h are defined as in Section 2.4. Since we are using F, instead of f in the protocol, we need F to be collision-resistant. So we show that this transformation from f to F preserves collision-resistance. This means that if we assume the existence of a family of length-preserving collision-resistant quantum one-way functions, then we have a family of length-preserving collision-resistant quantum one-way functions with quantum hard-predicates. More formally,

Theorem 4.1. *If $\{f_k\}$ is collision-resistant, then $\{F_k\}$ is collision-resistant.*

Proof. Assume that $\{F_k\}$ is not collision-resistant, we show that $\{f_k\}$ is not collision-resistant.

For every $\mathtt{Gen}_f(1^n)$, let $\mathtt{Gen}_F(1^{2n})$ be the same algorithm as $\mathtt{Gen}_f(1^n)$, then, by the assumption, there is a QPT adversary $\mathcal{A}$ and (possible) advice $\{\sigma_n\}_{n\in\mathbb{N}}$

such that

$$\Pr\left[\, x||y \neq x'||y' \wedge F_k(x||y) = F_k(x'||y') \mid k \leftarrow \mathtt{Gen}_f(1^n), (x, y, x', y') \leftarrow \mathcal{A}(k, \sigma_n) \,\right] \tag{4.1}$$

is not negligible.

Note that $F_k(x||y) = F_k(x'||y')$ iff $f_k(x) = f_k(x') \wedge y = y'$ by the definition of F_k. So

$$x||y \neq x'||y' \wedge F_k(x||y) = F_k(x'||y') \tag{4.2}$$

$$\text{iff } x||y \neq x'||y' \wedge f_k(x) = f_k(x') \wedge y = y' \tag{4.3}$$

$$\Rightarrow x \neq x' \wedge f_k(x) = f_k(x') \tag{4.4}$$

So the following holds:

$$\Pr\left[\, x \neq x' \wedge f_k(x) = f_k(x') \mid k \leftarrow \mathtt{Gen}_f(1^n), (x, x') \leftarrow \mathcal{A}(k, \sigma_n) \,\right] \tag{4.5}$$

is not negligible.

This shows exactly that $\{f_k\}$ is not collision-resistant, which ends the proof. □

Then we formalize the generalized quantum Goldreich-Levin theorem as follows:

Theorem 4.2. *For a function $f : \{0,1\}^n \to \{0,1\}^n$, if f can be computed with a quantum circuit of size $o(T)$, and there is a quantum circuit G that also has size $o(T)$ such that $\Pr_{y,x}[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \varepsilon$, then there exists a quantum circuit C of size $O(T/\varepsilon)$ such that $\Pr_y[f(C(f(y))) = f(y)] \geq \varepsilon/2$.*

For an intuition, we consider the ideal case where there is a circuit G that can perfectly predict $h(y,x)$ given $(f(y),x)$ as input, and we try to invert f. Given $f(y)$, we define $x^{(i)}$ as the string such that $x^{(i)}_j = 1$ iff $i = j$. We run G on inputs $(f(y), x^{(i)})$ for $1 \leq i \leq n$. By the definition of $x^{(i)}$, $h(y, x^{(i)}) = y_i$, and we can invert f by combining the outputs of G. Unfortunately, this attack fails as soon as G can do mistakes, hence a clever approach is needed.

We will prove this theorem in the following sections.

4.2 The Generalized GL Problem

We define a similar black box problem as in [AC02]. To deal with the problem that the function is not necessarily injective, we add an extra equivalence relation in the initial setting, and replace the EQ query with the EQR query.

We first define a generalized version of the GL problem as follows.

Definition 4.3 (Generalized GL problem)**.** *For some equivalence relation $R \subseteq \{0,1\}^n \times \{0,1\}^n$, and for some $a \in \{0,1\}^n$, the goal is to find an $x \in \{0,1\}^n$ such that xRa, by using only 2 kinds of queries: the IP queries and the EQR queries.*

We define the IP queries as follows (the same definition as in [AC02]):

Definition 4.4 (Quantum inner product query)**.** *A quantum inner product (IP) query (with bias ε) is a unitary transformation U_{IP}, or its inverse $U_{IP}^\dagger$, acting on $n+m$ qubits, satisfying the following properties:*

1. *If $x \in \{0,1\}^n$ is uniformly randomly chosen and the last qubit of $U_{IP}|x\rangle|0^m\rangle$ is measured, outputting w, then $\Pr[w = a \cdot x] \geq \frac{1}{2} + \varepsilon$.*
2. *For any $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$, the first n qubits of $U_{IP}|x\rangle|y\rangle$ is $|x\rangle$.*

Then we define the EQR queries.

Definition 4.5 (Quantum equivalence relation query)**.** *A quantum equivalence relation (EQR) query is the unitary operation U_{EQR} such that for all $x \in \{0,1\}^n$ and $b \in \{0,1\}$,*

$$U_{EQR}|x\rangle|b\rangle = \begin{cases} |x\rangle|b\rangle & \text{if } \neg xRa \\ |x\rangle|1-b\rangle & \text{if } xRa \end{cases} \tag{4.6}$$

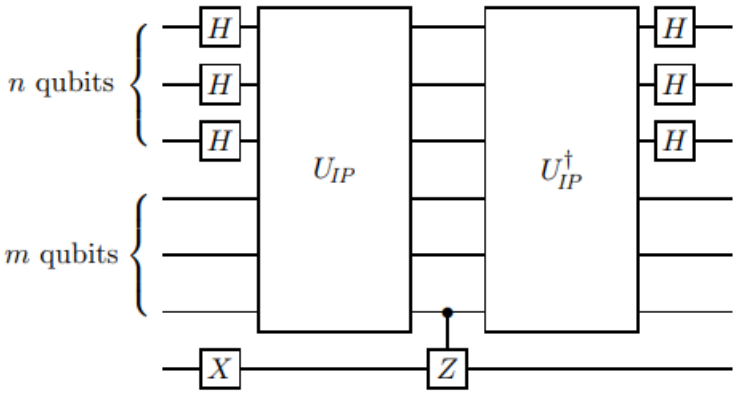


Figure 4.1: Quantum circuit C

[AC02] shows that by running the Quantum circuit C (as shown in Fig. 4.1) on input $|0^n, 0^m, 0\rangle$ and by measuring the first n qubits in computational basis, the probability that the outcome is a is at least $4\varepsilon^2$. More precisely, the following equation holds:

$$\langle a, 0^m, 1| \, C \, |0^n, 0^m, 0\rangle \geq 2\varepsilon. \tag{4.7}$$

So, if the process is repeated $O(1/\varepsilon^2)$ times and each result is checked using an EQR query, then we can find some x such that xRa with some constant probability. Moreover, we have the following theorem in [BHM+02]:

Theorem 4.6. *Let A be any quantum algorithm that uses no measurements, we denote the set of all possible measurement outcomes after performing A to be S. For any Boolean function $\chi : S \to \{0, 1\}$, we say $x \in S$ is a good solution of A if $\chi(x) = 1$. Then, for any such A and any χ, if the probability of finding a good solution of A, by running A once and measuring, is $a > 0$, then there is an algorithm that finds a good solution with a constant probability using $O(\frac{1}{\sqrt{a}})$ applications of A and $A^\dagger$.*

If we apply the above amplitude amplification to this process, then we can achieve a constant success probability using $O(1/\varepsilon)$ queries.

In conclusion, we obtain the following theorem:

Theorem 4.7. *For an IP query with bias ε, the generalized GL problem can be solved with constant probability using $O(1/\varepsilon)$ U_{IP}, $U_{IP}^\dagger$ and U_{EQR} queries.*

4.3 Proof of the Generalized Quantum Goldreich-Levin Theorem

We first prove the following lemma:

Lemma 4.8. *If* $\Pr_{y,x}[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \varepsilon$, *then*

$$\Pr_y[\Pr_x[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \frac{\varepsilon}{2}] \geq \varepsilon. \tag{4.8}$$

Proof. Assume that it is not the case that $\Pr_y[\Pr_x[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \frac{\varepsilon}{2}] \geq \varepsilon$. We call y "good" if $\Pr_x[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \frac{\varepsilon}{2}$. Then it holds that $\Pr_y[y \text{ is "good"}] < \varepsilon$. With the above inequality, we have

$$\Pr_{y,x}[G(F(y,x)) = h(y,x)] \tag{4.9}$$

$$= \Pr_y[y \text{ is "good"}] \Pr_{y,x}[G(F(y,x)) = h(y,x) | y \text{ is "good"}] \tag{4.10}$$

$$+ \Pr_y[y \text{ is not "good"}] \Pr_{y,x}[G(F(y,x)) = h(y,x) | y \text{ is not "good"}] \tag{4.11}$$

$$< \Pr_y[y \text{ is "good"}] \times 1 + \Pr_y[y \text{ is not "good"}] \times (\frac{1}{2} + \frac{\varepsilon}{2}) \tag{4.12}$$

$$< \varepsilon \times 1 + (1-\varepsilon)(\frac{1}{2} + \frac{\varepsilon}{2}) \tag{4.13}$$

$$= \frac{1-\varepsilon^2}{2} + \varepsilon \tag{4.14}$$

$$< \frac{1}{2} + \varepsilon \tag{4.15}$$

contradicting the assumption that $\Pr_{y,x}[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \varepsilon$. □

The above lemma says that if a circuit G can predict $h(y,x)$ with a relatively high probability, then there is a set of y's such that, when we fix a y from that set and only x is sampled randomly, G can predict $h(y,x)$ with a relatively high probability.

Now we proceed to prove Theorem 4.2 using Theorem 4.7.

Proof. This proof is similar to the proof of the quantum Goldreich-Levin Theorem (for quantum permutations) in [AC02]. The main idea is to reduce the problem of inverting f to the Generalized GL Problem.

«««< HEAD More specifically, in the setting of inverting f, the equivalence relation R is defined as xRy iff $f(x) = f(y)$. Then we consider the selection of a. We consider the case where we fix a to be some y that is "good", that is, $\Pr_x[G(F(a,x)) = h(a,x)] \geq \frac{1}{2} + \frac{\varepsilon}{2}$. Now the goal is to find some x such that xRa. We simulate U_{IP} and U_{EQR} queries using the given information of $f(a)$ and the circuits that compute f and predict $h(y,x)$ from $F(y,x)$. For the EQR query, since we are allowed to compute f, we can construct the unitary that maps $|x\rangle|z\rangle|b\rangle$ to $|x\rangle|z\rangle|1-b\rangle$ if $f(x) = z$, and maps $|x\rangle|z\rangle|b\rangle$ to $|x\rangle|z\rangle|b\rangle$ if $f(x) \neq z$. This unitary simulates the EQR query when we input $f(a)$ to the second register. For the IP query, since a is "good", performing G on $(f(a), x)$ simulates IP query (with bias $\varepsilon/2$). Then, by Theorem 4.7, there is a circuit C that has size $O(\frac{2}{\varepsilon} \times T) = O(T/\varepsilon)$ that inverts f with constant probability (WLOG, we can set the probability to be $\frac{1}{2}$). This circuit C is independent of the choice of a, which means that for every a that is "good", $\Pr[f(C(f(a))) = f(a)] \geq 1/2$. Also, by the previous lemma, along with the assumption from the theorem that $\Pr_{y,x}[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \varepsilon$, we conclude that $\Pr_y[\Pr_x[G(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \frac{\varepsilon}{2}] \geq \varepsilon$, which means

that when y is uniformly randomly sampled, then at least ε of all y's are "good", meaning that when we use C to invert f, $\Pr_y[f(C(f(y))) = f(y)] \geq \Pr_y[y \text{ is "good"}] \Pr_y[f(C(f(y))) = f(y) | y \text{ is "good"}] \geq \varepsilon \times \frac{1}{2} = \varepsilon/2$.

□

Chapter 5

Quantum OT Protocol

Of course, our analysis in Chapter 3 is very informal, and cannot really exclude other kinds of attacks. Hence, in this section, we formalize the notions that we described above.

5.1 The Protocol

First we define the OT protocol shown in Protocol 1. And we prove the following main theorem, where $w_d^{(c)} = v_d^{(c)}||x_d^{(c)}||u_d^{(c)} \in \{0,1\}^{n-1} \times \{0,1\} \times \{0,1\}^{n-1}$ and $h_d^{(c)} = h(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$:

Theorem 5.1. *Let $\{F_k\}_{k\in\mathcal{K}}$ be constructed from $\{f_k\}_{k\in\mathcal{K}}$, a family of collision-resistant length preserving quantum one-way functions, such that $F_k(y,x) = (f_k(y),x)$ for every k. Let $\Pi_F = (\mathbf{A}_F, \mathbf{B}_F)$ be a protocol* C-QSA *realizes $\mathcal{F}_{CRS}^{\mathtt{Gen}}$, and $\Pi_{zk} = (\mathbf{A}_{zk}, \mathbf{B}_{zk})$ be a protocol that* C-QSA *realizes $\mathcal{F}_{ZK}^{\mathcal{R}}$ where*

$$(F_0^{(0)}, F_1^{(0)}, F_0^{(1)}, F_1^{(1)})\mathcal{R}(w_0^{(0)}, w_1^{(0)}, w_0^{(1)}, w_1^{(1)}) \Leftrightarrow \forall c, d, F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)}) = F_d^{(c)} \tag{5.1}$$

$$\text{and } \exists c, d, h_d^{(c)} = 1. \tag{5.2}$$

Then Protocol 1, with F obtained by running Π_F first, C-QSA *realizes $\mathcal{F}_{OT}$.*

To check that a protocol realizes a functionality, we need to check a number of properties, for different (possible) corrupted parties: when everyone is honest, when Alice is malicious, and when Bob is malicious. We study these cases separately in the following sections.

Protocol 1: Protocol for 2-message chosen bit Oblivious Transfer

Alice$(b \in \{0,1\})$ **Bob**$((m_0, m_1) \in \{0,1\}^2)$

// Witness for $\mathcal{L}^{(b)} = \{0,1\}$:

$\forall d \in \{0,1\}, w_d^{(b)} \xleftarrow{\$} \{0,1\}^{2n-1}$ **such that** $h_d^{(b)} = 0$

For convenience, We decompose $w_d^{(c)}$ into $w_d^{(c)} = v_d^{(c)}||x_d^{(c)}||u_d^{(c)} \in \{0,1\}^{n-1} \times \{0,1\} \times \{0,1\}^{n-1}$ and $h_d^{(c)} = h(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$
By randomly sampling $w_d^{(c)}$ $O(n)$ times, Alice can get an appropriate $w_d^{(c)}$ with probability ≈ 1.

// Witness for $\mathcal{L}^{(1-b)} = \{l\}$:

$l \xleftarrow{\$} \{0,1\}$

$w_l^{(1-b)} \xleftarrow{\$} \{0,1\}^{2n-1}$ **such that** $h_l^{(1-b)} = 0$

$w_{1-l}^{(1-b)} \xleftarrow{\$} \{0,1\}^{2n-1}$ **such that** $h_{(1-l)}^{(1-b)} = 1$

// Compute the characterization

// of the languages:

$\forall (c,d) \in \{0,1\}^2, F_d^{(c)} \coloneqq F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$

F is a collision-resistant length-preserving quantum OWF with quantum hard predicate h distributed using some $\mathcal{F}_F$.

// Proof that at least one language

// contains a single element

$\pi \coloneqq$ (NI)ZK proof that:

$\exists (w_d^{(c)})_{c,d}, \forall c, d, F_d^{(c)} = F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)})$

and $\exists c, d$ such that $h_d^{(c)} = 1$.

If the ZK proof is interactive, then we actually run the ZK protocol (before sending the quantum state) instead of sending the proof (of course this adds additional rounds of communication).

// Define the quantum states:

$r^{(b)} \xleftarrow{\$} \{0,1\}$

$|\psi^{(b)}\rangle \coloneqq |0\rangle |w_0^{(b)}\rangle + (-1)^{r^{(b)}} |1\rangle |w_1^{(b)}\rangle$

$|\psi^{(1-b)}\rangle \coloneqq |l\rangle |w_l^{(1-b)}\rangle$

$\forall (c,d) : F_d^{(c)}, \pi, |\psi^{(0)}\rangle, |\psi^{(1)}\rangle$ →

// Check that one language has size ≤ 1:

Check (or run if interactive proof) π.

We can define a Boolean function $g(i||v||x||u)$ such that $g(i||v||x||u) = 1$ if $h(i||v,x||u) \neq 1 \wedge \exists d, F(i||v,x||u) = F_d^{(c)}$, and $g(i||v||x||u) = 0$ otherwise. Hence, we can define unitary U that maps $|i||v||x||u||0\rangle$ to $|i||v||x||u||1\rangle$ if $h(i||v,x||u) \neq 1 \wedge \exists d, F(i||v,x||u) = F_d^{(c)}$, and to $|i||v||x||u||0\rangle$ otherwise.

// Check that the state contains a superposition

// of (valid) elements of $\mathcal{L}^{(0)}$ and $\mathcal{L}^{(1)}$:

$\forall c$, apply on $|\psi^{(c)}\rangle |0\rangle$ the unitary U

measure the last (output) register

and check that the outcome is 1.

At that step, $|\psi^{(b)}\rangle = |0\rangle \pm |1\rangle$ and $|\psi^{(1-b)}\rangle = |l\rangle$, but Bob does not know b.

$\forall c$, measure the second register of $|\psi^{(c)}\rangle$

in the Hadamard basis (outcome $s^{(c)}$).

$\forall c$, apply Z^{m_c} on $|\psi^{(c)}\rangle$ and measure it

in the Hadamard basis (outcome $z^{(c)}$).

← $\forall c, s^{(c)}, z^{(c)}$

Compute $\alpha \coloneqq r^{(b)} \oplus \langle s^{(b)}, w_0^{(b)} \oplus w_1^{(b)} \rangle$

return $\alpha \oplus z^{(b)}$ // Should be m_b

5.2 Correctness Check

In this part, we check the correctness of the protocol, by analyzing the case when both parties are honest.

We can divide the protocol into 4 parts: the first part obtains F (we assume the existence of collision-resistant quantum OWFs), the second part samples $w_d^{(c)}$ and calculates $F_d^{(c)}$, the third part runs the ZK protocol, and the fourth part does the remaining calculation and communication of the protocol.

By the assumption of the completeness of the protocol generating F and the ZK proof, we can always replace these parts with the ideal functionalities and dummy parties.

For the second part, we need to sample valid strings $w_d^{(c)}$. By randomly sampling $w_d^{(c)}$, and checking whether it is a valid string, and repeating this procedure until a valid string is sampled or the procedure has been repeated for $O(n)$ times. This whole process takes polynomial time (intuitively, at every sample we have a probability close to $1/2$ of sampling a valid element), and the success probability of obtaining a valid string is $1 - 1/2^{O(n)}$. For proof, see Theorem A.1.

Then we consider the last part. Since the strings and states are generated as in the second part, which means that $\forall c, d, F(d||v_d^{(c)}, x_d^{(c)}||u_d^{(c)}) = F_d^{(c)}$ and $\exists c, d, h_d^{(c)} = 1$, so the ZK check succeeds. Also, the unitary maps $|\psi^{(c)}\rangle |0\rangle$ to $|\psi^{(c)}\rangle |1\rangle$ for all c. Measuring the last register outputs 1 and leaves the other qubits unchanged. More specifically, the unitary maps $|i\rangle |w\rangle |0\rangle$ to $|i\rangle |w\rangle |1\rangle$ only if $h(i||v, x||u) \neq 1 \wedge \exists d, F(i||v, x||u) = F_d^{(c)}$ where $w = v||x||u$, which is true for all terms in $|\psi^{(b)}\rangle$ and $|\psi^{(1-b)}\rangle$ by construction, so we have:

$$|\psi^{(b)}\rangle |0\rangle = |0\rangle |w_0^{(b)}\rangle |0\rangle + (-1)^{r^{(b)}} |1\rangle |w_1^{(b)}\rangle |0\rangle \tag{5.3}$$

$$\mapsto |0\rangle |w_0^{(b)}\rangle |1\rangle + (-1)^{r^{(b)}} |1\rangle |w_1^{(b)}\rangle |1\rangle = |\psi^{(b)}\rangle |1\rangle \tag{5.4}$$

$$|\psi^{(1-b)}\rangle |0\rangle = |l\rangle |w_l^{(1-b)}\rangle |0\rangle \tag{5.5}$$

$$\mapsto |l\rangle |w_l^{(1-b)}\rangle |1\rangle = |\psi^{(1-b)}\rangle |1\rangle \tag{5.6}$$

Measuring the last qubit outputs 1, and factors out the last qubit without changing the remaining state.

Then we measure the second register in Hadamard basis, which is equivalent to applying Hadamards to the second registers and measuring them in the computational basis.

For $|\psi^{(b)}\rangle$, the state after the Hadamards becomes (omitting the global phase)

$$(I \otimes H^{2n-1}) |\psi^{(b)}\rangle \tag{5.7}$$

$$= |0\rangle H^{2n-1} |w_0^{(b)}\rangle + (-1)^{r^{(b)}} |1\rangle H^{2n-1} |w_1^{(b)}\rangle \tag{5.8}$$

$$= |0\rangle \sum_{s^{(b)} \in \{0,1\}^{2n-1}} (-1)^{\langle s^{(b)}, w_0^{(b)} \rangle} |s^{(b)}\rangle + (-1)^{r^{(b)}} |1\rangle \sum_{s^{(b)} \in \{0,1\}^{2n-1}} (-1)^{\langle s^{(b)}, w_1^{(b)} \rangle} |s^{(b)}\rangle \tag{5.9}$$

$$= \sum_{s^{(b)} \in \{0,1\}^{2n-1}} (|0\rangle + (-1)^{r^{(b)} \oplus \langle s^{(b)}, w_0^{(b)} \oplus w_1^{(b)} \rangle} |1\rangle) |s^{(b)}\rangle \tag{5.10}$$

If we measure the second register and get the outcome $s^{(b)}$, then the first register collapses to $|0\rangle + (-1)^{\alpha}|1\rangle$ where $\alpha = r^{(b)} \oplus \langle s^{(b)}, w_0^{(b)} \oplus w_1^{(b)} \rangle$.

For $|\psi^{(1-b)}\rangle$, since it is in computational basis, the first qubit is not entangled with the rest of the qubits, the measurement does not change the state of the first qubit. So the state collapses to $|l\rangle$.

Then, the Z^{m_c} rotation changes these 2 states to (omitting the global phase):

$$Z^{m_b}(|0\rangle + (-1)^{\alpha}|1\rangle) = |0\rangle + (-1)^{\alpha \oplus m_b}|1\rangle \tag{5.11}$$

$$Z^{m_{1-b}} |l\rangle = |l\rangle \tag{5.12}$$

Then Bob measures in the Hadamard basis, let $z^{(b)}$ be the measurement outcome of the remaining state of $|\psi^{(b)}\rangle$, then $z^{(b)} = \alpha \oplus m_b$, and $m_b = \alpha \oplus z^{(b)}$ is exactly what Alice outputs in the end. So the protocol is proven to be correct.

5.3 Security check

In this part, we check the security of the protocol, by analysing the cases when one party is malicious. We use a hybrid game to define a sequence of worlds, where in the first world (the real world) the parties run the actual protocol, and in the last world (the ideal world) the parties run a simulated version with ideal functionality, and by proving that every 2 adjacent worlds are indistinguishable, by transitivity, we obtain the fact that the real world and the ideal world are indistinguishable.

5.3.1 Malicious Bob

We define World_0 to World_6 as follows:

- World_0 (Fig. 5.1) is the real world.
- World_1 (Fig. 5.2) is World_0 except that the protocol that generates F is replaced by the simulated version.
- World_2 (Fig. 5.3) is World_1 except that the ZK protocol is replaced by the simulated version.
- World_3 (Fig. 5.4) is World_2 except that we send $\forall c, d, F_d^{(c)}$ without running the ZK.
- World_4 (Fig. 5.5) is World_3 except that we sample $w_{1-l}^{(1-b)}$ such that $h_{1-l}^{(1-b)} = 0$.
- World_5 (Fig. 5.6) is World_4 except that we define $|\psi^{(1-b)}\rangle = |0\rangle\,|w_0^{(1-b)}\rangle + (-1)^{r^{(1-b)}}|1\rangle\,|w_1^{(1-b)}\rangle$ with some randomly sampled $r^{(1-b)}$.
- World_6 (Fig. 5.7) is a reorder of the operations of World_5, and is the ideal world. More precisely, we can see that all the interaction and calculation in World_5, except the last operation that outputs $\alpha \oplus z^{(b)}$, are independent of b, so we can just reorder these parts to Bob's part to construct a simulator that forwards 2 messages $m^{(0)}$ and $m^{(1)}$. For the remaining part of Alice, we separate it into 2 parts: the idealized functionality receives $m^{(0)}$ and $m^{(1)}$ from Bob and b from Alice, and outputs $m^{(b)}$ to Alice, and the corresponding idealized party of Alice that forwards b to the functionality, and outputs $m^{(b)}$ received from the functionality.

Then we show that $\mathsf{World}_0 \approx \mathsf{World}_6$.

1. $\mathsf{World}_0 \approx \mathsf{World}_1$: Any distinguisher that distinguishes World_0 and World_1 can be converted to a distinguisher that distinguishes the protocol of obtaining F and the idealized functionality $\mathcal{F}_F^{Gen}$. We assumed that $\mathcal{F}_F^{Gen}$ is realized by the protocol of obtaining F, so such distinguisher does not exist. So $\mathsf{World}_0 \approx \mathsf{World}_1$.
2. $\mathsf{World}_1 \approx \mathsf{World}_2$: Any distinguisher that distinguishes World_1 and World_2 can be converted to a distinguisher that distinguishes the ZK protocol

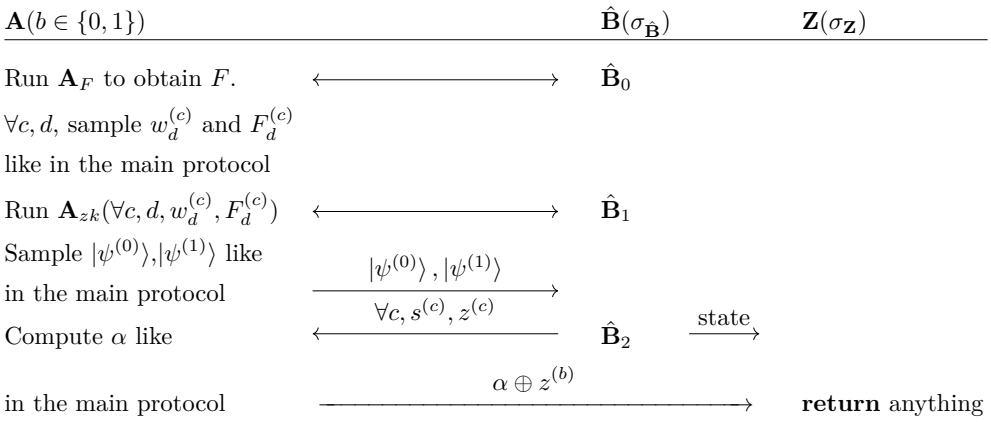


Figure 5.1: World_0

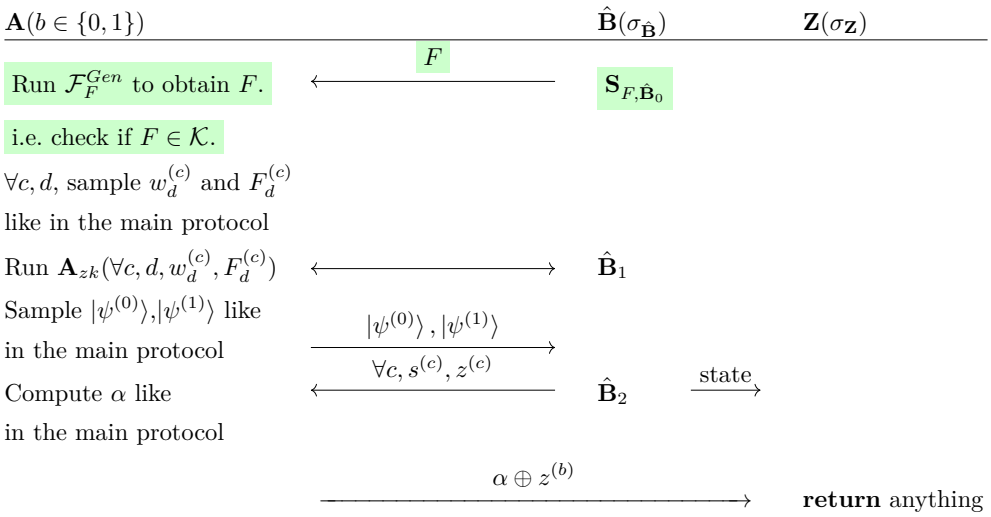


Figure 5.2: World_1

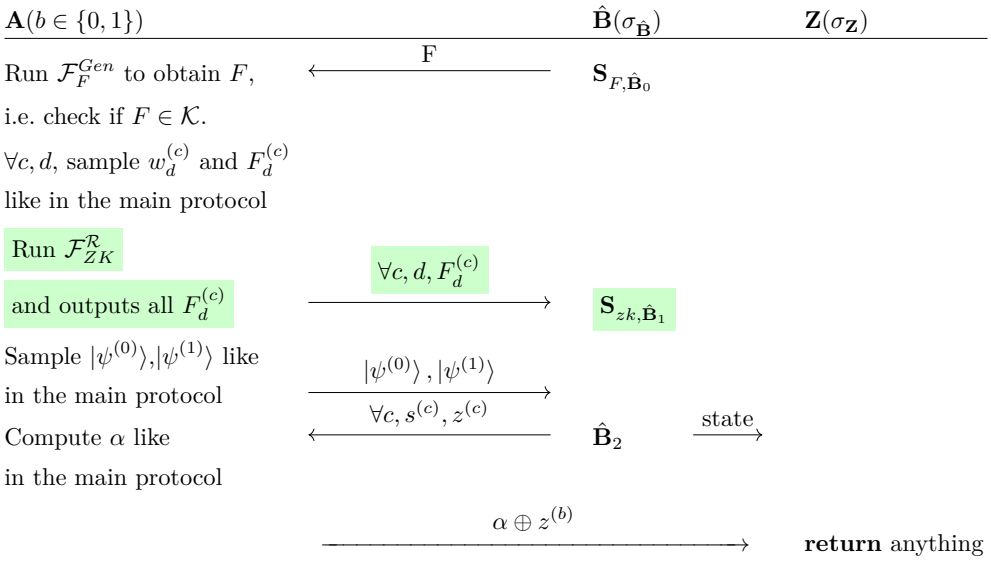


Figure 5.3: World_2

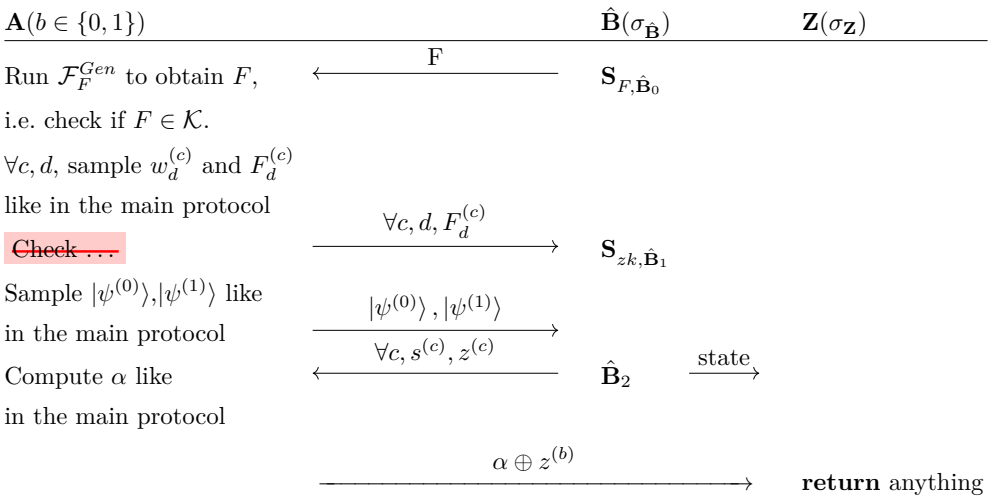


Figure 5.4: World_3

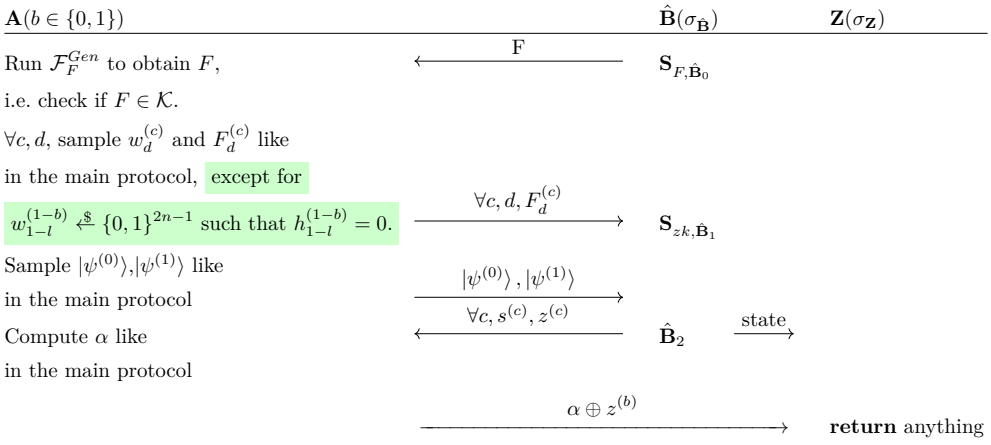


Figure 5.5: World_4

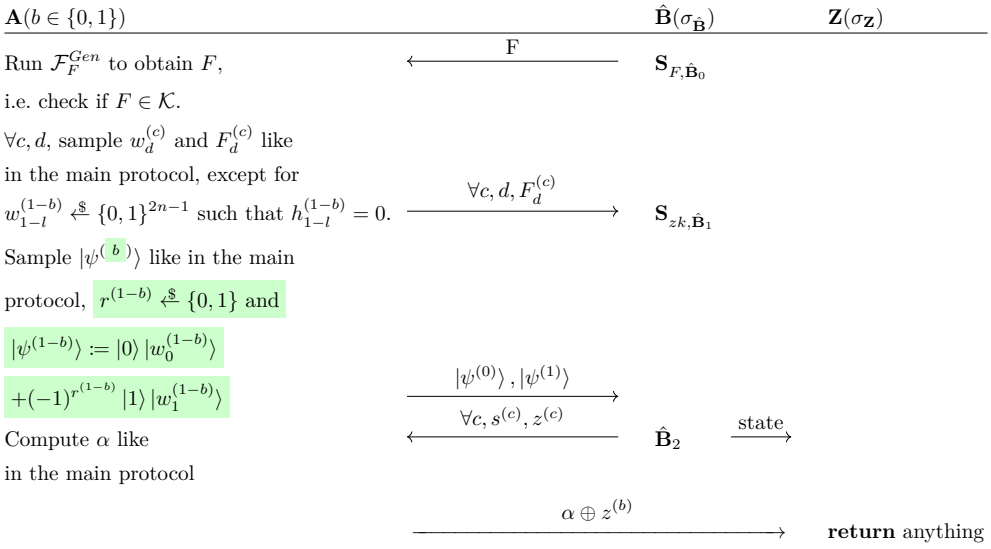


Figure 5.6: World_5

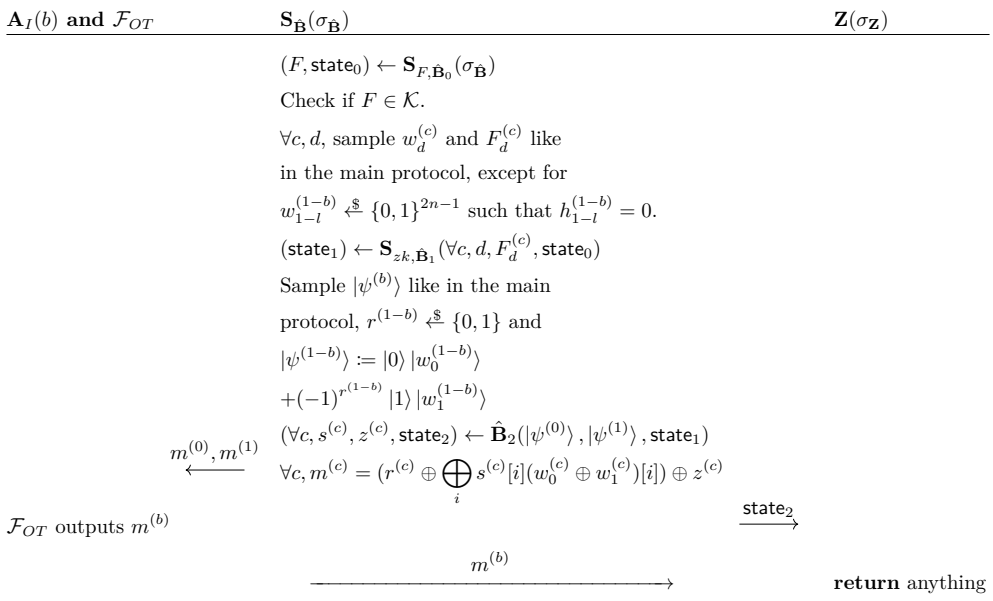


Figure 5.7: World_6

and the idealized functionality $\mathcal{F}_{ZK}^{\mathcal{R}}$. We assumed that $\mathcal{F}_{ZK}^{\mathcal{R}}$ is realized by a ZK protocol, so such distinguisher does not exist. So $\mathsf{World}_1 \approx \mathsf{World}_2$.

3. $\mathsf{World}_2 = \mathsf{World}_3$: By construction, the ideal Alice always samples valid strings. So the ZK proof always passes and as a result, it is always the case that $\forall c, d, F_d^{(c)}$ is forwarded.

4. $\mathsf{World}_3 \approx \mathsf{World}_4$: Suppose that $\mathsf{World}_3 \not\approx \mathsf{World}_4$, then there is a quantum polynomial-time distinguisher that can distinguish these 2 worlds with non-negligible probability. Then we can define C as the distinguisher combined with the World_3 except that $w_{1-l}^{(1-b)}$ is sampled externally and only provides $F_{1-l}^{(1-b)}$ to C, which has size $O(n^k)$ for some k, then C can distinguish the validity of the preimage of the given string with non-negligible probability, that is (WLOG), for some c,

$$\Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 1 | h(x,y) = 1] \tag{5.13}$$

$$- \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 1 | h(x,y) = 0] \tag{5.14}$$

$$\geq \frac{1}{n^c} \tag{5.15}$$

Let

$$a = \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 1 | h(x,y) = 1] \tag{5.16}$$

$$b = \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 1 | h(x,y) = 0] \tag{5.17}$$

Given some $F(x,y)$, we use C to predict $h(x,y)$. Combined with Theorem A.2, this gives us a success probability

$$\begin{aligned}
\Pr_{\text{success}} = & \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 1 \wedge h(x,y) = 1] && (5.18)\\
& + \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 0 \wedge h(x,y) = 0] && (5.19)\\
= & \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 1 | h(x,y) = 1] \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[h(x,y) = 1] && (5.20)\\
& + \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[C(F(x,y)) = 0 | h(x,y) = 0] \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[h(x,y) = 0] && (5.21)\\
= & a \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[h(x,y) = 1] + (1-b) \Pr_{x,y \xleftarrow{\$} \{0,1\}^n}[h(x,y) = 0] && (5.22)\\
= & a(\frac{1}{2} + \mathsf{negl}(n)) + (1-b)(\frac{1}{2} + \mathsf{negl}(n)) && (5.23)\\
\geq & \frac{1}{2}(1 + \frac{1}{n^c}) + \mathsf{negl}(n) && (5.24)
\end{aligned}$$

which can be lower bounded by $\frac{1}{2} + \frac{1}{n^{c'}}$ for some c'.

To conclude the above, assume that $\mathsf{World}_3 \not\approx \mathsf{World}_4$, there exists a quantum circuit C of size $O(n^k)$ such that $\Pr_{y,x}[C(F(y,x)) = h(y,x)] \geq \frac{1}{2} + \frac{1}{n^{c'}}$ for some k and c'.

Then, by Theorem 2.28, there is a polynomial-size quantum circuit that inverts f with non-negligible probability, thus contradicting the assumption that f is a quantum one-way function.

So $\mathsf{World}_3 \approx \mathsf{World}_4$.

5. $\mathsf{World}_4 = \mathsf{World}_5$: To distinguish World_4 and World_5 equals to distinguish 2 quantum states $|x_4\rangle$ and $|x_5\rangle$ corresponding to $|\psi^{(1-b)}\rangle$ in World_4 and World_5. Let $|0\rangle |w_0^{(1-b)}\rangle = |x\rangle$, $|1\rangle |w_1^{(1-b)}\rangle = |x'\rangle$. Then $|x_4\rangle = |x\rangle$ with probability $\frac{1}{2}$ and $|x_4\rangle = |x'\rangle$ otherwise, and $|x_5\rangle = \frac{1}{\sqrt{2}}(|x\rangle + |x'\rangle)$ with

probability $\frac{1}{2}$ and $|x_5\rangle = \frac{1}{\sqrt{2}}(|x\rangle - |x'\rangle)$ otherwise. Let ρ_4 and ρ_5 be the density matrices corresponding to $|x_4\rangle$ and $|x_5\rangle$, then we have,

$$\rho_4 = \frac{1}{2}|x\rangle\langle x| + \frac{1}{2}|x'\rangle\langle x'| \tag{5.25}$$

$$\rho_5 = \frac{1}{2}(\frac{1}{\sqrt{2}}(|x\rangle + |x'\rangle))(\frac{1}{\sqrt{2}}(\langle x| + \langle x'|)) + \frac{1}{2}(\frac{1}{\sqrt{2}}(|x\rangle - |x'\rangle))(\frac{1}{\sqrt{2}}(\langle x| - \langle x'|)) \tag{5.26}$$

$$= \frac{1}{4}((|x\rangle\langle x| + |x\rangle\langle x'| + |x'\rangle\langle x| + |x'\rangle\langle x'|) + (|x\rangle\langle x| - |x\rangle\langle x'| - |x'\rangle\langle x| + |x'\rangle\langle x'|)) \tag{5.27}$$

$$= \frac{1}{4}(2|x\rangle\langle x| + 2|x'\rangle\langle x'|) \tag{5.28}$$

$$= \frac{1}{2}|x\rangle\langle x| + \frac{1}{2}|x'\rangle\langle x'| \tag{5.29}$$

So the density matrices of $|x_4\rangle$ and $|x_5\rangle$ are the same, both equal $\frac{1}{2}(|x\rangle\langle x| + |x'\rangle\langle x'|)$, it is therefore impossible to distinguish $|x_4\rangle$ and $|x_5\rangle$. So $\mathsf{World}_4 = \mathsf{World}_5$.

6. $\mathsf{World}_5 = \mathsf{World}_6$: Since World_6 is a reorder of World_5, so the distribution of outputs are the same as in World_5. So $\mathsf{World}_5 = \mathsf{World}_6$.

By transitivity, we conclude that $\mathsf{World}_0 \approx \mathsf{World}_6$, which ends the proof of this part.

5.3.2 Malicious Alice

We define World_0 to World_4 as follows:

- World_0 (Fig. 5.8) is the real world.
- World_1 (Fig. 5.9) is World_0 except that the protocol that generates F is replaced by the simulated version.
- World_2 (Fig. 5.10) is World_1 except that the ZK protocol is replaced by the simulated version.
- World_3 (Fig. 5.11) is World_2 except that we do not perform the Z rotation for $|\psi\rangle^{(1-b)}$.
- World_4 (Fig. 5.12) is a reorder of the operations of World_3, and is the ideal world. More precisely, we can see that all the interaction and calculation

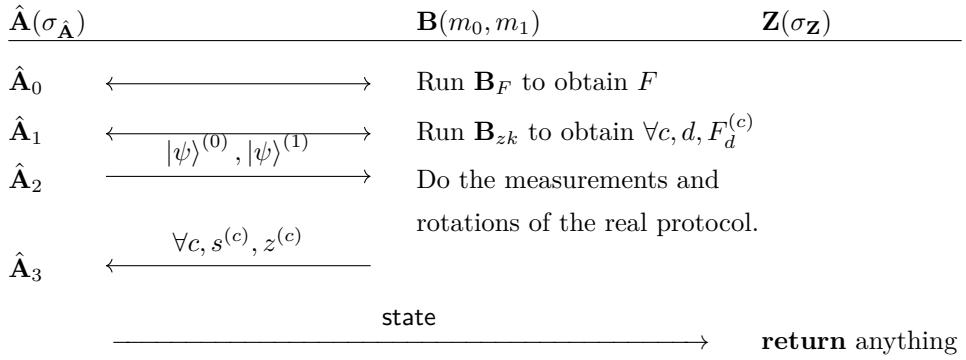


Figure 5.8: Case 3 (malicious Alice): World_0

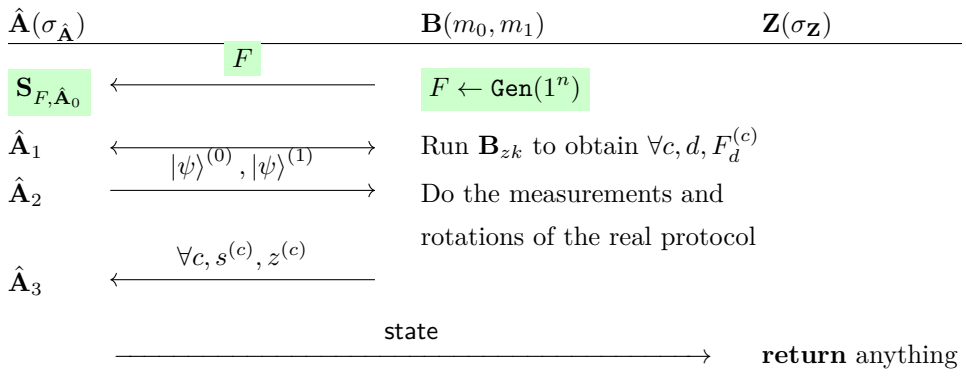


Figure 5.9: Case 3 (malicious Alice): World_1

in World_3 are independent of m_{1-b}, so we can just reorder these parts to Alice's part to construct a simulator that first forwards b, and gets m_b from the remaining part of Bob, which is seperated into 2 parts: the idealized functionality receives $m^{(0)}$ and $m^{(1)}$ from Bob and b from Alice, and outputs $m^{(b)}$ to Alice, and the corresponding idealized party of Bob that forwards $m^{(0)}$ and $m^{(1)}$ to the functionality.

1. $\mathsf{World}_0 \approx \mathsf{World}_1$: Any distinguisher that distinguishes World_0 and World_1 can be converted to a distinguisher that distinguishes the protocol of obtaining F and the idealized functionality $\mathcal{F}_F^{Gen}$. We assumed that $\mathcal{F}_F^{Gen}$ is realized by the protocol of obtaining F, so such distinguisher does not exist. So $\mathsf{World}_0 \approx \mathsf{World}_1$.

2. $\mathsf{World}_1 \approx \mathsf{World}_2$: Any distinguisher that distinguishes World_1 and World_2 can be converted to a distinguisher that distinguishes the ZK protocol

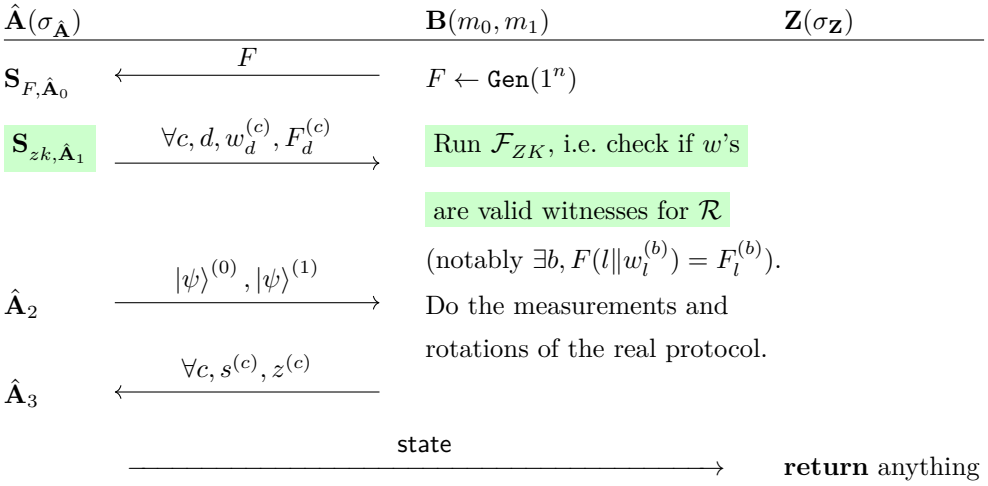


Figure 5.10: Case 3 (malicious Alice): World_2

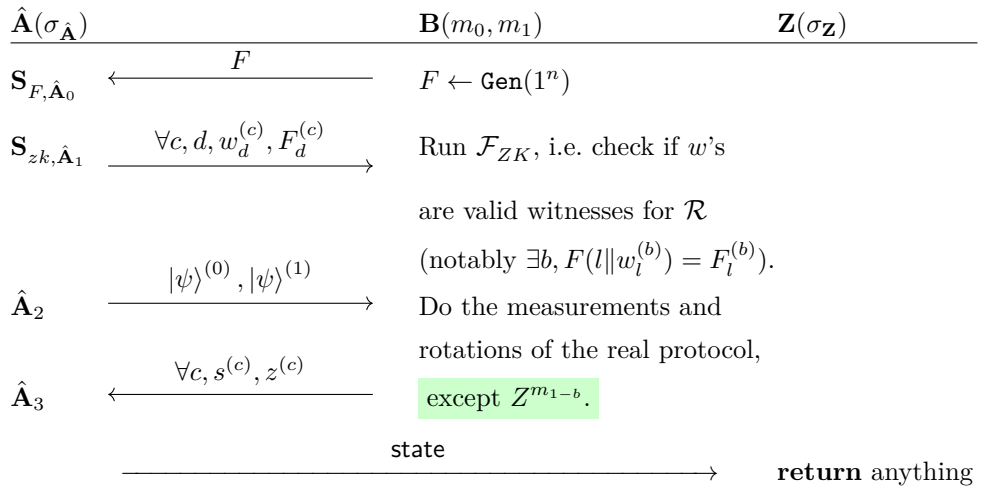


Figure 5.11: Case 3 (malicious Alice): World_3

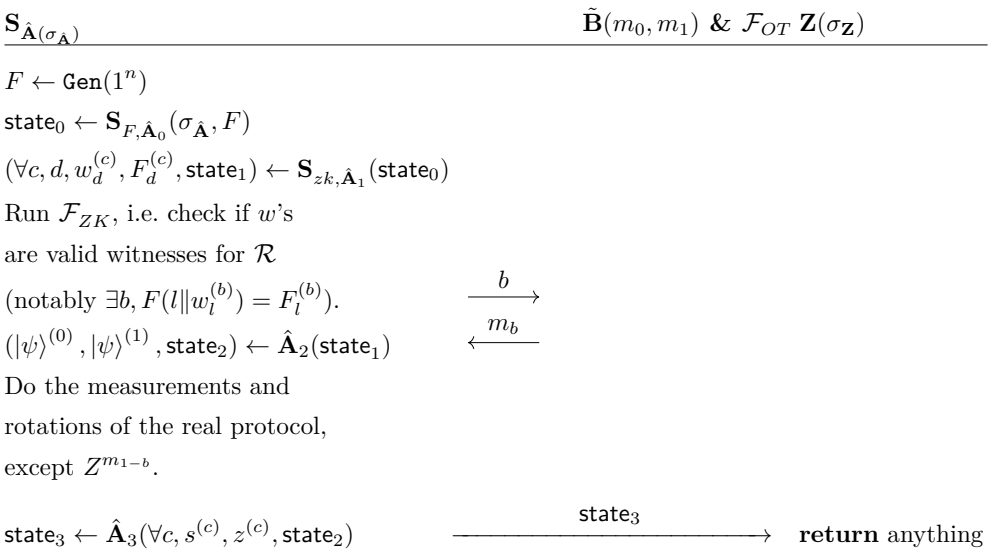


Figure 5.12: Case 3 (malicious Alice): World_4

and the idealized functionality $\mathcal{F}_{ZK}^{\mathcal{R}}$. We assumed that $\mathcal{F}_{ZK}^{\mathcal{R}}$ is realized by a ZK protocol, so such distinguisher does not exist. So $\mathsf{World}_1 \approx \mathsf{World}_2$.

3. $\mathsf{World}_2 \approx \mathsf{World}_3$: In this part, we consider 2 cases, depending on whether F is a permutation or a collision-resistant function.

   - We first consider the case when F is a permutation.
     For $|\psi\rangle^{(1-b)}$, we consider the step of applying the unitary U and measuring the last qubit. If the outcome is 0, then the whole procedure halts and in this case World_2 is identical to World_3.
     Then we consider the case where the measurement outcome is 1. This means that the state collapses to a superposition of $\sum \alpha_j |i, v, x, u\rangle$ where $h(i||v, x||u) \neq 1 \wedge \exists d, F(i||v, x||u) = F_d^{(c)}$. By injectivity, only 2 such strings satisfy $\exists d, F(i||v, x||u) = F_d^{(c)}$. And by the ZK proof, we know that only one of the two strings satisfies $h(i||v, x||u) \neq 1$. This means that the state is $|i, v, x, u\rangle$ for some string i, v, x, u. Applying $Z^{m_{1-b}}$ only adds a global phase flip to the state, which is impossible to distinguish from the original state. So in this case we also have $\mathsf{World}_2 = \mathsf{World}_3$
     So $\mathsf{World}_2 = \mathsf{World}_3$.
   - Then we consider when F is not necessarily injective, but still holds the property of collision-resistance.

For $|\psi\rangle^{(1-b)}$, we consider the step of applying the unitary U and measuring the last qubit. The state collapses to $|\psi\rangle = (\sum \alpha_j |i, v, x, u\rangle) |t\rangle$. If $t = 0$, then the whole process stops. If $t = 1$, then $h(i||v, x||u) \neq 1 \wedge \exists d, F(i||v, x||u) = F_d^{(c)}$. We define x^* to be the string such that $F(x^*) = F_l^{(1-b)}$, then we can always write the remaining state (with some auxiliary qubits) to a pure state $|\psi\rangle = (\sqrt{\beta} |\phi\rangle + \sqrt{1-\beta} |x^*\rangle |\phi^*\rangle) |t\rangle$ such that $\mathrm{Tr}((|x^*\rangle \langle x^*| \otimes I) |\phi\rangle \langle\phi|) = 0$.
Now we have 2 observations: Firstly, we observe that by measuring the state $|\psi\rangle$, we can find a collision with probability $\geq t\beta$.
Secondly, we can observe that the trace distance between $|\psi\rangle$ and the state after the (possible) Z operation is $\leq 2t\sqrt{\beta}$:
For the case $t = 0$, the Z rotation is not performed, so the trace distance between these 2 states is 0.
For case $t = 1$, by using the triangle inequality we have the following:

$$\begin{aligned}
&\mathrm{TD}(|\psi\rangle, Z^{m_{1-b}} |\psi\rangle) && (5.30)\\
\leq &\mathrm{TD}(|\psi\rangle, |x^*\rangle |\phi^*\rangle |t\rangle) + \mathrm{TD}(|x^*\rangle |\phi^*\rangle |t\rangle, Z^{m_{1-b}} |x^*\rangle |\phi^*\rangle |t\rangle) && (5.31)\\
&+ \mathrm{TD}(Z^{m_{1-b}} |x^*\rangle |\phi^*\rangle |t\rangle, Z^{m_{1-b}} |\psi\rangle) && (5.32)\\
= &\sqrt{\beta} + 0 + \sqrt{\beta} = 2\sqrt{\beta} = 2t\sqrt{\beta} && (5.33)
\end{aligned}$$

Then, for any distinguisher trying to distinguish World_2 and World_3, we can define 2 operations: ξ_0 is the combination of all the operation before the (possible) Z rotation on $|\psi\rangle$, and ξ_1 is the combination of all the operation after the (possible) Z rotation on $|\psi\rangle$. These operations can be converted to the form of POVM. Then, by Theorem 2.8, we have

$$\begin{aligned}
&|\Pr[\mathsf{World}_3 = 1] - \Pr[\mathsf{World}_2 = 1]| && (5.34)\\
\leq &\mathbb{E}_{|\psi\rangle \leftarrow \xi_0(\sigma)} |\Pr[\xi_1 |\psi\rangle = 1] - \Pr[\xi_1 Z^{m_{1-b}} |\psi\rangle = 1]| && (5.35)\\
= &\mathbb{E}_{|\psi\rangle \leftarrow \xi_0(\sigma)} \frac{1}{2} (|\Pr[\xi_1 |\psi\rangle = 1] - \Pr[\xi_1 Z^{m_{1-b}} |\psi\rangle = 1]| && (5.36)\\
&+ |\Pr[\xi_1 |\psi\rangle = 0] - \Pr[\xi_1 Z^{m_{1-b}} |\psi\rangle = 0]|) && (5.37)\\
\leq &\mathbb{E}_{|\psi\rangle \leftarrow \xi_0(\sigma)} \mathrm{TD}(|\psi\rangle, Z^{m_{1-b}} |\psi\rangle) && (5.38)\\
\leq &\mathbb{E}_{|\psi\rangle \leftarrow \xi_0(\sigma)} 2t\sqrt{\beta_\psi} && (5.39)
\end{aligned}$$

On the other hand, the average probability of finding a collision is $\mathbb{E}_{|\psi\rangle \leftarrow \xi_0(\sigma)} t\beta_\psi$.

Since $\mathbb{E}_{|\psi\rangle\leftarrow\xi_0(\sigma)} 2t\sqrt{\beta_\psi} \leq 2\sqrt{\mathbb{E}_{|\psi\rangle\leftarrow\xi_0(\sigma)} t\beta_\psi}$, which means that $|\Pr[\mathsf{World}_3 = 1] - \Pr[\mathsf{World}_2 = 1]| \leq 2\sqrt{\mathbb{E}_{|\psi\rangle\leftarrow\xi_0(\sigma)} t\beta_\psi}$, Also, the possibility to find a collision is negligible, that is, $\mathbb{E}_{|\psi\rangle\leftarrow\xi_0(\sigma)} t\beta_\psi$ is negligible, which means that $2\sqrt{\mathbb{E}_{|\psi\rangle\leftarrow\xi_0(\sigma)} t\beta_\psi}$ is also negligible. As a result, we can conclude that $|\Pr[\mathsf{World}_3 = 1] - \Pr[\mathsf{World}_2 = 1]|$ is negligible.
So $\mathsf{World}_2 \approx \mathsf{World}_3$.

To conclude the above argument, we show that $\mathsf{World}_2 \approx \mathsf{World}_3$.

4. $\mathsf{World}_3 = \mathsf{World}_4$: Since World_4 is a reorder of World_3, so the distribution of outputs are the same as in World_3. So $\mathsf{World}_3 = \mathsf{World}_4$.

By transitivity, we conclude that $\mathsf{World}_0 \approx \mathsf{World}_4$, which ends the proof of this part.

Chapter 6

Conclusion

6.1 Summary

The oblivious transfer protocols can be used to construct multi-party computation protocols, that can play a vital role in fields like auctions and electronic votes. Classical OT protocols require trapdoor functions, while a 2-message quantum OT protocol in [CMS23] constructed out of (non-interactive) zero-knowledge proof requires the assumption of collision-resistant hiding functions. However, since specific assumptions might be broken in the future, it is natural to study whether weaker assumptions can be applied to construct a protocol for OT.

In this thesis, we show that we can construct a quantum OT protocol out of (non-interactive) zero-knowledge proof that does not require the hiding property of the function, but only requires the assumption of length-preserving collision-resistant quantum one-way functions, while keeping the protocol optimal in communication.

The generalized quantum Goldreich-Levin Theorem plays a vital role in the construction of the protocol. This theorem mainly states that any length-preserving quantum one-way function can be converted to a quantum one-way function with a quantum hard-predicate, where the quantum hard-predicate plays the role of the hiding bit to hide information in the protocol. An original version of the theorem is shown in [AC02], that only applies to quantum one-way permutations. We generalize the results to length-preserving quantum one-way functions by defining a generalized version of the GL problem and reduce the problem of inverting the function to the generalized GL problem.

We use a quantum stand-alone security model described in [HSS11] to prove the security of the protocol, that is, given the existence of collision-resistant length-preserving quantum one-way functions and the existence of non-interactive ZK proof, we can obtain a 2-message OT protocol that is computational secure.

6.2 Future Work

The method also raises a number of open questions.

Removing length-preserving property. The generalized Goldreich-Levin Theorem requires the assumption of being length-preserving. We expect that the theorem also holds without the assumption of length-preserving, thus we can remove the assumption from the protocol.

Even weaker assumptions. [JLS18] introduces the notion of Pseudorandom Quantum States, which can be constructed assuming quantum one-way functions. This weaker assumption (compared with quantum one-way functions) can be used to build OT protocols [AQY22]. It is interesting to study whether it can be applied to a protocol similar to the one described in this thesis, thus leading to more efficient protocols.

Bibliography

[AC02] M. Adcock and R. Cleve. A Quantum Goldreich-Levin Theorem with Cryptographic Applications. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 323–334. Springer, 2002.

[AQY22] P. Ananth, L. Qian, and H. Yuen. Cryptography from Pseudorandom Quantum States. In Y. Dodis and T. Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022*, Lecture Notes in Computer Science, pages 208–236. Springer Nature Switzerland, 2022.

[BCK+21] J. Bartusek, A. Coladangelo, D. Khurana, and F. Ma. One-Way Functions Imply Secure Computation in a Quantum World. In T. Malkin and C. Peikert, editors, *Advances in Cryptology – CRYPTO 2021*, Lecture Notes in Computer Science, pages 467–496. Springer International Publishing, 2021.

[BD18] Z. Brakerski and N. Döttling. Two-message Statistically Sender-private OT from LWE. In *Theory of Cryptography: 16th International Conference, TCC 2018, Panaji, India, November 11–14, 2018, Proceedings, Part II 16*, pages 370–390. Springer, 2018.

[BHM+02] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp. Quantum Amplitude Amplification and Estimation. *Contemporary Mathematics*, 305:53–74, 2002.

[CD23] W. Castryck and T. Decru. An Efficient Key Recovery Attack on SIDH. In C. Hazay and M. Stam, editors, *Advances in Cryptology – EUROCRYPT 2023*, pages 423–447. Springer Nature Switzerland, 2023.

[Che24] Y. Chen. Quantum Algorithms for Lattice Problems. Cryptology ePrint Archive, Paper 2024/555, 2024.

[CMS23] L. Colisson, G. Muguruza, and F. Speelman. Oblivious Transfer from Zero-Knowledge Proofs: Or How to Achieve Round-Optimal Quantum Oblivious Transfer and Zero-Knowledge Proofs on Quantum States. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 3–38. Springer, 2023.

[EGL85] S. Even, O. Goldreich, and A. Lempel. A Randomized Protocol for Signing Contracts. *Communications of the ACM*, 28(6):637–647, 1985.

[GL89] O. Goldreich and L. A. Levin. A Hard-Core Predicate for All One-Way Functions. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, pages 25–32, New York, NY, USA. Association for Computing Machinery, February 1, 1989.

[GLS+21] A. B. Grilo, H. Lin, F. Song, and V. Vaikuntanathan. Oblivious Transfer Is in MiniQCrypt. In A. Canteaut and F.-X. Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021*, Lecture Notes in Computer Science, pages 531–561. Springer International Publishing, 2021.

[HSS11] S. Hallgren, A. Smith, and F. Song. Classical Cryptographic Protocols in a Quantum World. In P. Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, Lecture Notes in Computer Science, pages 411–428, Berlin, Heidelberg. Springer, 2011.

[Imp95] R. Impagliazzo. A Personal View of Average-Case Complexity. In *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*. Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference, pages 134–147, June 1995.

[JLS18] Z. Ji, Y.-K. Liu, and F. Song. Pseudorandom Quantum States. In H. Shacham and A. Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018*, Lecture Notes in Computer Science, pages 126–152. Springer International Publishing, 2018.

[Kil88] J. Kilian. Founding Crytpography on Oblivious Transfer. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, pages 20–31, New York, NY, USA. Association for Computing Machinery, January 1, 1988.

[NC10] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[PVW08] C. Peikert, V. Vaikuntanathan, and B. Waters. A Framework for Efficient and Composable Oblivious Transfer. In D. Wagner, editor, *Advances in Cryptology – CRYPTO 2008*, Lecture Notes in Computer Science, pages 554–571, Berlin, Heidelberg. Springer, 2008.

[Qua20] W. Quach. UC-Secure OT from LWE, Revisited. In C. Galdi and V. Kolesnikov, editors, *Security and Cryptography for Networks*, Lecture Notes in Computer Science, pages 192–211. Springer International Publishing, 2020.

[Unr15] D. Unruh. Non-Interactive Zero-Knowledge Proofs in the Quantum Random Oracle Model. In E. Oswald and M. Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015*, Lecture Notes in Computer Science, pages 755–784, Berlin, Heidelberg. Springer, 2015.

[Yao82] A. C. Yao. Protocols for Secure Computations. In *23rd Annual Symposium on Foundations of Computer Science (Sfcs 1982)*. 23rd Annual Symposium on Foundations of Computer Science (Sfcs 1982), pages 160–164, November 1982.

Appendix A

Appendix

A.1 Sampling $w_d^{(c)}$ and the Probability Distribution of $h_d^{(c)}$

In this part, we show that sampling valid $w_d^{(c)} = v_d^{(c)}||x_d^{(c)}||u_d^{(c)}$ can be done in polynomial time, with negligible probability of failure, more formally,

Theorem A.1. *For any c, b, by sampling $w_d^{(c)} \xleftarrow{\$} \{0,1\}^{2n-1}$ $O(n)$ times, with probability $1-\mathsf{negl}(n)$, one can obtain some $w_d^{(c)}$ such that $\langle c||v_d^{(c)}, x_d^{(c)}||u_d^{(c)}\rangle = b$.*

We also show:

Theorem A.2. *for every b, $\Pr_{x\xleftarrow{\$}\{0,1\}^n, y\xleftarrow{\$}\{0,1\}^n}[\langle y, x\rangle = b] = \frac{1}{2} \pm \mathsf{negl}(n)$.*

The following main lemma proves the above 2 theorems:

Lemma A.3 (Main Lemma)**.** *For any c, b, $\Pr_{u\xleftarrow{\$}\{0,1\}^{n-1}, v\xleftarrow{\$}\{0,1\}^{n-1}, x\xleftarrow{\$}\{0,1\}}[\langle c||v, x||u\rangle = b] = \frac{1}{2} \pm \mathsf{negl}(n)$*

To prove the main lemma, we have the following lemmas:

Lemma A.4. $\Pr_{u\xleftarrow{\$}\{0,1\}^{n-1}}[\langle 0^{n-1}, u\rangle = 1] = 0$

Proof. Trivial. □

Lemma A.5. *For $v \in \{0,1\}^{n-1} \setminus \{0^n\}$, $\Pr_{u\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v, u\rangle = 1] = \frac{1}{2}$*

Proof. Since $v \neq 0^n$, we may assume that $v_k = 1$.

Then we have

$$\Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[\langle v, u \rangle = 1] \tag{A.1}$$

$$= \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[v_k u_k \oplus \bigoplus_{i \neq k} v_i u_i = 1] \tag{A.2}$$

$$= \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[u_k \oplus \bigoplus_{i \neq k} v_i u_i = 1] \tag{A.3}$$

$$= \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[u_k = 0] \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[u_k \oplus \bigoplus_{i \neq k} v_i u_i = 1 | u_k = 0] \tag{A.4}$$

$$+ \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[u_k = 1] \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[u_k \oplus \bigoplus_{i \neq k} v_i u_i = 1 | u_k = 1] \tag{A.5}$$

$$= \frac{1}{2} \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[\bigoplus_{i \neq k} v_i u_i = 1] + \frac{1}{2} \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[\bigoplus_{i \neq k} v_i u_i = 0] \tag{A.6}$$

$$= \frac{1}{2} \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[\bigoplus_{i \neq k} v_i u_i = 1] + \frac{1}{2} \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}}[\bigoplus_{i \neq k} v_i u_i \neq 1] \tag{A.7}$$

$$= \frac{1}{2} \tag{A.8}$$

□

Proof. [Proof of Lemma A.3] Realize that

$$\Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}, v \xleftarrow{\$} \{0,1\}^{n-1}, x \xleftarrow{\$} \{0,1\}}[\langle 0||v, x||u \rangle = 1] + \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}, v \xleftarrow{\$} \{0,1\}^{n-1}, x \xleftarrow{\$} \{0,1\}}[\langle 0||v, x||u \rangle = 0] = 1 \tag{A.9}$$

and

$$\Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}, v \xleftarrow{\$} \{0,1\}^{n-1}, x \xleftarrow{\$} \{0,1\}}[\langle 1||v, x||u \rangle = 1] + \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}, v \xleftarrow{\$} \{0,1\}^{n-1}, x \xleftarrow{\$} \{0,1\}}[\langle 1||v, x||u \rangle = 0] = 1 \tag{A.10}$$

We just need to prove the following:

1.

$$\Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}, v \xleftarrow{\$} \{0,1\}^{n-1}, x \xleftarrow{\$} \{0,1\}}[\langle 0||v, x||u \rangle = 1] = \frac{1}{2} \pm \mathsf{negl}(n) \tag{A.11}$$

2.

$$\Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1},x\xleftarrow{\$}\{0,1\}}[\langle 1||v,x||u\rangle = 1] = \frac{1}{2} \pm \mathsf{negl}(n) \tag{A.12}$$

1.

$$\Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1},x\xleftarrow{\$}\{0,1\}}[\langle 0||v,x||u\rangle = 1] \tag{A.13}$$

$$= \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v,u\rangle = 1] \tag{A.14}$$

$$= \sum_{v'\in\{0,1\}^{n-1}} \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[v=v'] \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v,u\rangle = 1|v=v'] \tag{A.15}$$

$$= \sum_{v'\in\{0,1\}^{n-1}\setminus\{0^n\}} \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[v=v'] \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v,u\rangle = 1|v=v'] \tag{A.16}$$

$$+ \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[v=0^n] \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v,u\rangle = 1|v=0^n] \tag{A.17}$$

$$= \sum_{v'\in\{0,1\}^{n-1}} \frac{1}{2^{n-1}} \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v',u\rangle = 1] + \frac{1}{2^{n-1}} \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1}}[\langle 0^n,u\rangle = 1] \tag{A.18}$$

$$=(2^{n-1}-1)\frac{1}{2^{n-1}}\frac{1}{2} + 0 \tag{A.19}$$

$$=\frac{1}{2} - \frac{1}{2^n} \tag{A.20}$$

2.

$$\Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1},x\xleftarrow{\$}\{0,1\}}[\langle 1||v,x||u\rangle = 1] \tag{A.21}$$

$$= \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1},x\xleftarrow{\$}\{0,1\}}[x=0] \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1},x\xleftarrow{\$}\{0,1\}}[\langle 1||v,x||u\rangle = 1|x=0] \tag{A.22}$$

$$+ \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1},x\xleftarrow{\$}\{0,1\}}[x=1] \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1},x\xleftarrow{\$}\{0,1\}}[\langle 1||v,x||u\rangle = 1|x=1] \tag{A.23}$$

$$=\frac{1}{2} \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v,u\rangle = 1] + \frac{1}{2} \Pr_{u\xleftarrow{\$}\{0,1\}^{n-1},v\xleftarrow{\$}\{0,1\}^{n-1}}[\langle v,u\rangle = 0] \tag{A.24}$$

$$= \frac{1}{2} \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}, v \xleftarrow{\$} \{0,1\}^{n-1}} [\langle v, u \rangle = 1] + \frac{1}{2} \Pr_{u \xleftarrow{\$} \{0,1\}^{n-1}, v \xleftarrow{\$} \{0,1\}^{n-1}} [\langle v, u \rangle \neq 1] \tag{A.25}$$

$$= \frac{1}{2} \tag{A.26}$$

□