Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Quantum Fair Exchange

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Μάριου Γεωργίου

Επιβλέπων:
Αριστείδης Παγουρτζής
Επίκ. Καθηγητής Ε.Μ.Π.

Συνεπιβλέπων:
Ιορδάνης Κερενίδης
Permanent CNRS Researcher

Εργαστήριο Λογικής και Επιστήμης Υπολογισμών

Αθήνα, Μάρτιος 2013

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Λογικής και Επιστήμης Υπολογισμών

Μάριος Γεωργίου
Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Ευχαριστίες

Θέλω να ευχαριστήσω θερμά τα μέλη της επιτροπής κ. Ζάχο, κ. Παγουρτζή και κ. Φωτάκη για την εμπιστοσύνη που μου έδειξαν και για τη στήριξή τους τόσο σε ακαδημαϊκό όσο και σε προσωπικό επίπεδο.

Ιδιαίτερα θα ήθελα να ευχαριστήσω τους επιβλέποντες καθηγητές μου κ. Παγουρτζή και κ. Κερενίδη για την συνεχή καθοδήγηση τους και για τις ανεκτίμητες επιστημονικές συμβουλές και συζητήσεις που με ενέπνευσαν και με παρακίνησαν σε αυτήν την προσπάθεια.

Επίσης, ένα μεγάλο ευχαριστώ οφείλω στην Άννα Παππά για τις εποικοδομητικές μας συζητήσεις, για τις ώρες που αφιέρωσε για να μελετήσει την εργασία μου καθώς και για τις πολύτιμες παρατηρήσεις και διορθώσεις της.

Ευχαριστώ θερμά τους κρυπτογράφους Ελένη Μπακάλη, Δημήτρη Σακαβάλα, Γιώργο Ζιρδέλη και Χρήστο Λίτσα καθώς και όλα τα υπόλοιπα μέλη του εργαστηρίου για τη βοήθεια, την υποστήριξη και τους δημιουργικούς διαλόγους που είχαμε.

Τέλος, ευχαριστώ πάρα πολύ την οικογένειά μου και τους φίλους μου, που πάντα με υποστηρίζουν και με βοηθάνε.

Αθήνα, 4 Μαρτίου 2013 Μ. Γ.

Περίληψη

Μελετάμε το κβαντικό ανάλογο του κρυπτογραφικού προβλήματος της δίκαιης ανταλλαγής. Σε μία δίκαιη ανταλλαγή θέλουμε να εξασφαλίσουμε ότι δύο πρόσωπα είτε θα ανταλλάξουν τα μυστικά τους, είτε κανείς από τους δύο δε θα μάθει το μυστικό του άλλου. Πιο συγκεκριμένα τα δύο αυτά πρόσωπα, έστω η Αλίκη και ο Βασίλης (Α και Β), αλληλεπιδρούν μεταξύ τους τρέχοντας ένα πρωτόκολλο δίκαιης ανταλλαγής. Απαιτούμε δύο βασικές ιδιότητες:

1. Ορθότητα: Όταν και οι δύο παίκτες παίζουν τίμια (ακολουθούν το πρωτόκολλο) τότε στο τέλος μαθαίνουν και οι δύο το μυστικό.

2. Πληρότητα: Όταν ένας από τους δύο παίκτες (έστω η Αλίκη) αποκλίνει αυθαίρετα από το πρωτόκολλο (δηλαδή κλέβει με οποιοδήποτε τρόπο), τότε είτε θα πρέπει και οι δύο να μάθουν το μυστικό, είτε να μην το μάθει κανείς. Με άλλα λόγια, ακόμη κι αν κλέβει η Αλίκη, να μη βρεθεί σε μειονεκτική θέση ο Βασίλης.

Προτείνουμε δύο διαφορετικές προσεγγίσεις για τη λύση του προβλήματος.
Στην πρώτη προσέγγιση, δίνουμε έναν παραπλήσιο ορισμό, αυτόν της ταυτόχρονης ανταλλαγής. Σε ένα τέτοιο πρωτόκολλο, απαιτούμε το εξής: σε κάθε στιγμή του πρωτοκόλλου, η πιθανότητα η Αλίκη να μαντέψει το μυστικό του Βασίλη είναι σχεδόν ίδια με την πιθανότητα ο Βασίλης να μαντέψει το μυστικό της Αλίκης. Σε αυτήν την περίπτωση μπορούμε να εξασφαλίσουμε ασφάλεια από πληροφοριοθεωρητικής σκοπιάς: κατασκευάζουμε κβαντικό πρωτόκολλο τέτοιο ώστε ακόμα και ένας υπολογιστικά παντοδύναμος παίχτης να μην μπορεί να το παραβιάσει. Τονίζουμε ότι με τις κλασσικές μεθόδους είναι αδύνατο να επιτύχουμε απόλυτη ασφάλεια.
Στη δεύτερη προσέγγιση χρησιμοποιούμε έναν αρκετά διαφορετικό ορισμό, αυτόν του Coin Ripping. Εδώ, η Αλίκη θέλει να ανταλλάξει χρήματα με κάποιο προϊόν. Χρησιμοποιώντας κάποια πρόσφατα αποτελέσματα όπως ασφαλή κβαντικά νομίσματα δημοσίου κλειδιού και αποδείξεις μηδενικής γνώσης ασφαλείς ενάντια σε κβαντικούς αντιπάλους δημιουργούμε ένα πρωτόκολλο τέτοιο ώστε:

1. Αν η Αλίκη κλέψει τότε το καλύτερο που μπορεί να πετύχει είναι να πάρει το προϊόν και να αποτρέψει τον Βασίλη από το να πληρωθεί, αλλά η ίδια θα το χάσει το χαρτονόμισμά της.

2. Αν ο Βασίλης κλέψει τότε το καλύτερο που μπορεί να πετύχει είναι να αναγκάσει την Αλίκη να χάσει το χαρτονόμισμά της, αλλά ο ίδιος δε θα το αποκτήσει.

Με άλλα λόγια, δεν υπάρχει στρατηγική που να τους ευνοήσει αλλά υπάρχει στρατηγική που μπορεί να βλάψει τον αντίπαλο. Σε αυτήν την περίπτωση μπορούμε να εξασφαλίσουμε υπολογιστική ασφάλεια: ένας πολυωνυμικά φραγμένος κβαντικός αντίπαλος έχει αμελητέα πιθανότητα να το παραβιάσει. Τονίζουμε ότι με τις κλασσικές μεθόδους δεν μπορούμε να επιτύχουμε κάτι τέτοιο καθώς δεν υπάρχει σχήμα που να εξασφαλίζει την δημόσια επαλήθευση ενός νομίσματος (είναι απαραίτητη η χρήση κάποια έμπιστης αρχής που θα μπορεί να επαληθεύσει τη γνησιότητα των νομισμάτων).

Abstract

We study the Quantum analogue of Fair Exchange. In a Fair Exchange we want to guarantee that two parties either will exchange their secrets or neither will learn each other's secret. More specifically, the two parties, say Alice and Bob (A and B), interact running a Fair-Exchange protocol. We require two properties:

1. Completeness: When both parties are honest, then at the end they successfully exchange their secrets.

2. Soundness: When a party is dishonest then, either both learn the secrets or neither does. In other words, even if Alice is cheating, Bob will never be disadvantaged.

We suggest two different approaches to solve the problem.
In the first one we give a slightly different definition, that of Simultaneous Exchange. In such a protocol, we require the following: in every moment of the protocol, Alice's probability of guessing Bob's secret is almost the same as Bob's probability of guessing Alice's secret. In this case we can guarantee information theoretic security: we create a quantum protocol such that even a computationally unbounded adversary cannot break it. Note that classically it is impossible to guarantee perfect security.
In the second approach we use a different definition; namely the Coin Ripping. Now Alice wants to exchange money for some product. Using some recent results such as public Key Quantum Money and Quantum secure Zero-Knowledge Proofs we create a protocol such that:

1. If Alice is cheating then the best she can succeed is to take the product and prevent Bob from being paid, but she will surely lose her coin.

2. If Bob is cheating then the best he can succeed is to make Alice lose her coin, but he will not get the coin.

In other words, there is no strategy that they can use in their favor, but there is a strategy that can harm the honest one. In this case we can guarantee computational security: a polynomially bound adversary has negligible probability of breaking the protocol. Note that classically we cannot achieve such a protocol since there is no public key Quantum Money scheme.

Contents

List of Figures

Introduction

The problem of *Fair exchange* is one of the most important cryptographic problems of Modern Cryptography. Informally, suppose that Alice and Bob have secrets s_A, s_B respectively and they want to run a protocol so that either both will learn each other's secret or none. However, the two parties do not trust each other; Alice doesn't trust Bob that if she first tells him the secret, he will respond; the same holds for Bob. So we need a protocol to address this deadlock.
Already from the early 80's Even [9] proved an impossibility result; there is no classical protocol for fair exchange that does not use any Trusted Third Party (TTP). Since then, much work has been done in order to reduce the intervention of the TTP. In fact, the problem can be solved optimally using a TTP that takes both secrets and then sends Alice s_B and Bob s_A. However, these kinds of protocols pose heavy work to the TTP. The latest results try to exploit the TTP as less as possible and at the best scenario to use it only in exceptional cases (dishonest behavior or channel corruption).
In this thesis we will address this problem using the powerful tools of Quantum Mechanics. We will show that in the quantum world we can succeed much better results than in the classical counterpart. We will approach the problem using two different definitions and then give optimal protocols satisfying them. We note, that much less work has been made quantumly. In particular, the paper of Paunković et al [16] is (to the best of our knowledge) the only quantum protocol for contract signing (an application of fair exchange where the parties exchange their digital signature on a common contract).

Quantum mechanics have many applications to Modern Cryptography. The real power of quantum mechanics, lies in the fact that a measurement disturbs the system to be measured; the system collapses. Exploiting this advantage, Bennett and Brassard [4] in 1984 created the famous unconditionally secure protocol for key exchange BB84, paving the way for the the new era of *Quantum Cryptography*. Since then, much progress has been made to create more secure protocols based on the laws of quantum mechanics. In this work, we will use many of these results to create secure fair exchange protocols.
In particular, one of the most useful results is the work of Mochon [14] where he proved that we can have unconditionally secure balanced weak coin flipping and the work of Chailloux and Kerenidis [8] where they proved that we can have unconditionally secure unbalanced weak coin flipping. Using these results we will create a protocol for *simultaneous* exchange

of a single bit.
Another interesting option for fair exchange is the work of Jakobsson [12] where he proposes a totally different way of fair exchange; the idea of *ripping* a banknote. In this scenario Alice exchanges money for a product. As before, she doesn't trust Bob to pay him before getting the product, neither Bob trusts her to send her the product before getting paid. The idea is to let Alice rip the banknote (sometimes we will refer to the banknote as coin) in two halfs and send only the one part to Bob. Bob can verify that this part is valid and send the product, being sure that Alice can't use the other part as a full banknote. To avoid the need of a TTP this scheme requires at least a publicly verifiable quantum money scheme. Informally, a money scheme is publicly verifiable if there is no need of a TTP (or Bank) to verify the validity of the coins; the users can verify the coins just like in the real world. A second requirement for our protocol is the existence of quantum-secure Zero-Knowledge Proofs of Knowledge.
Recently, both problems were almost adressed. Aaronson and Christano [1] approached the problem of publicly verifiable quantum money with good candidate protocols. One of the main ideas of their paper is to compose digital signatures and quantum money *Mini-Schemes* to create a full quantum money scheme. Furthermore, the work of Watrous [20] proving quantum-secure Zero-Knowledge Proofs together with the work of Unruh [19] proving quantum-secure Proofs of Knowledge completes the toolbox needed for creating a quantum-secure ripping coins protocol.

The following chapters are organized as follows. In the first Chapter we make an introduction to the basic quantum concepts. It consists of three sections. In the first section we present the quantum computation principles and in the second the quantum information principles. In the third section we analyze the distance measures of quantum states that are necessary for our work. In the second Chapter we present the Quantum Tools that will be used for our constructions. It consists of three sections. In the first section we define Zero-Knowledge Proofs and Zero-Knowledge Proofs of Knowledge. In the second section we define some cryptographic primitives and in the third section we give the definition of a quantum money scheme and the results of [1]. In the third Chapter we define the Ripping Quantum Coins Scheme and we give a proof that there exists a secure protocol for ripping coins. In the fourth Chapter we define *simultaneous* exchange and we give a proof that there exists a protocol with arbitrarily small error for simultaneous exchange.

1 Basic Quantum Principles

Let's consider for a while the classical bit. A bit b can take the values 0,1 and these values are usually represented by two different values of voltage in a classical circuit. When we measure a bit, we simply read its value, and then work with it. Of course, we can copy a single bit; we just read it and learn its value and then we can make as many copies as we want. We define a register as an n-bit string for some n.
In the quantum world things are different; a quantum bit or *qubit* can take many more values than 0,1, a measurement of a qubit doesn't return its value and after the measurement the qubit changes. Also, a copy of an unknown qubit is impossible. These are some of the differences between the two computational models that enable us to do many more things quantumly than classically. Quantum computers, however, do not only offer great computational power, but also provide much more power in the information theory and cryptography.
In this chapter we will introduce the basic Quantum Computation and Quantum Information principles. For an extended introduction to the field see [15].

1.1 Quantum Computation

We begin by giving the basic building block of a quantum computer the qubit and then analyze the computation procedure and give some remarks that need attention.

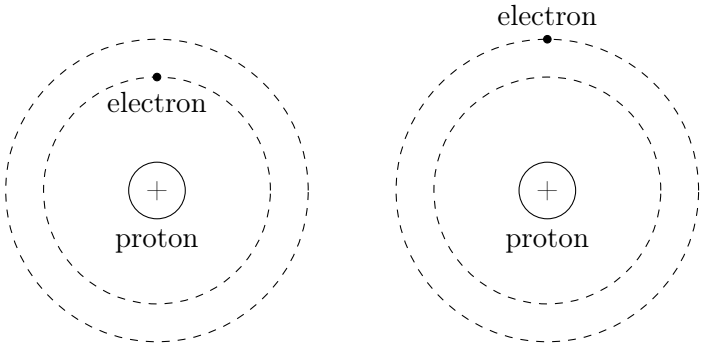
1.1.1 Qubits

Consider a hydrogen atom as a qubit. The state of a hydrogen atom is in general a combination of its ground and its excited state; see figure 1.1. If we assume that the ground and the excited states correspond to bits 0 and 1 respectively, we can say that a qubit can be in a linear combination (or *superposition*) of all the corresponding bits. We use Dirac's notation to represent the state of a qubit. So if $|\psi\rangle$ is the state of a qubit we

can write:

$$|\psi\rangle = a_0|0\rangle + a_1|1\rangle$$

The coefficients a_0, a_1 are in general complex numbers satisfying the requirement



(a) Ground state $|0\rangle$ (b) Excited state $|1\rangle$

Figure 1.1: States of a hydrogen atom

$$|a_0|^2 + |a_1|^2 = 1$$

We can see the state $|\psi\rangle$ as a column vector

$$|\psi\rangle = a_0 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + a_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix}$$

and in the same way we can define the conjugate transpose of the state $|\psi\rangle$ as

$$\langle\psi| = \begin{pmatrix} a_0^* & a_1^* \end{pmatrix}$$

where z^* is the complex conjugate of z.
If we have two qubits $|\psi\rangle = a_0|0\rangle + a_1|1\rangle$ and $|\phi\rangle = b_0|0\rangle + b_1|1\rangle$ we can define the inner product between the two states as

$$\langle\psi|\phi\rangle = \begin{pmatrix} a_0^* & a_1^* \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = a_0^* \cdot b_0 + a_1^* \cdot b_1$$

An intuitive way to think the state of a qubit is as a point on the unit circle as in figure 1.2. In the case of two qubits their state is in a superposition of all the possible classical values of two bits; namely

$$|\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle$$

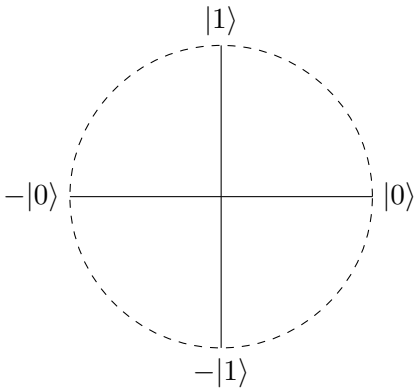


Figure 1.2: Qubit state on the unit cicle

Same as before, the sum of the squares of the coefficients' norm should add up to unity

$$|a_{00}|^2 + |a_{01}|^2 + |a_{10}|^2 + |a_{11}|^2 = 1$$

Generalizing the above procedure we can see that if we have a state $|\psi\rangle$ of n qubits then in general it can be in a superposition of all the corresponding states

$$|\psi\rangle = \sum_{i=1}^{2^n} a_i|i\rangle \text{ with } \sum_{i=1}^{2^n} |a_i|^2 = 1$$

Therefore, we see that quantum computers operate in exponentially more space than classical computers. However, to read a qubit we have to measure it, and the measurement of a qubit does not reveal all this information.

1.1.2 Measurement

The measurement of a qubit is completely different from the measurement of a bit. When we measure a qubit $|\psi\rangle = a_0|0\rangle + a_1|1\rangle$ we take only one bit, 0 or 1. If o is the outcome of the measurement then we have that

$$\Pr[o = |0\rangle] = |\langle\psi|0\rangle|^2 = |a_0|^2 \text{ and } \Pr[o = |1\rangle] = |\langle\psi|1\rangle|^2 = |a_1|^2$$

This is the main reason we need the normalization; if the squares of the coefficients' norms add to unity then they can be probabilities. Another useful remark is that we can see these probabilities as the square of the projection to the corresponding basis vector ($|0\rangle$ or $|1\rangle$).
After the measurement the state of the qubit *collapses* and becomes the outcome of the measurement. So if we measured 0 then the new state becomes $|0\rangle$ and if we measured 1 the new state becomes $|1\rangle$. In the case of two bits things are the same; if we measured 00 then the new state becomes $|00\rangle$ and so on.

What if we measure just a single bit? Suppose as before the state $|\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle$ and we want to measure only the first bit. Then with probability $|a_{00}|^2 + |a_{01}|^2$ the outcome will be 0 and with probability $|a_{10}|^2 + |a_{11}|^2$ it will be 1. If we measure 0 the new state has the form:

$$|\psi'\rangle = \frac{a_{00}|00\rangle + a_{01}|01\rangle}{\sqrt{|a_{00}|^2 + |a_{01}|^2}}$$

and if we measure 1 the new state has the form

$$|\psi'\rangle = \frac{a_{10}|10\rangle + a_{11}|11\rangle}{\sqrt{|a_{10}|^2 + |a_{11}|^2}}$$

This generalizes to n qubits. So despite that quantum computers operate in exponential space, the only way to see the qubits is via a measurement, which returns only n bits of information and even worse it destroys the state.

The measurement we just presented is called a measurement in the *computational* basis ($\{|0\rangle, |1\rangle\}$). This is not the only kind of measurement; in fact, we can measure in any orthonormal basis of the system. So for example we can define

$$|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

$$|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

and we can measure a qubit $|\psi\rangle$ in the basis $B = \{|+\rangle, |-\rangle\}$. B is orthonormal since

$$||+\rangle| = ||-\rangle| = 1 \text{ and also } \langle +|-\rangle = 0$$

So if o is the outcome, we have that

$$\Pr[o = |+\rangle] = |\langle\psi|+\rangle|^2 \text{ and } \Pr[o = |-\rangle] = |\langle\psi|-\rangle|^2$$

and as before the state collapses to $|+\rangle$ or $|-\rangle$.
In the same way, we can measure two qubits in the basis $\{|++\rangle, |+-\rangle, |-+\rangle, |--\rangle\}$ and so on for n qubits.

We note that we can write interchangeably

$$|\psi\rangle|\phi\rangle = |\psi\rangle \otimes |\phi\rangle = |\psi\phi\rangle$$

meaning the tensor product of the two states.

1.1.3 Unitary Operators

The building blocks of quantum computations are, just like classical, the quantum gates. As we have already seen, the quantum states can be represented as column vectors whose norm equals to 1. The quantum gates are essentially operators or matrices with the property that they are *unitary*. An operator U is unitary, if

$$UU^\dagger = \mathbf{I}$$

where $U^\dagger$ is the conjugate transpose of U.
Let's note some important properties. First, unitary operators preserve the inner product and the length of the vectors and consequently their only ability is to rotate or mirror a vector. Second, their columns create an orthonormal basis (and so do their rows). Third, if U is unitary, so is $U^\dagger$ and therefore can be applied to a state. So, if we apply U to a state $|\psi\rangle$ and then apply $U^\dagger$ the state we take is just $|\psi\rangle$. This is an important property; it states that quantum computations, unlike their classical counterparts, are reversible. In particular, consider the truth table of the classical XOR gate; see Table 1.1. It is obvious that once the gate is applied to two bits, it is impossible to go back. We have lost information. Such gates do not exist in the quantum world. The quantum counterpart of the XOR gate is the CNOT gate; see Figure 1.3. We see that it takes as input two qubits and outputs two qubits. Suppose that we give the CNOT gate the qubits $|0\rangle|0\rangle = |00\rangle$. This state (say $|\psi\rangle$) can be also written as

$$|\psi\rangle = 1 \cdot |00\rangle + 0 \cdot |01\rangle + 0 \cdot |10\rangle + 0 \cdot |11\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

and therefore we have

$$\text{CNOT}|00\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = |00\rangle$$

Accordingly, we also have $\text{CNOT}|01\rangle = |01\rangle$, $\text{CNOT}|10\rangle = |11\rangle$, $\text{CNOT}|11\rangle = |10\rangle$. So, we have that

$$|a, b\rangle \xrightarrow{\text{CNOT}} |a, a \otimes b\rangle$$

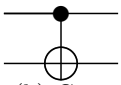
We see that in order to compute the XOR of two qubits, we also have to keep the first qubit. Note, also, that the CNOT gate is reversible. We just apply it once more and we return back to the first state.

bit 1	bit 2	result
0	0	0
0	1	1
1	0	1
1	1	0

Table 1.1: Truth table of XOR gate

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

(a) Matrix



(b) Gate

Figure 1.3: The CNOT Operator

A crucial remark is that we can apply the gate to a superposition of states; and by linearity we can have:

$$\begin{aligned} \text{CNOT}(a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle) = \\ a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|11\rangle + a_{11}|10\rangle \end{aligned}$$

This way we compute the XOR of all possible inputs at once.
Another remark is in the way we compute functions. Suppose we have a classical function $f : \{0,1\}^n \to \{0,1\}^m$. The way it is calculated quantumly is by giving as input to the circuit U_f daggerh the input qubits and some more qubits where the output will be written; see Figure 1.4. So just like

$$|a, b\rangle \xrightarrow{\text{CNOT}} |a, a \otimes b\rangle$$

we have

$$|x, b\rangle \xrightarrow{U_f} |x, f(x) \otimes b\rangle$$

and by initializing $|b\rangle$ to $|0^m\rangle$ we have

$$|x, b\rangle \xrightarrow{U_f} |x, f(x)\rangle$$

Observe that quantumly we can compute all the values of the function f by simply creating a superposition of all the possible inputs and then apply U_f. Then we will get a superposition of all the possible outputs of f.

One of the most useful gates in quantum computing is the *Hadamard* gate. The Hadamard

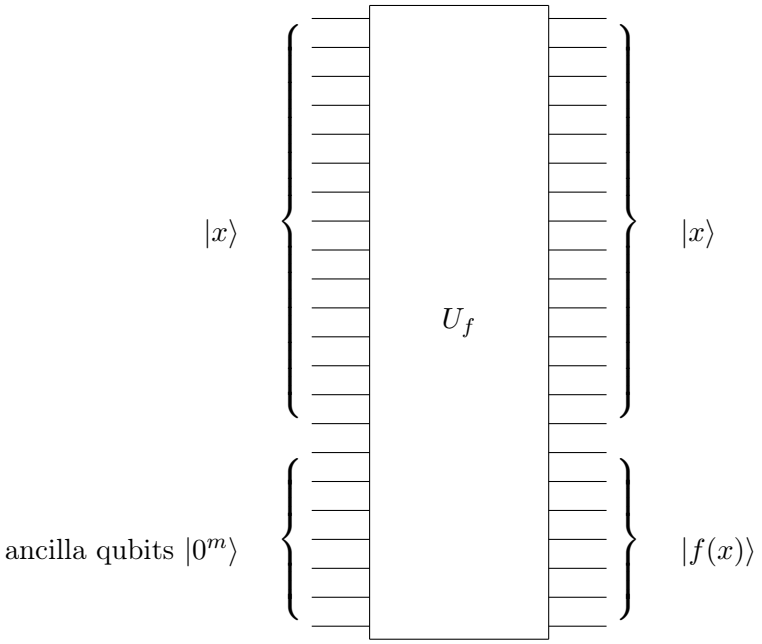


Figure 1.4: Quantum circuit U_f for computing f

operator has the form

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

and so we have $H|0\rangle = |+\rangle, H|1\rangle = |-\rangle, H|+\rangle = |0\rangle, H|-\rangle = |1\rangle$. In other words $HH = \mathbf{I}$. When applied to a superposition it gives

$$H(a_0|0\rangle + a_1|1\rangle) = \frac{a_0 + a_1}{\sqrt{2}}|0\rangle + \frac{a_0 - a_1}{\sqrt{2}}|1\rangle$$

Now, suppose that we have n qubits and the state of each qubit is $|0\rangle$. Then the whole state can be written as $|0^n\rangle$. We can apply the Hadamard gate to each of these qubits (applying in parallel n Hadamard gates can be written using the tensor product as $H^{\otimes n}$). This gives

$$H^{\otimes n}|0^n\rangle = \sum_{i \in \{0,1\}^n} \frac{1}{\sqrt{2^n}}|i\rangle = \frac{1}{\sqrt{2^n}} \sum_{i \in \{0,1\}^n} |i\rangle$$

This is an easy way of creating a superposition of all the different values. Giving this state to the previous circuit U_f together with the ancilla qubits initialized to the state $|0^m\rangle$ will output

$$U_f \left(\frac{1}{\sqrt{2^n}} \sum_{i \in \{0,1\}^n} |i\rangle \otimes |0^m\rangle \right) = \frac{1}{\sqrt{2^n}} \sum_{i \in \{0,1\}^n} |i\rangle \otimes |f(i)\rangle$$

However, as mentioned before, we do not have access to all this information. In particular, a measurement of this state will result one random value from the range of f.

1.1.4 Entanglement

Consider two qubits, one in the state $|\psi\rangle = a_0|0\rangle + a_1|1\rangle$ and the other in the state $|\phi\rangle = b_0|0\rangle + b_1|1\rangle$. Then the state of the two qubits can be written as

$$|\psi\phi\rangle = a_0b_0|00\rangle + a_0b_1|01\rangle + a_1b_0|10\rangle + a_1b_1|11\rangle$$

by taking the Cartesian product. Now, consider one of the famous Bell States which is also a state of two qubits

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

By simple calculations we can prove that it is impossible to find two qubits of the form $|\psi\rangle = a_0|0\rangle + a_1|1\rangle$ and $|\phi\rangle = b_0|0\rangle + b_1|1\rangle$ that can result in this state. We say that the two qubits of $|\Phi^+\rangle$ are *entangled*, whereas if they are not entangled we say that they are *separable*.

There is a very interesting property of the entangled qubits. Suppose that Alice and Bob share the state $|\Phi^+\rangle$, Alice possesses the first of the two qubits and Bob the second. Suppose also that they are very far apart. Sometime, Alice decides to measure her qubit. Then with probability $1/2$ she will measure $|0\rangle$ and with probability $1/2$ she will measure $|1\rangle$. Immediately after her measurement if Bob decides to measure his qubit, he will measure the same bit as Alice. In other words, if Alice measured $|0\rangle$ the state collapses to $|00\rangle$ and it is completely determined and therefore Bob's measurement will also give $|0\rangle$.

1.1.5 No Cloning Theorem

We now give an important theorem limiting the power of unitary operators.

Theorem 1.1. *Suppose we have an unknown state $|\psi\rangle$. Then there is no unitary operator that can copy it.*

Proof. Suppose that there is a unitary U that can copy quantum states; $U(|\psi\rangle|0\rangle) = |\psi\rangle|\psi\rangle$ and $U(|\phi\rangle|0\rangle) = |\phi\rangle|\phi\rangle$. The inner product of the initial states is

$$(\langle\psi| \otimes \langle 0|)(|\phi\rangle \otimes |0\rangle) = \langle\psi|\phi\rangle\langle 0|0\rangle = \langle\psi|\phi\rangle$$

whereas the inner product of the final states is

$$(\langle\psi| \otimes \langle\psi|)(|\phi\rangle \otimes |\phi\rangle) = \langle\psi|\phi\rangle\langle\psi|\phi\rangle = (\langle\psi|\phi\rangle)^2$$

We observe that

$$\langle\psi|\phi\rangle \neq (\langle\psi|\phi\rangle)^2$$

unless $|\psi\rangle = |\phi\rangle \Leftrightarrow \langle\psi|\phi\rangle = 1$ or $\langle\psi|\phi\rangle = 0$ in which case they are not unknown since we can measure them without destroying them. Therefore we come to a contradiction. □

1.2 Quantum Information

We will now present the basic quantum information principles. Let's begin with a simple example. Suppose Alice and Bob share the *bipartite* entangled state

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}|0\rangle_A|0\rangle_B + \frac{1}{\sqrt{2}}|1\rangle_A|1\rangle_B$$

As we have already mentioned this state is not separable and therefore the state of Alice's or Bob's qubit separately cannot be expressed with the previous way. We need a more general definition for this kind of states. We will introduce the *mixed* states. Note that the states we have presented in the previous section are called *pure* states.

1.2.1 Mixed States

Consider the following: if Alice measures her qubit then, as we have already mentioned, she will output $|0\rangle$ or $|1\rangle$ with probability 1/2 each. Therefore, we can say that Alice possesses a probability distribution of two pure states; namely she possesses the mixed state $\{(1/2, |0\rangle), (1/2, |1\rangle)\}$. In general, a mixed state can be any probability distribution over pure states.
The *density matrix* or *density operator* is a good way to work with quantum mixed states. Suppose that we have the mixed state $\{p_i, |\psi_i\rangle\}$. We define the density matrix of a state as:

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$$

For example, the density matrix of the previous state is

$$\rho = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1| = \frac{1}{2}\cdot\begin{pmatrix}1\\0\end{pmatrix}\begin{pmatrix}1 & 0\end{pmatrix} + \frac{1}{2}\cdot\begin{pmatrix}0\\1\end{pmatrix}\begin{pmatrix}0 & 1\end{pmatrix} = \frac{1}{2}\cdot\begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}$$

An important fact about mixed states is that different mixed states can have the same density matrix. For example, the density matrix of the mixed state $\{(1/2, |+\rangle), (1/2, |-\rangle)\}$

is also

$$\rho = \frac{1}{2} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Let's note some important properties.

1. The trace of a density matrix equals to 1; $\operatorname{Tr} \rho = 1$
2. The trace of the square of a density matrix is less or equal to 1; $\operatorname{Tr} \rho^2 \leq 1$. In particular, if ρ is a density matrix of a pure state then $\operatorname{Tr} \rho^2 = 1$, and if it is a density matrix of a mixed state $\operatorname{Tr} \rho^2 < 1$
3. ρ is Hermitian, meaning $\rho = \rho^\dagger$, it has non-negative eigenvalues and is a non-negative operator, meaning that for any state $|\psi\rangle$ it holds that $\langle\psi|\rho|\psi\rangle \geq 0$.

The measurement of a mixed state can also be made in any basis. So if we measure the state $\rho = \sum_{i=1}^{n} p_i |\psi_i\rangle\langle\psi_i|$ in the basis $\{|b_1\rangle, |b_2\rangle, \ldots, |b_m\rangle\}$ then the probabilities of the outcome o are:

$$\Pr[o = |b_k\rangle] = \sum_{i=1}^{n} p_i |\langle b_k|\psi_i\rangle|^2 = \sum_{i=1}^{n} p_i \langle b_k|\psi_i\rangle\langle\psi_i|b_k\rangle = \langle b_k| \left(\sum_{i=1}^{n} p_i |\psi_i\rangle\langle\psi_i| \right) |b_k\rangle$$

and therefore

$$\Pr[o = |b_k\rangle] = \langle b_k|\rho|b_k\rangle$$

Also, the evolution of a mixed state is accomplished using unitary operators. If U is a unitary operator then the new density operator is

$$\rho' = \sum_i p_i U|\psi_i\rangle\langle\psi_i|U^\dagger = U \left(\sum_i p_i |\psi_i\rangle\langle\psi_i| \right) U^\dagger$$

and therefore

$$\rho' = U\rho U^\dagger$$

1.2.2 Partial Trace and Purification

In the previous subsection we showed an informal way to define the state of Alice's qubit by calculating the probabilities of the measurement outcomes. The formal procedure is called *Partial Trace* and we say that we *trace out* Bob's state. Also the new density operator is called *reduced* density operator.

Suppose the general scenario where Alice and Bob share the state

$$|\psi\rangle = \sum_{ij} c_{ij}|i\rangle_A|j\rangle_B$$

Then, we can consider the density matrix ρ of this pure state and we have $\rho = |\psi\rangle\langle\psi|$. We define the reduced density matrix of Alice as

$$\rho_A = \mathrm{Tr}_B\,\rho = \sum_j (\langle j|_B \otimes \mathbf{I}_A)\rho(\mathbf{I}_A \otimes |j\rangle_B)$$

and similarly for the state of Bob

$$\rho_B = \mathrm{Tr}_A\,\rho = \sum_i (\langle i|_A \otimes \mathbf{I}_B)\rho(\mathbf{I}_B \otimes |i\rangle_A)$$

where $\{|j\rangle_B\}, \{|i\rangle_A\}$ are the basis vectors of Alice's space and Bob's space respectively. So in the previous example

$$|0\rangle_B = \begin{pmatrix}1\\0\end{pmatrix},\ |1\rangle_B = \begin{pmatrix}0\\1\end{pmatrix} \text{ and } \mathbf{I}_A = \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}$$

and therefore

$$\mathbf{I}_A \otimes |0\rangle_B = \begin{pmatrix}1 & 0\\0 & 0\\0 & 1\\0 & 0\end{pmatrix} \text{ and } \mathbf{I}_A \otimes |1\rangle_B = \begin{pmatrix}0 & 0\\1 & 0\\0 & 0\\0 & 1\end{pmatrix}$$

and with simple calculations we have the previous result

$$\rho_A = \begin{pmatrix}\frac{1}{2} & 0\\0 & 0\end{pmatrix} + \begin{pmatrix}0 & 0\\0 & \frac{1}{2}\end{pmatrix} = \begin{pmatrix}\frac{1}{2} & 0\\0 & \frac{1}{2}\end{pmatrix}$$

An important remark is that the reduced density matrix of Bob is independent from the measurement basis of Alice. Furthermore, if the whole state is separable, then its density matrix ρ is simply the tensor product of ρ_A and ρ_B; $\rho = \rho_A \otimes \rho_B$.
Taking it one step further, the state Alice and Bob share is not necessarily a pure state; it can be a mixed state as well.

We will now see the opposite direction. Suppose that Alice possesses the mixed state ρ_A. Then we define a *purification* of ρ_A any bipartite pure state shared between Alice and Bob where

$$\mathrm{Tr}_B\,|\psi\rangle\langle\psi| = \rho_A$$

For example, one purification of $\rho_A = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}$ is the state $\frac{1}{\sqrt{2}}|0\rangle_A|0\rangle_B + \frac{1}{\sqrt{2}}|1\rangle_A|1\rangle_B$. We note that the purification of a mixed state is not unique; there can be many different pure states that if they be partially traced will give ρ_A.

1.2.3 Superoperators

Now suppose that we have a composite bipartite system on the Hilbert Space $H_A \otimes H_B$ with density matrix ρ and Alice possesses the reduced state ρ_A such that $\rho_A = \mathrm{Tr}_B\, \rho$. We apply a unitary transformation to ρ and the new state is

$$\rho' = U\rho U^\dagger$$

Now, if we trace out Bob's space we take Alice's new density matrix

$$\rho'_A = \mathrm{Tr}_B\, \rho'$$

We define the *superoperator* S as

$$\rho'_A = S(\rho_A)$$

If H_A is a Hilbert space, we define $\mathbf{L}(H_A)$ as the set of all linear operators on H_A (have in mind the set of all density operators on H_A). A superoperator essentially is a linear map from one Hilbert Space to another $H_A \to H'_A$. Some examples of superoperators are the unitary operator on a density matrix ($\mathbf{L}(H_A) \to \mathbf{L}(H_A)$), a partial trace of a density matrix ($(\mathbf{L}(H_A) \otimes \mathbf{L}(H_B)) \to \mathbf{L}(H_A)$) or an operator that adds more qubits to a state ($\mathbf{L}(H_A) \to \mathbf{L}(H_A \otimes H_B)$); namely $S(\rho) = \rho \otimes |0\rangle\langle 0|$. Furthermore, a superoperator $S : \mathbf{L}(H_A) \to \mathbf{L}(H'_A)$ can be extended to a superoperator S' that is applied to a larger system by taking the tensor product between this and the identity operator; $S' = S \otimes \mathbf{I} : \mathbf{L}(H_A \otimes F) \to \mathbf{L}(H'_A \otimes F)$.

Superoperators can also be written as

$$S(\rho_A) = \mathrm{Tr}_B \left[U(\rho_A \otimes |0\rangle\langle 0|)U^\dagger \right]$$

for some $U, |0\rangle$ and have the following properties:

1. They map Hermitian matrices to Hermitian matrices
2. They map positive matrices to positive matrices
3. They preserve the trace $\mathrm{Tr}\, \rho_A = \mathrm{Tr}\, (S(\rho_A))$
4. They are completely positive operators meaning that for every positive density matrix ρ and for every Hilbert space M, it holds that $(S \otimes \mathbf{I}_M)(\rho)$ is also positive,

where $\mathbf{I}_M$ is the identity operator on M.

Properties 2,3 show that if S is applied to a density matrix, the output is also a density matrix.

1.3 Distance Measures

In this section we will define the necessary distance measures for our work. Then we will use them to define different kinds of indistinguishability which is a key property for the security of cryptographic protocols.

1.3.1 Fidelity

The *fidelity* of two pure states $|\psi\rangle, |\phi\rangle$ is simply defined as

$$\mathrm{F}(|\psi\rangle, |\phi\rangle) = |\langle\psi|\phi\rangle|^2$$

So, if we measure the qubit $|\psi\rangle$ in the basis $\{|0\rangle, |1\rangle\}$ we will take $|0\rangle$ with probability $\mathrm{F}(|\psi\rangle, |0\rangle)$ and $|1\rangle$ with probability $\mathrm{F}(|\psi\rangle, |1\rangle)$.
The fidelity between a density matrix and a pure state is defined similarly

$$\mathrm{F}(\rho, |\psi\rangle) = \langle\psi|\rho|\psi\rangle$$

where in the case that $\rho = |\psi\rangle\langle\psi|$ we take the previous expression.
Finally, an easy way to define the fidelity of two mixed states is

$$\mathrm{F}(\rho, \sigma) = \max |\langle\psi|\phi\rangle|^2$$

where the maximum is taken over all purifications $|\psi\rangle$ of ρ and all purifications $|\phi\rangle$ of σ.

1.3.2 Statistical Distance - Trace Distance

An important measure in probability theory is the *statistical distance* between two distributions. If X and Y are two random variables following the probability distributions D_1 and D_2 respectively then the statistical distance between D_1 and D_2 is defined as

$$\Delta(D_1, D_2) = \frac{1}{2} \sum_{i \in \Omega} | \Pr_{X \leftarrow D_1}[X = i] - \Pr_{Y \leftarrow D_2}[Y = i]|$$

where Ω is the set of all the possible values X and Y can take.

The quantum analogue of the statistical distance is the *trace distance* between two density

matrices. There are many ways to define it, the simplest being

$$\mathrm{D}(\rho,\sigma) = \frac{1}{2}\|\rho-\sigma\|_1 = \frac{1}{2}\sum_i |\lambda_i|$$

where $\|\cdot\|_1$ is the trace norm and λ_i are the eigenvalues of the matrix $\rho-\sigma$.

1.3.3 Diamond norm

The *diamond distance* [3] between two superoperators $S_1, S_2 : \mathbf{L}(N) \to \mathbf{L}(M)$ is defined as

$$\Diamond(S_1,S_2) = \max\Big\{ \mathrm{D}\left((S_1\otimes \mathbf{I}_H)(\rho), (S_2\otimes \mathbf{I}_H)(\rho)\right)\Big\}$$

where the maximum is taken over all finite dimensional Hilbert spaces H and all $\rho \in \mathbf{L}(H\otimes N)$. In other words, we take N and we increase it with H. Then we apply $S_1\otimes \mathbf{I}_H$ to all possible states $\rho\in\mathbf{L}(H\otimes N)$ and take the maximum trace norm for all ρ and H.

1.3.4 Quantum Indistinguishability

Here, we will analyze the different notions of indistinguishability. There are many different aspects such as perfect, statistical, computational indistinguishability of classical probability distributions or quantum states. There is also indistinguishability of quantum algorithms.

We will denote a function f negligible on some security parameter n if for all polynomials p there exists some n_0 such that for all $n > n_0$ it holds that

$$f(n) < \frac{1}{p(n)}$$

Sometimes we will write $\mathrm{negl}(n)$ to denote any negligible function on n.

Perfect Indistinguishability

We begin with a very classical result from probability theory. If two distributions D_1, D_2 have zero statistical distance then this means that the two distributions are the same and therefore no algorithm can distinguish between the two random variables X_1, X_2 following D_1, D_2 respectively. We say that X_1 and X_2 are perfectly indistinguishable.

The quantum analogue is very similar. If two mixed states $\{p_i, |\psi_i\rangle\}, \{p'_i, |\psi'_i\rangle\}$ have same density matrices then they are perfectly indistinguishable; no quantum algorithm can distinguish between the two states.

Measure of similarity - Fidelity

Let's analyze for a while the use of fidelity. Fidelity, doesn't measure indistinguishability; it measures how indistinguishable two states can become. Suppose that Alice sends to Bob some qubits that are entangled with some qubits of Alice and the whole state is $\rho = |\psi\rangle\langle\psi|$. As we have already said, Bob's state is $\rho_B = \mathrm{Tr}_A\, \rho$. Now, Alice wants to change the whole state ρ to some other $\sigma = |\phi\rangle\langle\phi|$ but she doesn't possess the whole state; she possesses only one part of the qubits and therefore she is allowed to apply operations only to those qubits. We define similarly $\sigma_B = \mathrm{Tr}_A\, \sigma$. How close can she bring ρ and σ? The answer depends on $\mathrm{F}(\rho_B, \sigma_B)$. In particular, if she sends her remaining qubits to Bob then the best probability that Bob will accept that the whole state is σ (and not ρ) is $\mathrm{F}(\rho_B, \sigma_B)$. This is completely related to the definition of fidelity; at worst, Alice will create the purification of ρ_B that has the maximum inner product with $|\phi\rangle$.
For example she can transform the state $|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle_A|0\rangle_B + \frac{1}{\sqrt{2}}|1\rangle_A|1\rangle_B$ to $|\phi\rangle = \frac{1}{\sqrt{2}}|1\rangle_A|0\rangle_B + \frac{1}{\sqrt{2}}|0\rangle_A|1\rangle_B$ by applying the NOT gate to her qubit. In other words, the fidelity of the reduced matrices is equal to 1 so the probability that Bob will accept that the state is $|\phi\rangle$ (despite that initially it was $|\psi\rangle$) is 1.

Indistinguishability of variables - states

A measure of indistinguishability between two random variables uses the statistical distance.

Proposition 1.1. *Suppose two random variables X_1, X_2. Then for all classical algorithms V it holds that:*

$$\left| \Pr_{x \leftarrow X_1}[V(x) = 1] - \Pr_{x \leftarrow X_2}[V(x) = 1] \right| \leq \Delta(X_1, X_2)$$

Definition 1.1 (Statistical Indistinguishability)**.** *Suppose two random variables X_1, X_2. If $\Delta(X_1, X_2) = \mathrm{negl}(n)$ (for some security parameter n) then X_1, X_2 are statistically indistinguishable.*

The previous definition gives only one aspect of indistinguishability. Another aspect is the computational indistinguishability with a weaker definition. Two random variables can be computationally indistinguishable even if their statistical distance is non-negligible.

Definition 1.2 (Computational Indistinguishability)**.** *Suppose two random variables X_1, X_2. If*

$$\left| \Pr_{x \leftarrow X_1}[V(x) = 1] - \Pr_{x \leftarrow X_2}[V(x) = 1] \right| = \mathrm{negl}(n)$$

for all classical polynomial time algorithms V (for some security parameter n) then X_1, X_2 are computationally indistinguishable.

One example of computational indistinguishability is the DDH assumption; the random variables $\langle g^x, g^y, g^{xy}\rangle$ and $\langle g^x, g^y, g^z\rangle$ in a cyclic group G with generator g have great statistical distance but it is conjectured that they are computationally indistinguishable.

The previous definitions can be extended to the quantum world by replacing the statistical distance with the trace distance.

Proposition 1.2. *Suppose two mixed states ρ_1, ρ_2. Then for all quantum algorithms Q it holds that:*

$$\left|\langle 1|Q(\rho_1)|1\rangle - \langle 1|Q(\rho_2)|1\rangle\right| \leq \mathrm{D}(\rho_1, \rho_2)$$

Definition 1.3 (Weak Quantum Statistical Indistinguishability)**.** *Suppose two mixed states ρ_1, ρ_2. If $\mathrm{D}(\rho_1, \rho_2) = \mathrm{negl}(n)$ then ρ_1, ρ_2 are weakly quantum statistically indistinguishable.*

Definition 1.4 (Weak Quantum Computational Indistinguishability)**.** *Suppose two mixed states ρ_1, ρ_2. If*

$$\left|\langle 1|Q(\rho_1)|1\rangle - \langle 1|Q(\rho_2)|1\rangle\right| = \mathrm{negl}(n)$$

for all quantum polynomial time algorithms Q then ρ_1, ρ_2 are weakly quantum computationally indistinguishable.

The previous definition is a weak definition of quantum indistinguishability. The strong definition gives to the distinguisher Q some extra state that may help it distinguish the states.

Definition 1.5 (Strong Quantum Computational Indistinguishability [20])**.** *Suppose two mixed states ρ_1, ρ_2. The two states are strongly quantum computationally indistinguishable if for all quantum polynomial time algorithms Q and for all auxiliary states σ it holds that:*

$$\left|\langle 1|Q(\rho_1 \otimes \sigma)|1\rangle - \langle 1|Q(\rho_2 \otimes \sigma)|1\rangle\right| = \mathrm{negl}(n)$$

for security parameter n.

Indistinguishability of Quantum Operations

Next, we extend the previous notion to the case of superoperators. As before we can have weak and strong indistinguishability.

Definition 1.6 (Weak Computational Indistinguishability of Superoperators)**.** *Suppose two superoperators $S_1, S_2 : \mathbf{L}(N) \to \mathbf{L}(M)$. S_1, S_2 are weakly quantum computationally indistinguishable if for all quantum polynomial time algorithms Q and for all states*

$\rho \in \mathbf{L}(N)$ it holds that

$$\left|\langle 1|Q(S_1(\rho))|1\rangle - \langle 1|Q(S_2(\rho))|1\rangle\right| = \mathrm{negl}(n)$$

for security parameter n.

Allowing as before some auxiliary space gives the strong indistinguishability of superoperators.

Definition 1.7 (Strong Computational Indistinguishability of Superoperators)**.** *Suppose two superoperators $S_1, S_2 : \mathbf{L}(N) \to \mathbf{L}(M)$. S_1, S_2 are strongly quantum computationally indistinguishable if for all quantum polynomial time algorithms Q for all spaces H and for all states $\rho \in \mathbf{L}(N \otimes H)$ it holds that*

$$\left|\langle 1|Q(S_1 \otimes \mathbf{I}_H(\rho))|1\rangle - \langle 1|Q(S_2 \otimes \mathbf{I}_H(\rho))|1\rangle\right| = \mathrm{negl}(n)$$

for security parameter n.

The analogue of Proposition 1.2 is the following:

Proposition 1.3. *Suppose two superoperators $S_1, S_2 : \mathbf{L}(N) \to \mathbf{L}(M)$. Then for all quantum algorithms Q it holds that:*

$$\left|\langle 1|Q(S_1 \otimes \mathbf{I}_H(\rho))|1\rangle - \langle 1|Q(S_2 \otimes \mathbf{I}_H(\rho))|1\rangle\right| \leq \Diamond(\rho_1, \rho_2)$$

and therefore using the diamond distance we can define the strong statistical indistinguishability between superoperators.

Definition 1.8 (Strong Statistical Indistinguishability of Superoperators)**.** *Suppose two superoperators $S_1, S_2 : \mathbf{L}(N) \to \mathbf{L}(M)$. S_1, S_2 are strongly quantum statistically indistinguishable if*

$$\Diamond(\rho_1, \rho_2) = \mathrm{negl}(n)$$

for security parameter n.

2 Quantum Tools

In this chapter we will see the quantum tools that are needed for the construction of our protocols. In particular we will define Zero-Knowledge Proofs, Proofs of Knowledge, some useful cryptographic primitives such as Bit Commitment and Coin Flipping and finally private and public key Quantum Money.

2.1 Zero-Knowledge Proofs

For a thorough introduction to Zero-Knowledge Proofs see [10]. Informally, a Zero-Knowledge Proof is a protocol between a Prover and a Verifier ($\mathcal{P}$ and $\mathcal{V}$), where $\mathcal{P}$ proves to $\mathcal{V}$ only the validity of a statement and nothing more; at the end of the protocol $\mathcal{V}$ is convinced that the statement is true but he knows nothing more than this. Essentially, the information he gets is exactly one bit.

2.1.1 Classical Zero-Knowledge Proofs

Consider an NP problem such as SAT; we are given a boolean formula f and have to decide if there exists some assignment to the variables that makes the formula satisfiable. The problem is in NP because if we are given a satisfying assignment we can easily verify that it indeed satisfies it. Now consider that $\mathcal{P}$ knows the satisfying assignment and wants to prove to $\mathcal{V}$ that the formula is satisfiable without revealing the assignment. Even more, we want $\mathcal{V}$ to learn nothing more than this. We call this a Zero-Knowledge Proof.
How can we define that $\mathcal{V}$ will learn nothing more than the validity of the statement? An informal definition is the following; whatever $\mathcal{V}$ can compute after his interaction with $\mathcal{P}$, can also be computed by $\mathcal{V}$ without this interaction. Posed in another way, what $\mathcal{V}$ can compute using his data and the interaction can also be computed using only the data.
Before giving the formal definition we note that every NP problem L can be expressed by a relation R_L. If an instance x is in L then there is a *witness* or *certificate* w such that $(x, w) \in R_L$. So $\mathcal{P}$ knows the witness w and wants to prove to $\mathcal{V}$ that $x \in L$.

Definition 2.1 (Classical Statistical (Computational) Zero-Knowledge Proofs)**.** *Let L be a language in* NP *and x an instance. Suppose also an interactive protocol between $\mathcal{P}$ and $\mathcal{V}$ denoted by $\langle \mathcal{P}, \mathcal{V} \rangle$. At the end of the interaction $\mathcal{V}$ outputs $\mathrm{out}_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, s)$ where x is the input for $\mathcal{P}$ and (x, s) is the input for $\mathcal{V}$. We say that $\langle \mathcal{P}, \mathcal{V} \rangle$ is a statistical (computational) Zero-Knowledge Proof if it has the following three properties:*

- *Completeness; If both $\mathcal{P}$ and $\mathcal{V}$ are honest and $x \in L$ then*

$$\Pr[\mathrm{out}_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, s) = \mathit{accept}] = 1$$

- *Soundness; If $x \notin L$ then for every $\mathcal{P}^*$*

$$\Pr[\mathrm{out}_{\mathcal{V}}^{\langle \mathcal{P}^*, \mathcal{V} \rangle}(x, s) = \mathit{accept}] = \mathrm{negl}(n)$$

- *Zero-Knowledge; For every classical polynomial time algorithm $\mathcal{V}^*$ there exists a classical polynomial algorithm S (the simulator) such that $\mathrm{out}_{\mathcal{V}^*}^{\langle \mathcal{P}, \mathcal{V}^* \rangle}(x, s)$ and $S(x, s)$ are statistically (computationally) indistinguishable.*

for a security parameter n.

The Soundness property guarantees that a dishonest Prover cannot convince the Verifier. The Zero-Knowledge property guarantees that a dishonest Verifier cannot gain anything from the interaction.

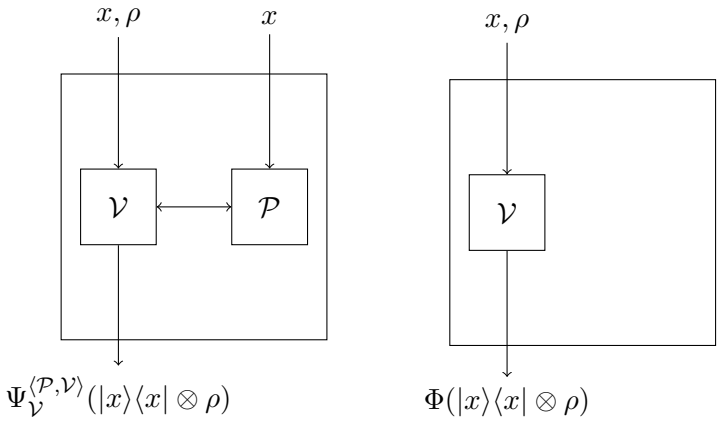
There exist Classical Zero-Knowledge Proofs for every problem in NP. More specifically

Theorem 2.1 (Classical Zero-Knowledge Proofs for every NP problem [11])**.** *Under the assumption that there exist classically secure one way functions, all languages in* NP *have Classical Computational Zero-Knowledge Proofs.*

2.1.2 Quantum Zero-Knowledge Proofs

In the quantum world things are somewhat the same. Here $\mathcal{P}$ and $\mathcal{V}$ are Quantum Polynomial time algorithms that interact classically as before. We can consider weak and strong Zero-knowledge Proofs; we will give the definitions for both but for our purpose it is enough to consider only weak Zero-Knowledge Proofs.

Definition 2.2 (Quantum Statistical (Computational) Weak (Strong) Zero-Knowledge Proofs)**.** *Let L be a language in* NP *and x an instance. Suppose also an interactive protocol between $\mathcal{P}$ and $\mathcal{V}$ denoted by $\langle \mathcal{P}, \mathcal{V} \rangle$. At the end of the interaction $\mathcal{V}$ outputs $\mathrm{out}_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, \rho) = \Psi_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(|x\rangle\langle x| \otimes \rho)$ where x is the input for $\mathcal{P}$ and (x, ρ) is the input for $\mathcal{V}$ (ρ is in general a mixed state and $\Psi_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}$ is a superoperator). We say that $\langle \mathcal{P}, \mathcal{V} \rangle$ is*



(a) Zero-Knowledge Protocol (b) Verifier Simulator

Figure 2.1: Quantum Zero-Knowledge Proofs; we want the superoperators $\Psi_{\mathcal{V}^*}^{\langle \mathcal{P}, \mathcal{V}^* \rangle}$ and Φ to be indistinguishable.

a quantum statistical (computational) weak (strong) Zero-Knowledge Proof if it has the following three properties:

- *Completeness; If both $\mathcal{P}$ and $\mathcal{V}$ are honest and $x \in L$ then*

$$\left| \langle 1 | \operatorname{out}_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, \rho) | 1 \rangle \right| = 1$$

- *Soundness; If $x \notin L$ then for every $\mathcal{P}^*$*

$$\left| \langle 1 | \operatorname{out}_{\mathcal{V}}^{\langle \mathcal{P}^*, \mathcal{V} \rangle}(x, \rho) | 1 \rangle \right| = \operatorname{negl}(n)$$

- *Zero-Knowledge; For every quantum polynomial time $\mathcal{V}^*$ there exists a quantum polynomial time superoperator Φ (the simulator) such that $\Psi_{\mathcal{V}^*}^{\langle \mathcal{P}, \mathcal{V}^* \rangle}$ and Φ are statistically (computationally) weakly (strongly) indistinguishable.*

for a security parameter n.

There exist Quantum Zero-Knowledge Proofs for every problem in NP. More specifically

Theorem 2.2 (Quantum Zero-Knowledge Proofs for every NP problem [20, 2])**.** *Under the assumption that there exist quantum secure one way functions, all languages in* NP *have Quantum Computational Strong Zero-Knowledge Proofs.*

2.1.3 Classical Zero-Knowledge Proofs of Knowledge

Next, we extend the previous definitions to the case of Zero-Knowledge Proofs of Knowledge. In a ZK Proof of Knowledge $\mathcal{P}$ wants to prove to $\mathcal{V}$ that he knows (he possesses) some secret value and not that some instance belongs to a language. So $\mathcal{P}$ does not want to prove that there exists some secret witness (or certificate) for this instance. For example, $\mathcal{P}$ may want to prove that he knows the discrete logarithm of some element y in a cyclic group. In this case, $\mathcal{P}$ doesn't want to prove that y belongs to some language (for example the language of the elements that have discrete logarithm; the discrete logarithm always exists even if nobody has computed it).
But how can we define the knowledge of some secret? Of course, looking at the code of $\mathcal{P}$ and searching for the memory address to find the discrete logarithm is not the best way to define it. One way would be to say that whatever $\mathcal{P}$ can compute using its input can also be computed using its input and the discrete logarithm and then show that these two outputs are indistinguishable. This looks like the definition of Zero-Knowledge. Another way, which is the one we will use, is the following. We can run the code of $\mathcal{P}$ in any way we want (repeat it, create breakpoints and rewind to a previous breakpoint) and extract from it the discrete logarithm.
If we show that a protocol has this property (the *validity* property) then we can assure the verifier that if he is convinced of $\mathcal{P}$'s statement 'I know the discrete logarithm', then $\mathcal{P}$ indeed knows it, or if he doesn't he can easily learn it.

Definition 2.3 (Classical ZK Proofs of Knowledge)**.** *Suppose a language L in* NP *and R the NP-relation for L. Let $\langle \mathcal{P}, \mathcal{V} \rangle$ be a pair of interactive classical algorithms. $\mathcal{P}$ has input (x, w) and $\mathcal{V}$ has input (x, z). Let $\text{out}_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, w, z)$ be the ouput of $\mathcal{V}$ after the interaction. We say that $\langle \mathcal{P}, \mathcal{V} \rangle$ is a classical statistical (computational) Zero-Knowledge Proof of Knowledge if it has the following three properties:*

- *Completeness; If $\mathcal{P}$ and $\mathcal{V}$ are honest and $x \in L$ and $(x, w) \in R$ for some witness w then for all z*

  $$\Pr[\text{out}_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, w, z) = 1] = 1$$

- *Validity; For all classical polynomial time algorithms $\mathcal{P}^*$ there exists a PPT K (the knowledge extractor) such that if*

  $$\Pr[\text{out}_{\mathcal{V}}^{\langle \mathcal{P}^*, \mathcal{V} \rangle}(x, w, z) = 1] = \text{non-negl}$$

  then also

  $$\Pr[K(x, w) = w' : R(x, w') = 1] = \text{non-negl}$$

- *Zero-Knowledge; For all PPT algorithms $\mathcal{V}^*$ there exists a PPT S (the simulator)*

such that for all x, w with $(x, w) \in R$ and for all z, the random variables $S(x, z)$ and $\text{out}_{\mathcal{V}^*}^{\langle \mathcal{P}, \mathcal{V}^* \rangle}(x, w, z)$ are statistically (computationally) indistinguishable.

Note that the validity property is a stronger condition than soundness. If an instance doesn't belong to a language then

$$\Pr[K(x, w, z) = w' : R(x, w') = 1] = \text{negl}$$

and in fact zero, so it implies the soundness property.

Theorem 2.3. *Under the assumption that there exist secure commitment schemes there exist classically secure Zero-Knowledge Proofs of Knowledge.*

2.1.4 Quantum Zero-Knowledge Proofs of Knowledge

We now give the quantum analogue of ZK PoK.

Definition 2.4 (Quantum Zero-Knowledge Proofs of Knowledge)**.** *Suppose a language L in* NP *and R the NP-relation for L. Let $\langle \mathcal{P}, \mathcal{V} \rangle$ be a pair of interactive quantum algorithms. $\mathcal{P}$ has input (x, ρ) and $\mathcal{V}$ has input (x, σ) where x is a classical string and ρ, σ are quantum mixed states. At the end of the interaction $\mathcal{V}$ outputs $out_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, \rho, \sigma) = \Psi_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(|x\rangle\langle x| \otimes \sigma)$ where $\Psi_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}$ is a superoperator. We say that $\langle \mathcal{P}, \mathcal{V} \rangle$ is a quantum statistical (computational) Zero-Knowledge Proof of Knowledge if it has the following three properties:*

- *Completeness; If both $\mathcal{P}$ and $\mathcal{V}$ are honest and $x \in L$ and $(x, w) \in R$ for some witness w then for all σ*

  $$\Pr[\text{out}_{\mathcal{V}}^{\langle \mathcal{P}, \mathcal{V} \rangle}(x, |w\rangle\langle w|, \sigma) = 1] = 1$$

- *Validity; For all quantum polynomial time algorithms $\mathcal{P}^*$ and for all x, ρ, σ there exists a quantum polynomial time algorithm K (the knowledge extractor) such that if*

  $$\left| \langle 1 | \, \text{out}_{\mathcal{V}}^{\langle \mathcal{P}^*, \mathcal{V} \rangle}(x, \rho, \sigma) | 1 \rangle \right| = \text{non-negl}$$

  then also

  $$\Pr[K(x, \rho, \sigma) = w' : R(x, w') = 1] = \text{non-negl}$$

- *Zero-Knowledge; For all quantum polynomial time algorithms $\mathcal{V}^*$ there exists a quantum polynomial time $S_{\mathcal{V}^*}$ (the simulator) such that the superoperators $\Psi_{\mathcal{V}^*}^{\langle \mathcal{P}, \mathcal{V} \rangle}$ and $S_{\mathcal{V}^*}$ are statistically (computationally) indistinguishable.*

Notice the sequence of the quantifiers in the definition of the zero-knowledge property. What we say is

$$\forall\, \mathcal{V}^* \,\exists S \;s.t. \ldots.$$

We can interchange these quantifiers to give a stronger definition

$$\exists S \;\forall\, \mathcal{V}^* \;\; s.t. \ldots.$$

In fact, Unruh recently proved quantum security of ZKPoK in the model with the quantifiers swapped. This is one of the most important theorems for our construction.

Theorem 2.4 (Quantum Secure Zero-Knowledge Proofs of Knowledge [19, 20])**.** *Under the assumption that there exist quantum secure one way functions, there also exist computational Zero-Knowledge Proofs of Knowledge for any language in* NP *secure against quantum attacks.*

Note that it is not enough to prove the security of a classical protocol against quantum attacks by just proving the security of the primitives it uses. Zero-Knowledge Proofs is an example:

1. Classically secure Commitment Schemes imply Classically secure ZKPoK.
2. Quantumly secure Commitment Schemes imply Quantumly secure ZKPoK but this is not trivially implied from the previous.

The classical proofs of security include the creation of breakpoints and rewinding to a previous step if the simulator (and/or the the extractor) does not succeed. Quantumly, the measurement disturbs the system and therefore the rewinding is not a simple task.

2.1.5 Relativized Quantum Zero-Knowledge Proofs

We can guarantee security of QZKPoK relative to a permutation oracle.

Theorem 2.5 (Relativized Quantum One-Way Permutations [5])**.** *Relative to a random permutation oracle, there exists an one-way permutation secure against quantum attacks.*

What comes from the above theorem is that if we have a random permutation oracle Q, then there exists an one-way permutation that can be efficiently computed but even a quantum algorithm needs exponentially many queries to Q in order to reverse the permutation.
Now combining theorems 2.4 and 2.5 we have:

Proposition 2.1 (Relativized QZKPoK)**.** *Relative to a random permutation oracle, there exist computational Zero-Knowledge Proofs of Knowledge for any language in* NP *secure against quantum attacks.*

2.2 Cryptographic Primitives

Next, we introduce some basic Cryptographic Primitives; namely strong/weak Coin Flipping, Bit Commitment and Oblivious Transfer. We will show the connections and implications between them and their information theoretic bounds.

2.2.1 Oblivious Transfer

In an OT protocol Alice possesses one secret string s and wants to interact with Bob in such a way that Bob's probability of learning s is $1/2$ and Alice's probability of learning if Bob learnt s is also $1/2$ (she is oblivious of whether he learnt it). There are many different versions of OT. The most common one is the *1 out of 2* OT. In this version Alice possesses two secrets (they can be strings or single bits) and interacts with Bob in such a way that Bob can choose which one he wants to learn but Alice's probability of learning Bob's choice remains $1/2$. This can be shown to be equivalent with the *1 out of 2 random* OT where the secret bits of Alice and the secret choice of Bob are chosen randomly. Another version is the k *out of* n OT. Here, Bob chooses to learn k of the secrets but Alice's probability to learn Bob's choices remains $\binom{n}{k}^{-1}$. Here we will give the formal definition of the 1 out of 2 OT (OT_2^1).

Definition 2.5 (Quantum OT_2^1)**.** *A quantum* OT_2^1 *with bias* ε *is a protocol between Alice and Bob where:*

- *Alice has secret input* $s_0, s_1 \in \{0,1\}$ *and Bob has secret input* $b \in \{0,1\}$*.*
- *At the end of the protocol Alice outputs* $o^A \in \{0,1,abort\}$ *and Bob outputs* $o_1^B \in \{0,1,abort\}$ *and* $o_2^B \in \{0,1,abort\}$*.*
- *If any output gives abort then the protocol aborts.*
- *If both are honest then*
  - *They never abort*
  - $o_1^B = s_b$
  - $\Pr[o^A = b] = 1/2$
  - $\Pr[o_2^B = s_{\bar{b}}] = 1/2$
- *If Alice is dishonest then*

$$\Pr[o^A = b \text{ and } o_1^B \neq abort] \leq \frac{1}{2} + \varepsilon_A$$

- *If Bob is dishonest then*

$$\Pr[(o_1^B, o_2^B) = (s_0, s_1) \text{ and } o^A \neq \textit{abort}] \leq \frac{1}{2} + \varepsilon_B$$

- $\varepsilon = \max\{\varepsilon_A, \varepsilon_B\}$

If ε can be arbitrarily close to zero then the protocol is information theoretically secure.

Unfortunately, it has been shown that there does not exist protocol with negligible bias. In fact, Chailloux et al. [7] showed that $\varepsilon \geq 0.0586$ and proposed a protocol with $\varepsilon \leq 1/4$. It is still an open problem of what are the optimal bounds for Oblivious Transfer.

2.2.2 Bit Commitment

Suppose that Bob has a lottery and Alice wants to bet on some numbers. Today, when Alice wants to gamble in a lottery, she ticks some numbers in a piece of paper, then she sends it to Bob in order to be bound to her choices and then she trusts that Bob will randomly choose the winning numbers. Of course, this is not the best for Alice, since Bob may announce different numbers in order to prevent her from winning. The answer to this problem, can be given by using a cryptographic primitive called *Bit Commitment*. Informally, Alice commits to the numbers she wants to bet on but without Bob learning these numbers ahead of time. Then Bob randomly chooses some numbers, and Alice reveals the numbers she was committed to, having Bob sure that she can't reveal some different numbers from those she was committed to. This protocol can be proven that is equivalent to a simpler one, where Alice commits only to one bit.

Definition 2.6 (Bit Commitment)**.** *A quantum Bit Commitment with bias ε denoted by* $\mathrm{BC}(\varepsilon)$ *is a protocol between Alice and Bob consisting of two Phases;*

- *Commit Phase; where Alice interacts with Bob in order to commit to some bit b*
- *Reveal Phase; where Alice interacts with Bob in order to reveal b. If Bob accepts the revealed value then we say that Alice successfully reveals the bit b.*

A secure Bit Commitment protocol satisfies three properties:

- *Completeness; If both Alice and Bob are honest then Alice successfully reveals the committed bit b.*
- *Binding property; If Alice is cheating then*

$$\frac{1}{2}\left(\Pr[\textit{Alice successfully reveals } 0] + \Pr[\textit{Alice successfully reveals } 1]\right) \leq \frac{1}{2} + \varepsilon_A$$

- *Hiding property; If Bob is cheating then*

$$\Pr[\textit{Bob guesses } b \textit{ after the Commit Phase}] \leq \frac{1}{2} + \varepsilon_B$$

Define $\varepsilon = \max\{\varepsilon_A, \varepsilon_B\}$. *A* BC($\varepsilon$) *protocol is secure if* ε *can be arbitrarily close to zero.*

Bit Commitment is a weaker primitive that Oblivious transfer. In other words, if we have a secure OT we can create a secure BC. However, as we have already seen, there does not exist a secure OT protocol. Trying to build an imperfect BC from an imperfect OT does not preserve the security parameters. In other words if the optimal bounds for OT are, say, b this doesn't imply that the optimal bounds of BC are also b.
The optimal bias for Quantum Bit Commitment has been recently found [6] to be 0.239. A natural way to prove lower bounds for a stronger primitive is by using it to create a weaker one. Then, by using the lower bound for the weaker one we come up with a lower bound for the stronger one. Of course, this is not always trivial.

2.2.3 Coin Flipping

Consider the following scenario: Alice and Bob want to flip a coin (or play some gamble game in general) by telephone. How can they succeed this? Of course, Alice doesn't trust Bob that he will fairly flip a coin and send her the result, neither Bob trusts Alice. It has been shown very early that this can be accomplished if we make some computational assumptions. However, in the information theory setting things change. Let us first give the formal definition of Coin Flipping and then give the results.

Definition 2.7 (Strong Coin Flipping)**.** *A Strong Coin Flipping Protocol with bias* ε *denoted by* SCF(ε) *is a protocol between Alice and Bob where:*

- *At the end of the protocol Alice outputs* $o^A \in \{0, 1, abort\}$*, Bob outputs* $o^B \in \{0, 1, abort\}$ *and the protocol outputs* $o \in \{0, 1, abort\}$.
- *If* $o^A = o^B$ *then* $o = o^A$
- *If* $o^A \neq o^B$ *then* $o = abort$
- *If both are honest then* $\Pr[o = 0] = \Pr[o = 1] = 1/2$
- *If Alice is dishonest then* $\max\{\Pr[o = 0], \Pr[o = 1]\} \leq 1/2 + \varepsilon_A$
- *If Bob is dishonest then* $\max\{\Pr[o = 0], \Pr[o = 1]\} \leq 1/2 + \varepsilon_B$
- $\varepsilon = \max\{\varepsilon_A, \varepsilon_B\}$

Kitaev in '03 proved that there does not exist any SCF protocol with bias less that $1/\sqrt{2} - 1/2$. In '09 Chailloux and Kerenidis [8] proved that this is also an upper bound by giving a protocol with bias arbitrarily close to $1/\sqrt{2} - 1/2$. In their construction they used a weaker version of Coin Flipping; namely the *Weak* Coin Flipping. In a weak coin flipping, it is enough to prevent the parties only from one of the two outcomes. Informally, we can consider that $o = 0$ implies that Alice wins and $o = 1$ implies that Bob wins. Now, we don't mind if a party wants to lose, but we only mind to prevent a party from winning with probability greater than $1/2$.

Definition 2.8 (Balanced Weak Coin Flipping)**.** *A balanced weak coin flipping protocol with bias ε denoted by* $\mathrm{WCF}(\frac{1}{2}, \varepsilon)$ *is a protocol between Alice and Bob where at the end of the protocol:*

- *Alice outputs $o_A \in \{0,1\}$, Bob outputs $o_B \in \{0,1\}$ and the protocol outputs $o \in \{0,1,abort\}$*
- *If $o_A = o_B$ then $o = o_A$ otherwise $o = abort$*
- *If both parties are honest then* $\Pr[o=0] = \Pr[o=1] = \frac{1}{2}$
- *If Alice is dishonest then* $\Pr[o=0] \leq \frac{1}{2} + \varepsilon_A$
- *If Bob is dishonest then* $\Pr[o=1] \leq \frac{1}{2} + \varepsilon_B$
- $\varepsilon = \max\{\varepsilon_A, \varepsilon_B\}$

If $o = 0$ we say that Alice wins and if $o = 1$ we say that Bob wins. If ε can be arbitrarily close to zero then the protocol is secure.

It can be seen that in this last definition Alice can cheat in favour of Bob with probability 1 (and same for Bob). In '07 Mochon [14] created a $\mathrm{WCF}(\frac{1}{2}, \varepsilon)$ with bias ε arbitrarily close to zero.

Theorem 2.6 (Secure Balanced Weak Coin [14])**.** *There exists a secure Weak Coin Flipping Protocol.*

By letting Alice win with probability z and Bob with probability $1 - z$ we have the unbalanced version of Weak Coin Flipping.

Definition 2.9 (Unbalanced Weak Coin Flipping)**.** *An unbalanced weak coin flipping protocol with parameter z and bias ε* ($\mathrm{WCF}(z, \varepsilon)$) *is a protocol between Alice and Bob where at the end of the protocol:*

- *Alice outputs $o_A \in \{0,1\}$, Bob outputs $o_B \in \{0,1\}$ and the protocol outputs $o \in \{0,1,abort\}$*

- *If* $o_A = o_B$ *then* $o = o_A$ *otherwise* $o = abort$
- *If both parties are honest then* $\Pr[o = 0] = z$ *and* $\Pr[o = 1] = 1 - z$
- *If Alice is dishonest then* $\Pr[o = 0] \leq z + \varepsilon_A$
- *If Bob is dishonest then* $\Pr[o = 1] \leq 1 - z + \varepsilon_B$
- $\varepsilon = \max\{\varepsilon_A, \varepsilon_B\}$

If $o = 0$ *we say that Alice wins and if* $o = 1$ *we say that Bob wins.*

We now give one of the most useful results for our constructions:

Proposition 2.2 (Secure Unbalanced Weak Coin [8])**.** *Let* P *be a* $\mathrm{WCF}(\frac{1}{2}, \varepsilon)$ *protocol with* N *rounds. Then for all* $z \in [0, 1]$ *and for all* $k \in \mathbb{N}$ *there exists a* $\mathrm{WCF}(x, \varepsilon_0)$ *protocol* Q *such that:*

- Q *uses* $k \cdot N$ *rounds.*
- $|x - z| \leq 2^{-k}$.
- $\varepsilon_0 \leq 2\varepsilon$.

Corollary 2.1. *For every parameter* z *there exists an unbalanced weak coin flipping protocol* $\mathrm{WCF}(z, \varepsilon)$ *having bias* ε *arbitrarily close to zero.*

2.3 Quantum Money

In this section we will introduce the basic notions of public key Quantum Money; for an extended analysis see [1]. The construction of Aaronson and Christiano uses a simpler scheme named Quantum Money *Mini-Schemes* as well as a digital signatures scheme.

The key advantage of a public key money scheme over a private key money scheme, lies on the verification algorithm. In particular, in a public key money scheme we have one crucial property; a coin (or banknote) can be verified by anyone without any help from a Trusted Third party (TTP). In other words, in a public key money scheme, the banknotes have almost the same properties as today's banknotes; no-one can counterfeit them but anyone can verify their validity.
Quantum mechanics seem very appealing for the creation of a money scheme: No-Cloning Theorem guarantees that we can't copy an unknown quantum state, which is a good beginning step for creating money. The second step is to prevent the copy of a state even if there exists a public algorithm for verifying it.

Definition 2.10 (Quantum Money Scheme)**.** *A quantum money scheme* S *consists of three public algorithms:*

1. $\mathbf{KeyGen}(0^n) = (\mathrm{k_{pb}}, \mathrm{k_{pr}})$; *a classical algorithm which takes as input a security parameter* n *and outputs a public and a private key.*

2. $\mathbf{Bank}(0^n, \mathrm{k_{pr}}) = (s, \rho)$; *a quantum algorithm which takes as input the security parameter and the private key and outputs a valid banknote* $\$ = (s, \rho)$ *where* s *is a classical string and* ρ *is a mixed quantum state.*

3. $\mathbf{Ver}(\mathrm{k_{pb}}, (s, \rho)) = \{accept, reject\}$ *which takes a public key and a possible coin* (s, ρ) *and either accepts or rejects the coin.*

We say that S *is secure if it has two properties:*

1. *Completeness;* $\Pr[\mathbf{Ver}(\mathrm{k_{pb}}, \mathbf{Bank}(0^n, \mathrm{k_{pr}})) = accept] = 1$.

2. *Soundness; Let* $\mathbf{C}(\mathrm{k_{pb}}, \$_1, \ldots, \$_q) = (\hat{\$}_1, \ldots, \hat{\$}_{q'})$ *be a quantum algorithm (the counterfeiter) that takes the public key and* q *valid banknotes and outputs* q' *possibly entangled banknotes. Let also* $\mathbf{Count}(\mathrm{k_{pb}}, \hat{\$}_1, \ldots, \hat{\$}_{q'})$ *be a quantum algorithm, which uses* **Ver***, takes as input* q' *possible banknotes and outputs the number of them that are accepted by* **Ver***. Then for every polynomial* **C** *it holds that* $\Pr[\mathbf{Count}(\mathrm{k_{pb}}, \mathbf{C}(\mathrm{k_{pb}}, \$_1, \ldots, \$_q)) > q] = \mathrm{negl}(n)$.

The Completeness property states that a valid coin will always be accepted. The Soundness property states that there is no counterfeiter that can create more money than what he already has with non-negligible probability. Note that the verification algorithm is public in this definition; anybody can verify a coin. Note, also, that classically it is trivial to show that there is no way to construct such a scheme.

In the construction of a full quantum scheme Aaronson and Christiano used a secure digital signature scheme.

Definition 2.11. *A digital signature scheme consists of the following three probabilistic algorithms:*

1. $\mathbf{KeyGen}(0^n) = (\mathrm{k_{pb}}, \mathrm{k_{pr}})$; *a classical algorithm which takes as input a security parameter* n *and outputs a public and a private key.*

2. $\mathbf{Sign}(\mathrm{k_{pr}}, m) = s$; *a classical algorithm which takes as input the private key and a message and outputs the signature of the message.*

3. $\mathbf{Ver}(\mathrm{k_{pb}}, m, s) = \{accept, reject\}$; *a classical algorithm which takes as input the public key, a message and its potential signature and either accepts or rejects.*

The scheme is quantum secure against existential forgery under non-adaptive chosen message attacks if it has the following properties:

1. *Completeness;* $\Pr[\mathbf{Ver}(\mathrm{k_{pb}}, m, \mathbf{Sign}(\mathrm{k_{pr}}, m)) = accept] = 1$

2. *Soundness; Let* $\mathbf{C}$ *be a quantum algorithm that takes as input the public key* $\mathrm{k_{pb}}$*, gives to a signing oracle some messages* $m_1, \ldots, m_q$*, then the oracle returns their signatures and at the end* $\mathbf{C}$ *outputs a pair* (m, s) *where for all* $i \in [q]$*,* $m \neq m_i$*. Then for every* $\mathbf{C}$ *it holds that* $\Pr[\mathbf{Ver}(\mathrm{k_{pb}}, m, s) = accept] = \mathrm{negl}(n)$*.*

Theorem 2.7 (Secure Digital Signatures [17])**.** *If there exists a quantum one-way function then there also exists a quantum secure against chosen message attacks digital signature scheme.*

Theorem 2.8 (Relativized Digital Signatures [1])**.** *Relative to an oracle there exists a quantum secure-against-chosen-message-attacks digital signature scheme.*

2.3.1 Quantum Money Mini-Schemes

Informally, a mini-scheme is a scheme with two algorithms; one for producing a banknote (as before a banknote is a serial together with a quantum state) and one for validating a banknote. There is no notion of public or private keys. The goal of a counterfeiter is to create one more state that corresponds to the known serial.

Definition 2.12 (Quantum Money Mini-Schemes)**.** *A quantum money mini-scheme consists of the following two public algorithms:*

1. $\mathbf{Bank}(0^n) = (s, \rho)$*; a quantum probabilistic algorithm which takes as input a security parameter* n *and outputs a coin* $\$ = (s, \rho)$

2. $\mathbf{Ver}(s, \rho) = \{accept, reject\}$*; a quantum algorithm which takes as input a pair* (s, ρ) *and either accepts or rejects.*

The scheme is secure if it satisfies the following two properties

1. *Completeness;* $\Pr[\mathbf{Ver}(\mathbf{Bank}(0^n)) = accept] = 1$

2. *Soundness; Let* $\mathbf{C}(s, \rho) = (\rho_1, \rho_2)$ *be a quantum algorithm that takes as input a valid coin* (s, ρ) *and produces two possibly entangled states* (ρ_1, ρ_2)*. Let also* $\mathbf{Ver}_2(s, (\rho_1, \rho_2)) = (\mathbf{Ver}(s, \rho_1) \wedge \mathbf{Ver}(s, \rho_2)) = \{accept, reject\}$ *be an algorithm that takes a serial number and two possibly entangled states and accepts if and only if both* $\mathbf{Ver}(s, \rho_1)$ *and* $\mathbf{Ver}(s, \rho_2)$ *accept. Then for all polynomial* $\mathbf{C}$ *it holds that* $\Pr[\mathbf{Ver}_2(s, \mathbf{C}(s, \rho)) = accept] = \mathrm{negl}(n)$*.*

If the algorithm **Bank** first generates a random string r and then produces the coin $\$ = (s_r, \rho_r)$ then we say that the Mini-Scheme is *secret based.* Note that a party can in

general create many valid pairs. However, given a valid coin (s, ρ) no party can create two states that correspond to s. In the case of non-secret based Mini-Schemes even the **Bank** cannot create two states for the same s.

Proposition 2.3 (From secret-based Mini-Schemes to One-Way functions [1])**.** *If there exists a secure secret-based Mini-Scheme, then there also exists an one-way function secure against quantum attacks.*

Using quantum secure digital signatures and secure quantum money mini-scheme we can create a public key quantum money scheme.

Theorem 2.9 (Standard Construction [1])**.** *If there exists a quantum secure against chosen message attacks digital signature scheme and a secure quantum money mini-scheme then there also exists a public key Quantum Money Scheme.*

Corollary 2.2. *If there exists a secure secret-based Mini-Scheme, then there also exists a Quantum Money Scheme.*

Proposition 2.4 (Relativized version [1])**.** *If there exists a quantum secure against chosen message attacks digital signature scheme relative to some oracle and a secure quantum money mini-scheme relative to some other oracle then there also exists a public key Quantum Money Scheme relative to some third oracle.*

Corollary 2.3. *If there exists a secure secret-based Mini-Scheme relative to some oracle, then there also exists a public key Quantum Money Scheme relative to some other oracle.*

Aaronson and Christiano created a mini-scheme that is secure relative to some oracle. Moreover, they created a candidate for the non-relativized definition without formal proof of its security. They also proved some very interesting theorems such as that if a counterfeiter can break a mini-scheme with non-negligible probability then he can also break it with probability almost 1. Therefore, by proving that a scheme cannot be broken with probability close to 1, then it automatically comes that it is secure.

In the next chapter we will present the Ripping Money scheme. For our work it will be sufficient to consider the mini-schemes as black boxes with either relativized or non-relativized security.

3 Ripping Quantum Money

3.1 Introduction

Suppose Bob is a trader and Alice is a customer and wants to buy something from Bob. Alice doesn't trust Bob to send the money and wait for the product and Bob doesn't trust Alice to send the product and wait for the money. In some way, we want the two parties to simultaneously exchange the money for the product. In '96 Jakobsson [12] proposed the idea of letting Alice rip her coin; instead of Alice sending the coin, she rips it into two parts and sends the one part to Bob. Bob can verify that this half coin is valid but cannot use it as a full coin. Therefore, he can send the product to Alice. Alice, on the other hand, cannot use her coin any more, because she has already lost one part of it. When Alice receives the product, she can send the other half of the coin to Bob.

The previous "protocol" doesn't guarantee that Bob will send the product or Alice will send the second half but it encourages this behavior. Alice has already lost her coin, so she has no reason to keep the second half. Also, Bob hasn't got the full coin so he has no reason not to send his product. In other words, the parties can harm each other but cannot use this strategy for their own advantage.

Before giving a formal definition, let's make an important remark. We can split such a scheme into two basic phases (see fig. 3.1). In the first phase; the *Binding Phase* (BP), Alice basically is bound to some coin (she rips the coin and she sends the first half). So in this phase, Alice loses her coin and lets Bob verify that she has lost it and he partially has it. In the second phase; the *Exchange Phase* (EP), Alice sends the remaining parts of the coin to Bob; in other words she sends the second half. Note that we have omitted an intermediate phase when Bob sends the good to Alice. This is inevitable since we cannot have any security property in this phase; except perhaps the fact that Alice has to verify that the product is the expected one.

A crucial drawback that is possible in this setting, is that Bob can use this half coin to

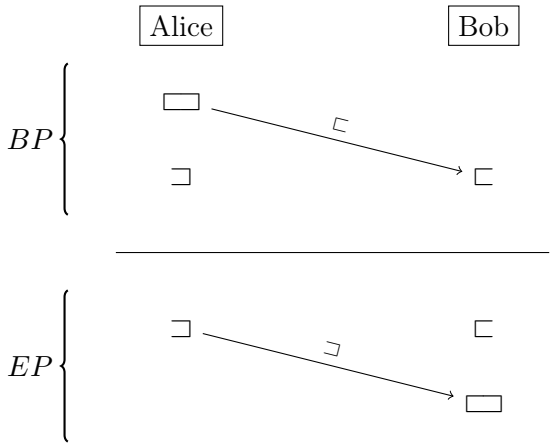


Figure 3.1: Jakobsson's Coin Ripping Idea.

buy something else, and so on, and this could lead to a chain where all players buy things with only a half coin. Fortunately, in the quantum setting we can succeed a scheme that prevents this behavior.

Definition 3.1 (Ripping Quantum Money Scheme)**.** *A ripping quantum money scheme $\mathcal{R}$ consists of the following three probabilistic algorithms:*

1. $\mathbf{KeyGen}(0^n) = (\mathrm{k_{pr}}, \mathrm{k_{pb}})$*; a classical algorithm which takes a security parameter n and outputs a private and a public key.*

2. $\mathbf{Bank}(0^n, \mathrm{k_{pr}}) = \$$*; a quantum probabilistic algorithm which takes a security parameter n and a private key and outputs a valid coin $\$ = (s, \rho)$ where s is a classical string and ρ is a quantum state.*

3. $\mathbf{Ver}(\mathrm{k_{pb}}, (s, \rho)) = \{accept, reject\}$*; a quantum algorithm which takes a public key and a possible coin (s, ρ) and either accepts or rejects the coin.*

Suppose two parties Alice and Bob where Alice possesses q_A valid coins $(\$_1^{Alice}, \ldots, \$_{q_A}^{Alice})$; same for Bob $(\$_1^{Bob}, \ldots, \$_{q_B}^{Bob})$. The scheme, also, consists of a two-phase protocol $Q_{Alice,Bob}$ between Alice and Bob:

- *Binding Phase ($BP_{Alice,Bob}$) where Alice proves to Bob that she is bound to some coin.*

- *Exchange Phase ($EP_{Alice,Bob}$) where Bob gets the coin from Alice.*

Let $\mathrm{out}_B^{Alice}, \mathrm{out}_E^{Alice}$ be the output of Alice after $BP_{Alice,Bob}$ and after $EP_{Alice,Bob}$ respectively; same for Bob. We say that $\mathcal{R}$ is secure if it satisfies the following two properties:

1. *Completeness; If both parties are honest then* $\Pr[\text{out}_B^{Bob} = accept] = 1$ *and*

$$\Pr[\text{out}_E^{Bob} = \$_1^{Bob} \otimes \cdots \otimes \$_{q_B}^{Bob} \otimes \$_i^{Alice}] = 1$$

2. *Soundness; Let* **Count** *be defined as in definition 2.10. Consider the following scenario. Suppose that Alice has already begun to run* $Q_{p_i,Alice}$ *with* q'_A *honest parties* $(p_1, \ldots, p_{q'_A})$ *where* $i \in [q'_A]$ *but has not begun* $EP_{p_i,Alice}$ *with any of them. During these interactions she begins* $Q_{Alice,Bob}$*. Then for all strategies of Alice it holds that*

$$\Pr[\mathbf{Count}(\text{k}_{\text{pb}}, \text{out}_B^{Alice}) \geq q_A \text{ and } \text{out}_B^{Bob} = accept] = \text{negl}(n)$$

Informally, a negligible soundness error implies that Alice cannot send half of her coin without losing the whole coin even if she is interacting with other honest parties. In other words, she may have any polynomial number of half coins but cannot use any of them as part of any of her coins. Therefore, a successful Binding Phase implies that Alice has lost her coin with high probability.

Next, we show that the previous definition is a tighter definition for Quantum Money schemes.

Proposition 3.1. *If there exists a Quantum-Secure Money Ripping Scheme, then there also exists a Quantum-Secure Money Scheme.*

Proof. In fact a Ripping Quantum Money Scheme satisfies the properties of a Quantum Money Scheme. Suppose that Alice has a polynomial quantum circuit $\mathbf{C}$ that can break the Money Scheme:

$$\Pr[\mathbf{Count}(\text{k}_{\text{pb}}, \mathbf{C}(\text{k}_{\text{pb}}, \$_1^{Alice}, \ldots, \$_{q_A}^{Alice})) > q_A] \geq a$$

for some non-negligible a. Then Alice can use the following strategy to break the Money Ripping Scheme:

1. She plays the $BP_{Alice,Bob}$ fairly and she is bound to the coin $\$_i^{Alice}$ for some $i \in [q_A]$. She doesn't output anything. Then
2. She outputs $\text{out}_B^{Alice} = \mathbf{C}(\text{k}_{\text{pb}}, \$_1^{Alice}, \ldots, \$_{i-1}^{Alice}, \$_{i+1}^{Alice}, \ldots, \$_{q_A}^{Alice})$

In the first step we have

$$\Pr[\text{out}_B^{Bob} = accept] = 1$$

In the second step we have:

$$\Pr[\mathbf{Count}(\text{k}_{\text{pb}}, \mathbf{C}(\text{k}_{\text{pb}}, \$_1^{Alice}, \ldots, \$_{i-1}^{Alice}, \$_{i+1}^{Alice}, \ldots, \$_{q_A}^{Alice})) > q_A - 1] \geq a$$

Since these two events are independent we have that:

$$\Pr[\mathbf{Count}(\mathrm{k_{pb}}, \mathrm{out}_B^{Alice}) \geq q_A \text{ and } \mathrm{out}_B^{Bob} = accept] \geq a$$

and therefore the Soundness Property of the Ripping Money is violated. □

3.2 Ripping Quantum Money Construction

We now present one of the most important theorems of the thesis:

Theorem 3.1. *Suppose*

- $\mathcal{M}$ *is a secure Quantum Mini-Scheme with*

$$\mathcal{M} = (\mathbf{Bank}_{\mathcal{M}}, \mathbf{Ver}_{\mathcal{M}})$$

- $\mathcal{D}$ *is a Digital Signature secure against quantum chosen message attacks with*

$$\mathcal{D} = (\mathbf{KeyGen}_{\mathcal{D}}, \mathbf{Sign}_{\mathcal{D}}, \mathbf{Ver}_{\mathcal{D}})$$

- $\mathcal{P}^x_{\mathrm{k^a_{pb}},\mathrm{k^b_{pb}}}$ *is a Quantum Secure Zero-Knowledge Proof of Knowledge for the NP relation* R *where* $(x,w) \in R \Leftrightarrow \exists\ w_1, w_2$ *s.t.* $w = w_1||w_2$ *and* $\mathbf{Ver}_{\mathcal{D}}(\mathrm{k^a_{pb}}, x, w_1) =$ *accept and* $\mathbf{Ver}_{\mathcal{D}}(\mathrm{k^b_{pb}}, w_1, w_2) =$ *accept for all signature keys* $\mathrm{k^a_{pb}}, \mathrm{k^b_{pb}}$.

Then, using $\mathcal{M}$, $\mathcal{D}$ *and* $\mathcal{P}$ *we can create a Ripping Quantum Money Scheme* $\mathcal{R}$.

Proof. First, let's define the three algoritms $\mathbf{KeyGen}_{\mathcal{R}}, \mathbf{Bank}_{\mathcal{R}}, \mathbf{Ver}_{\mathcal{R}}$:

- $\mathbf{KeyGen}_{\mathcal{R}}(0^n) = \mathbf{KeyGen}_{\mathcal{D}}(0^n)$ which produces the keys $\mathrm{k^{Bank}_{pr}}, \mathrm{k^{Bank}_{pb}}$.
- $\mathbf{Bank}_{\mathcal{R}}(0^n, \mathrm{k_{pr}})$ uses $\mathbf{Bank}_{\mathcal{M}}(0^n)$ which produces (s,ρ) and $\mathbf{Sign}_{\mathcal{D}}(\mathrm{k_{pr}}, s)$ which produces σ. The output is a valid coin of the form $((s,\sigma),\rho)$.
- $\mathbf{Ver}_{\mathcal{R}}(\mathrm{k_{pb}}, ((s,\sigma),\rho)) =$ *accept* iff $\mathbf{Ver}_{\mathcal{D}}(\mathrm{k_{pb}}, s, \sigma) =$ *accept* and $\mathbf{Ver}_{\mathcal{M}}(s,\rho) =$ *accept*

Now, let's define the protocol $Q_{Alice,Bob}$. Suppose Alice has keys $\mathrm{k^{Alice}_{pr}}, \mathrm{k^{Alice}_{pb}}$ and before the begining of the protocol she possesses the coins $(\$_1^{Alice}, \ldots, \$_{q_A}^{Alice})$ and wants to run $Q_{Alice,Bob}$ for the coin $\$_i^{Alice} = ((s_i,\sigma_i),\rho_i)$ for some $i \in [q_A]$. Bob possesses the coins $(\$_1^{Bob}, \ldots, \$_{q_B}^{Bob})$.

- Binding Phase ($BP_{Alice,Bob}$):

1. Alice sends to Bob (s_i, ρ_i).
2. Bob runs $\mathbf{Ver}_{\mathcal{M}}(s_i, \rho_i)$.
3. Alice (the prover) and Bob (the verifier) run $\mathcal{P}^{s_i}_{\mathrm{k}^{\mathrm{Bank}}_{\mathrm{pb}}, \mathrm{k}^{\mathrm{Alice}}_{\mathrm{pb}}}$ (proof that Alice knows the Bank's signature on s_i (σ_i) and her own signature on σ_i).
4. Bob outputs

$$\mathrm{out}_B^{Bob} = accept \text{ iff } \mathbf{Ver}_{\mathcal{M}}(s_i, \rho_i) = accept \text{ and } \Psi_{Bob}^{\langle Alice, Bob\rangle}(|s_i\rangle\langle s_i|) = accept$$

where $\Psi_{Bob}^{\langle Alice, Bob\rangle}$ is the superoperator of Bob that results from the ZK protocol. Alice doesn't output anything.

- Exchange Phase ($EP_{Alice,Bob}$):
  1. Alice sends to Bob σ_i.
  2. Bob outputs

$$\mathrm{out}_E^{Bob} = \$_1^{Bob} \otimes \cdots \otimes \$_{q_B}^{Bob} \otimes \$_i^{Alice}$$

Alice doesn't output anything.

Note that in the third step of the Binding phase Alice proves that she knows both σ_i and $\mathbf{Sign}_{\mathcal{D}}(\mathrm{k}^{\mathrm{Alice}}_{\mathrm{pr}}, \sigma_i)$. The reason for this type of ZK proof is to prevent Alice from proving that she knows σ_i by just forwarding messages between the owner of the coin and Bob (some kind of man-in-the-middle attack). For a graphic illustration of the protocol see fig. 3.2.

Next, suppose that Alice has already begun to run $Q_{p_i,Alice}$ with q'_A parties $(p_1, \ldots, p_{q'_A})$ where $i \in [q'_A]$ but has not begun the exchange phase with any of them. During these interactions she begins $Q_{Alice,Bob}$. Suppose, also, that there is a cheating strategy for Alice s.t.:

$$\Pr[\mathbf{Count}(\mathrm{k}_{\mathrm{pb}}, \mathrm{out}_B^{Alice}) \geq q_A \text{ and } \mathrm{out}_B^{Bob} = accept] \geq a$$

for some non-negligible a. For simplicity we represent by F the fact

$$\mathbf{Count}(\mathrm{k}_{\mathrm{pb}}, \mathrm{out}_B^{Alice}) \geq q_A \text{ and } \mathrm{out}_B^{Bob} = accept$$

and by NEW the fact 'Alice created a new coin'. Then:

$$\begin{aligned} \Pr[F] &= \Pr[NEW] \cdot \Pr[F|NEW] + \Pr[\neg NEW] \cdot \Pr[F|\neg NEW] \\ &= \Pr[NEW] + \Pr[\neg NEW] \cdot \Pr[F|\neg NEW] \geq a \end{aligned}$$

The first of the two addends corresponds to the case where the Zero-Knowledge property

(of the ZKP) is violated and the second one to the case where the Validity property (of the ZKP) is violated. Therefore, by assuming that one of them is non-negligible we come to a contradiction about the security of the ZK protocol.

In the case where $\Pr[NEW]$ is non-negligible, using the standard construction theorem of [1] we have that Alice has used both her valid coins $(\$_1^{Alice}, \ldots, \$_{q_A}^{Alice})$ and the information she got from the interactions with $(p_1, \ldots, p_{q'_A})$. Also note that in this case it is preferable for Alice to complete all $BP_{p_i,Alice}$ before beginning $BP_{Alice,Bob}$, since her interaction with Bob doesn't provide her with any information about some coin. So the things that Alice has in her hands to create a new coin are at worst:

1. Her q_A coins $(\$_1^{Alice}, \ldots, \$_{q_A}^{Alice})$.
2. The q'_A pairs (s'_i, ρ'_i) for $i \in [q'_A]$.

All of them create a big mixed quantum state τ which constitutes the auxiliary state of Alice. Also suppose a global Ψ to be the superoperator applied on τ and it's output is the output of Alice after interacting with all p_i. Then

$$\text{out}_A = \Psi(\tau)$$

and

$$\Pr[\textbf{Count}(\text{k}_{\text{pb}}, \text{out}_A) > q_A] = \text{non-negl}(n)$$

Let S be a simulator for Alice and Φ be it's operator. The states (s'_i, ρ'_i) for $i \in [q'_A]$ are of no use for S since they are perfectly indistinguishable from any other q'_A coins that are drawn randomly from $\textbf{Bank}_{\mathcal{M}}$. Furthermore, it holds that no quantum polynomial time algorithm can use the q_A valid coins to create more with non-negligible probability and therefore no such simulator S can:

$$\text{out}_S = \Phi(\tau)$$

and

$$\Pr[\textbf{Count}(\text{k}_{\text{pb}}, \text{out}_S) > q_A] = \text{negl}(n)$$

Therefore for every Φ it holds that Ψ and Φ *are* distinguishable. But Ψ consists of sub-operators that correspond to each execution of the q'_A ZK proofs and so it holds that there exists at least one execution where what Alice computes is (with non-negligible probability) distinguishable from whatever a quantum algorithm could compute using only τ.
In the case where $\Pr[\neg NEW] \cdot \Pr[F|\neg NEW]$ is non-negligible it holds that $\Pr[F|\neg NEW]$ is also non-negligible. Here, Alice first sends a valid pair (s, ρ) to Bob and then

she proves (with non-negligible probability) that she knows some $w = w_1||w_2$ s.t. $\mathbf{Ver}_{\mathcal{D}}(\mathrm{k}_{\mathrm{pb}}^{\mathrm{Bank}}, s, w_1) = \mathit{accept}$ and $\mathbf{Ver}_{\mathcal{D}}(\mathrm{k}_{\mathrm{pb}}^{\mathrm{Alice}}, x, w_2) = \mathit{accept}$. Here, the validity property of the Zero-Knowledge Proof is violated since Alice can prove that she knows w but she cannot use any polynomial extractor to learn w (and hence w_1) with non-negligible probability.

□

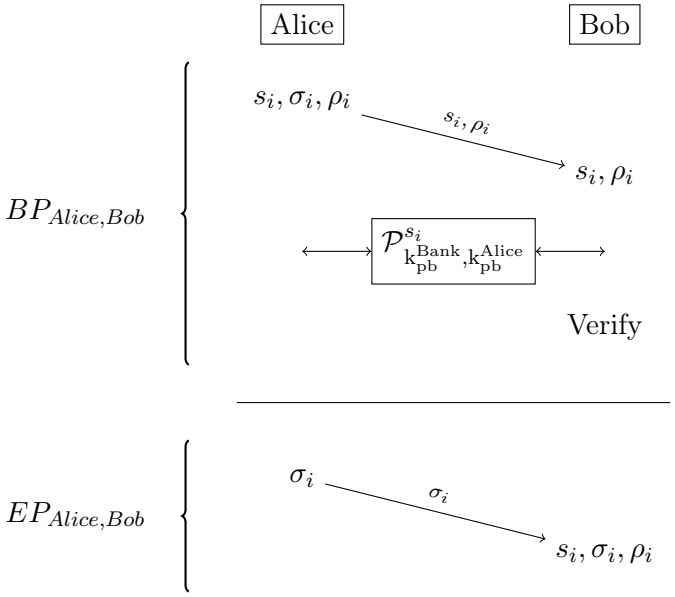


Figure 3.2: Quantum Coin Ripping Protocol. In the Binding Phase, Alice sends to Bob the quantum state ρ_i and the corresponding serial s_i. Then they run a Zero-Knowledge protocol where Alice proves to Bob that she knows both her signature and the Bank's signature on s_i. In the Exchange Phase, Alice sends to Bob Bank's signature on s_i.

Corollary 3.1 (Relativized Ripping Quantum Money)**.** *Suppose that $\mathcal{M}$ is a quantum secure mini-scheme relative to an oracle $\mathcal{O}_{\mathcal{M}}$, $\mathcal{D}$ is a quantum secure digital signature relative to an oracle $\mathcal{O}_{\mathcal{D}}$ and $\mathcal{P}$ is a quantum secure Zero-Knowledge Proof of Knowledge relative to an oracle $\mathcal{O}_{\mathcal{P}}$ (defined as above). Then there exists a secure Ripping Quantum Money scheme relative to some oracle.*

Proof. Comes from combining theorems 3.1, 2.8 and proposition 2.1. □

4 Simultaneous exchange

Suppose Alice and Bob have two secret bits s_A, s_B. They want to exchange their secrets in such a way that in every moment neither of them has better probability of knowing it than the other party. This could be succeeded perfectly if the two parties shared a simultaneous channel; they will send and receive the secret bits simultaneously. Unfortunately, simultaneous channels do not exist so they have to follow some protocol in which they gradually reveal their secrets.

4.1 Introduction

The problem of *Fair Exchange* is one of the most crucial cryptographic primitives of modern cryptography. Informally, when Alice wants to fairly exchange a secret with Bob, she wants Bob to learn her secret only if she learns his secret. Bob also wants the same thing. So in some sense they want to exchange their secrets simultaneously. Fair exchange has been studied extensively since the early 80's and much progress has been made since then. The first approaches used the rather weird assumption that both parties had the same computational power. In this type of protocols, the parties gradually reveal some information of their secret in turns and so any party would need approximately the same computational time to retrieve the secret if the other aborted. Another approach tried to bound the probabilities of each party to learn the other's secret and make these probabilities very close to one another. The impossibility result of Even [9] stating that no completely fair exchange can be achieved without the use of a third party gave rise to a whole new era where researchers tried to create protocols using the third party, as less as possible.
Simultaneous exchange of one single secret bit was first studied by Luby et al. in '83 [13]. In their protocol they make use of the Quadratic Residuosity Assumption. Of course, in the quantum setting, many assumptions like this are not longer valid; Shor's factoring algorithm [18] can be used to decide if a given number is a quadratic residue. Therefore, it has been an open problem whether simultaneous exchange is possible and to what

extent in the information theoretic setting. In this work we extend the protocol of [13] to the quantum setting.

Before giving the formal definitions we need, we must explain what we mean with a secret bit. The assumption we make is that there must be some procedure with which the parties verify that the secret is indeed the correct one. The most convenient way to think about it, is by using some completely secure digital signature or bit commitment scheme. Then, when the party learns the secret they can recognize it and verify that it is indeed the correct bit. Another way of thinking it, is by assuming that the parties decide their secret by the time the send it. Therefore, the secret of a party is the bit they sent. We note that we do not attempt to address this problem; what we care about is the probabilities of each party guessing the secret and not the procedure of verifying the secret. We want them to exchange their secrets at the same time on average.

Definition 4.1 (Simultaneous exchange)**.** *A simultaneous exchange protocol with error ε and n rounds is a protocol between Alice and Bob where:*

- *Alice has a secret bit $s_A \in_R \{0,1\}$ and Bob has a secret bit $s_B \in_R \{0,1\}$.*
- *At every round $i \in [n]$ Alice outputs $o_A^i \in \{0,1,Abort\}$ and Bob outputs $o_B^i \in \{0,1,Abort\}$.*
- *If a party aborts then the protocol aborts.*
- *If both are honest then $\Pr[o_A^n = s_B] = \Pr[o_B^n = s_A] = 1$*
- *If a party is dishonest then for all rounds $i \in [n]$ it holds that*

$$\left|\Pr[o_A^i = s_B] - \Pr[o_B^i = s_A]\right| \leq \varepsilon$$

Note that ε is at most $\frac{1}{2}$ since we can create a trivial protocol where Alice sends her bit and then Bob responds. At the beginning of the protocol both parties have $\frac{1}{2}$ to guess the secret. Immediately after Bob receives Alice's bit, his probability becomes 1 but Alice's probability remains $\frac{1}{2}$. Ideally, we would like a protocol where the difference between the two probabilities be inversely proportional to the number of rounds. So, by increasing the number of rounds, the difference goes to zero.

4.2 A simultaneous exchange with error 1/4

We begin by a first attempt to decrease the probability difference. The error of a simultaneous exchange can be reduced to almost $\frac{1}{4}$ using a balanced quantum weak coin protocol as a subprotocol.

Theorem 4.1. *There exists a quantum simultaneous exchange protocol with error $\frac{1}{4} + \varepsilon$.*

Proof. The protocol consists of the following steps; see fig. 4.1:

1. Alice and Bob flip a balanced quantum weak coin with bias 2ε (WCF($\frac{1}{2}, 2\varepsilon$)).
2. If Alice wins then she sends nothing to Bob. Otherwise she sends her secret s_A.
3. Bob responds with his secret s_B.
4. If Alice had won the weak coin she sends her secret s_A. Otherwise, this step is omitted.

First let's focus on the round after the weak coin (say the k^{th} round):
Both honest fig. 4.1(a)
Bob has probability $\frac{1}{2}$ to win the weak coin (in which case he knows the bit with certainty) plus $\frac{1}{2}$ to lose and so he has to flip a coin (fig. 1(a)):

$$\begin{aligned} \Pr[o_B^k = s_A] &= \Pr[\text{Alice loses}] \cdot \Pr[o_B^k = s_A | \text{Alice loses}] \\ &+ \Pr[\text{Alice wins}] \cdot \Pr[o_B^k = s_A | \text{Alice wins}] \\ &= \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4} \end{aligned}$$

Dishonest Alice fig. 4.1(b)
Bob has probability $\frac{1}{2} - 2\varepsilon$ to win the coin flipping plus $(\frac{1}{2} + 2\varepsilon)\frac{1}{2}$ to lose and so he has to flip a coin (fig. 1(b)):

$$\Pr[o_B^k = s_A] = \frac{1}{2} - 2\varepsilon + (\frac{1}{2} + 2\varepsilon) \cdot \frac{1}{2} = \frac{3}{4} - \varepsilon$$

Dishonest Bob fig. 4.1(c)
Bob has probability $\frac{1}{2} + 2\varepsilon$ to win the coin flipping plus $(\frac{1}{2} - 2\varepsilon)\frac{1}{2}$ to lose and so he has to flip a coin (fig. 1(c)):

$$\Pr[o_B^k = s_A] = \frac{1}{2} + 2\varepsilon + (\frac{1}{2} - 2\varepsilon) \cdot \frac{1}{2} = \frac{3}{4} + \varepsilon$$

In all three cases when Bob responds with his secret bit, Alice's probability of guessing it becomes 1. When Alice sends her secret bit, Bob's probability becomes 1.

What we also need to show is that during the rounds of the weak coin neither of the parties can increase his probability of guessing the other's secret. First, during the weak coin, neither of the parties uses his secret and therefore neither can gain information about the other's secret. Second, during the weak coin, neither party can increase his probability of winning without the other party aborting, since the weak coin flipping wouldn't be secure. □

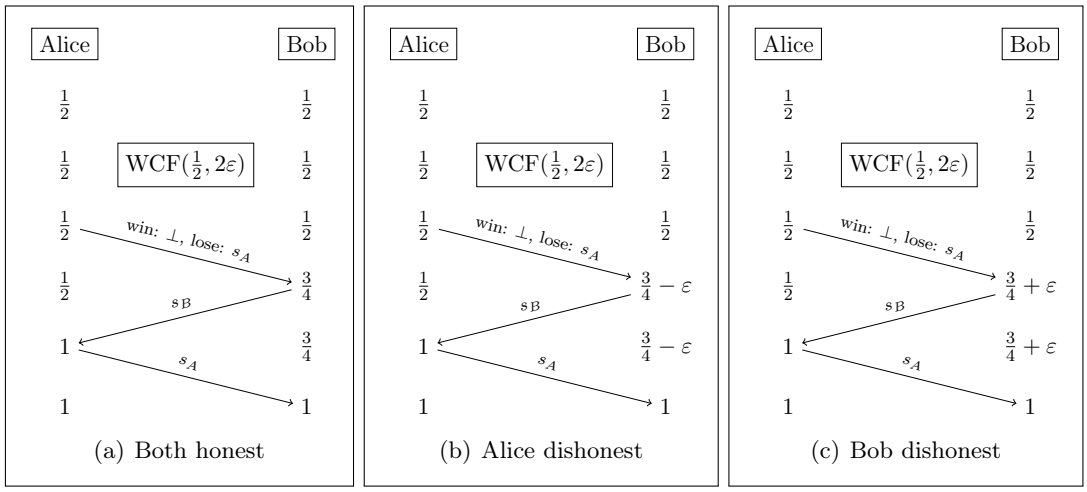


Figure 4.1: A protocol with error $\frac{1}{4} + \varepsilon$. The numbers below each party correspond to the probabilities of guessing the other's bit in each round.

4.3 A simultaneous exchange with arbitrarily small error

Before analyzing our optimal protocol let's first focus on the protocol of Micali et 'al [13]. Intuitively, suppose that Alice and Bob share a two-sided wheel. The wheel is placed in such way that each party can see only their own side. According to the secret she has, Alice paints the $\frac{1}{2} + \varepsilon$ of her surface with red and the $\frac{1}{2} - \varepsilon$ with green or vice-versa depending on whether she chose her secret to be 0 or 1 respectively. Same for Bob on the opposite side of the wheel (see figure 4.2(a)). Then, they cover the wheel and they turn it in such a way that if the wheel is uncovered, each party will see the way the other side was painted. Then, they follow the procedure below iteratively:

1. Alice spins the wheel and stops it at random.
2. Alice opens just a point of the surface in front of her and sees the color of that point.
3. Bob spins the wheel and stops it at random.
4. Bob opens just a point of the surface in front of him and sees the color of that point.

Notice that in every step

$$\Pr[\text{Alice guesses Bob's secret}] \leq \Pr[\text{Bob guesses Alice's secret}] + \varepsilon$$

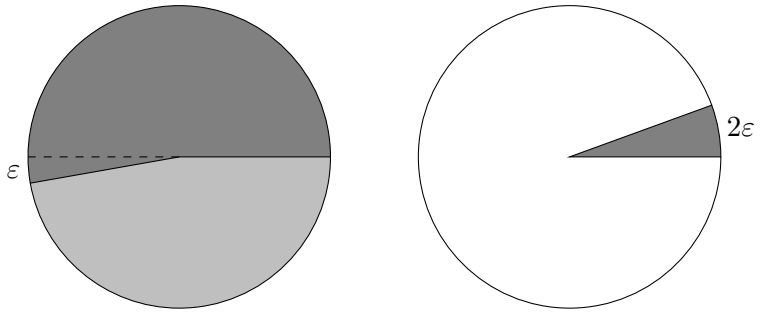
Now consider the following scenario. Instead of painting some portion of the surface red and some green, they paint only a small portion (say 2ε) of the surface with red or green (for 0 or 1 respectively) and they leave the rest surface white (see figure 4.2(b)). Now, if

a party was lucky and opened a point in this small portion of the surface he can be sure that he correctly guessed the secret bit. If, on the other hand, she opened a point on the white surface, she has to flip a coin. More formally, if we name C the fact 'Alice opened a colored point' then

$$\Pr[o_A = s_B] = \Pr[C]\cdot\Pr[o_A = s_B|C]+\Pr[\neg C]\cdot\Pr[o_A = s_B|\neg C] = 2\varepsilon+(1-2\varepsilon)\frac{1}{2} = \frac{1}{2}+\varepsilon$$

Taking it one step further, suppose that in every step these small portions are increased in such a way that:

1. Pr[Alice guesses Bob's secret] $= \frac{1}{2} + \varepsilon$
2. Pr[Bob guesses Alice's secret] $= \frac{1}{2} + 2\varepsilon$
3. Pr[Alice guesses Bob's secret] $= \frac{1}{2} + 3\varepsilon$
4. $\cdots$



(a) Monte-Carlo version (b) Las-Vegas version

Figure 4.2: Simultaneous exchange with error ε

Then this strategy can bring the two probabilities as close as we want by decreasing ε as much as we want.
Looking at these two different approaches we see how they relate to the probabilistic algorithms; the first approach corresponds to the Monte-Carlo algorithms, whereas our approach corresponds to the Las-Vegas algorithms. We continue by giving one of the main theorems of this thesis which is based mainly on the previous idea.

Theorem 4.2. *There exists a quantum simultaneous exchange protocol with error $\varepsilon + \varepsilon'$.*

Proof. The protocol consists of the following 6 steps that are repeated until both parties learn the secrets. In the begining $i = 1$.

1. $WCF(1 - 2i\varepsilon, \varepsilon')$

2. If Alice loses then she sends s_A. Otherwise she sends a random bit.
3. $i \leftarrow i + 1$
4. $WCF(2i\varepsilon, \varepsilon')$
5. If Bob loses then he sends s_B. Otherwise he sends a random bit.
6. $i \leftarrow i + 1$

We analyze the probabilities of each player to guess the others secret in each step.
We will just give the probabilities for the case where both parties are honest. The analysis in the dishonest cases comes as before.
At the end of the 2nd step, Bob's probability of guessing s_A is:

$$Pr[o_B = s_A] = 2i\varepsilon + (1 - 2i\varepsilon) \cdot \frac{1}{2} = \frac{1}{2} + i\varepsilon$$

whereas Alice's probability of guessing s_B is:

$$Pr[o_A = s_B] = 2(i-1)\varepsilon + (1 - 2(i-1)\varepsilon) \cdot \frac{1}{2} = \frac{1}{2} + (i-1)\varepsilon$$

and therefore the difference between the two probabilities is ε.
The proof that within the WCF the probabilities remain the same comes like before using also Proposition 2. If one of the parties is cheating then he slightly increases his probability of guessing the bit or hiding his own bit (see fig. 4.3). □

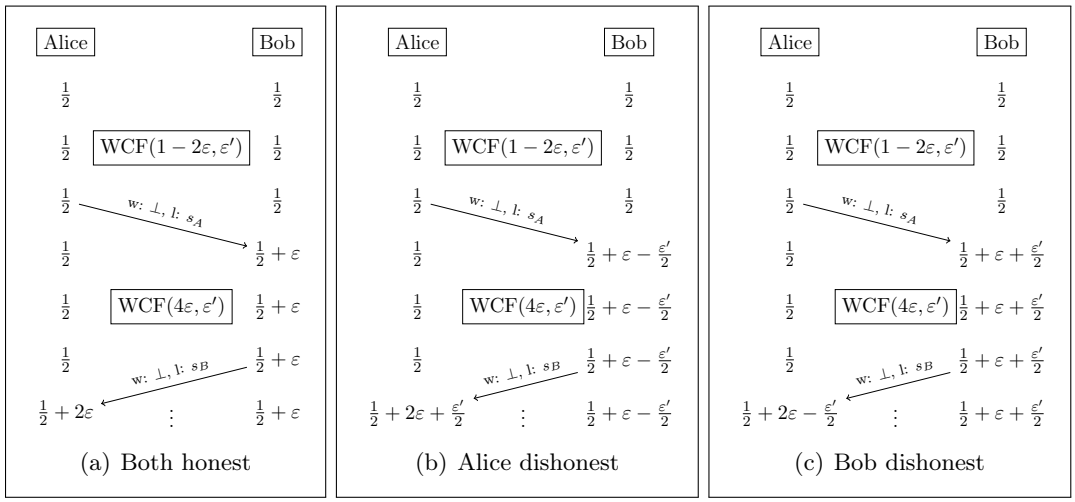


Figure 4.3: A protocol with error $\varepsilon + \varepsilon'$. The numbers below each party correspond to the probabilities of guessing the other's bit in each round.

By taking the bias ε' of the weak coin arbitrarily close to zero, we can take a protocol with error arbitrarily close to ε.

A question that arises from the definition given for simultaneous exchange is whether the existence of imperfect simultaneous exchange (such as the above) implies weak or strong coin flipping (perfect or imperfect).

Bibliography

[1] Scott Aaronson and Paul Christiano. Quantum money from hidden subspaces. In *Proceedings of the 44th symposium on Theory of Computing*, STOC '12, pages 41–60, New York, NY, USA, 2012. ACM.

[2] Mark Adcock and Richard Cleve. A quantum goldreich-levin theorem with cryptographic applications. In Helmut Alt and Afonso Ferreira, editors, *STACS 2002*, volume 2285 of *Lecture Notes in Computer Science*, pages 323–334. Springer Berlin Heidelberg, 2002.

[3] Dorit Aharonov, Alexei Kitaev, and Noam Nisan. Quantum circuits with mixed states. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, STOC '98, pages 20–30, New York, NY, USA, 1998. ACM.

[4] C. H. Bennett and G. Brassard. Quantum Cryptography: Public Key Distribution and Coin Tossing. In *Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing*, pages 175–179, New York, 1984. IEEE Press.

[5] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM J.Sci.Statist.Comput.*, 1996.

[6] A. Chailloux and I. Kerenidis. Optimal bounds for quantum bit commitment. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pages 354 –362, oct. 2011.

[7] A. Chailloux, I. Kerenidis, and J. Sikora. Lower Bounds for Quantum Oblivious Transfer. *ArXiv e-prints*, July 2010.

[8] André Chailloux and Iordanis Kerenidis. Optimal quantum strong coin flipping. In *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '09, pages 527–533, Washington, DC, USA, 2009. IEEE Computer Society.

[9] Shimon Even. A protocol for signing contracts. *SIGACT News*, 15(1):34–39, January 1983.

[10] Oded Goldreich. *Foundations of Cryptography: Basic Tools*. Cambridge University Press, New York, NY, USA, 2000.

[11] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. *J. ACM*, 38(3):690–728, July 1991.

[12] Markus Jakobsson. Ripping coins for a fair exchange. In *Proceedings of the 14th annual international conference on Theory and application of cryptographic techniques*, EUROCRYPT'95, pages 220–230, Berlin, Heidelberg, 1995. Springer-Verlag.

[13] Michael Luby, Silvio Micali, and Charles Rackoff. How to simultaneously exchange a secret bit by flipping a symmetrically-biased coin. In *Foundations of Computer Science, 1983., 24th Annual Symposium on*, pages 11 –22, nov. 1983.

[14] C. Mochon. Quantum weak coin flipping with arbitrarily small bias. *ArXiv e-prints*, November 2007.

[15] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, New York, NY, USA, 10th edition, 2011.

[16] N. Paunković, J. Bouda, and P. Mateus. Fair and optimistic quantum contract signing. *Phys. Rev. A*, 84:062331, Dec 2011.

[17] J. Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, STOC '90, pages 387–394, New York, NY, USA, 1990. ACM.

[18] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pages 124 –134, nov 1994.

[19] Dominique Unruh. Quantum proofs of knowledge. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 135–152. Springer Berlin Heidelberg, 2012.

[20] J. Watrous. Zero-knowledge against quantum attacks. *SIAM Journal on Computing*, 39(1):25–58, 2009.