2016

On quantum computation capabilities in an information assurance context

Garrett Ridge Yord
Iowa State University

On quantum computation capabilities

in an information assurance context

by

Garrett Yord

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Information Assurance

Program of Study Committee:

Yiu T. Poon, Major Professor

Clifford Bergman

Doug Jacobson

Iowa State University

Ames, Iowa

2016

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my thanks to my major professor, Dr. Yiu T. Poon, without whom this work would not have been possible. His insight and encouragement have given me new hope for the future. I would also like to thank my committee members as well as Ginny Anderson for their patience and guidance through this process.

ABSTRACT

Quantum computers harness quantum mechanical properties to perform computations beyond the scope of traditional computers. Quantum parallelism gives quantum computers incredible computational power and allows for algorithms such as Shor's algorithm which breaks many modern cryptosystems. The no-cloning theorem and measurement properties of quantum systems give rise to new cryptosystems. Entanglement allows for many new algorithms including error code correction which would otherwise be impossible in such systems (given the no-cloning theorem). This paper explores these concepts from an information assurance standpoint as they have significant implications for information systems.

CHAPTER 1. OVERVIEW

Quantum computers harness quantum mechanical properties to perform computations beyond the scope of traditional computers. Quantum parallelism gives quantum computers incredible computational power and allows for algorithms such as Shor's algorithm which breaks many modern cryptosystems. The no-cloning theorem and measurement properties of quantum systems give rise to new cryptosystems. Entanglement allows for many new algorithms including error code correction which would otherwise be impossible in such systems (given the no-cloning theorem). These concepts are important from an information assurance standpoint as they have significant implications for information systems.

Chapter 1 gives an overview of information assurance and introduces symmetric and asymmetric key cryptosystems. It gives a motivating example of a two-state quantum system and explains, conceptually, how quantum computers encoding information. Chapter 2 gives information on quantum mechanical properties which are taken advantage of by quantum computers. It begins with the postulates of quantum mechanics, including the measurement postulate which is important to quantum key distribution. It then covers specific properties such as the no-cloning theorem, entanglement, and quantum parallelism. Chapter 3 covers Shor's algorithm which breaks many currently used cryptosystems including RSA. Chapter 4 explores notions from quantum error correction. Chapter 5 explains the BB84 key distribution protocol as well as a second protocol which uses concepts from the BB84 protocol in a trusted server model. Chapter 6 discusses applications of new technologies which employ quantum key distribution as well as problems left by Shor's algorithm. It ends with a synopsis of what is being done to solve these problems.

1.1 Introduction

Information assurance is the process of managing risks related to information and information systems. These systems have traditionally been digital computers, but with the advent of quantum computers drawing ever closer, these processes must incorporate these new systems. Information assurance is particularly concerned with confidentiality, integrity, authenticity, availability, and non-repudiation. Quantum computers pose new problems and challenges on these fronts. Of particular interest in regards to confidentiality are Shor's algorithm and quantum key distribution. Shor's factorization algorithm, which can only be run on a quantum computer, can factor large numbers into primes very quickly and has the potential to break many of the cryptosystems currently in use on the internet. Quantum key distribution, on the other hand, offers a desirable property not available from digital key distribution protocols in that it allows for the detection of eavesdroppers. An additional aspect of quantum computers which is not present in digital computers is that, due to the quantum nature of how information is encoded, bits in a quantum computer (qubits) cannot copied. This makes error correction (which pertains to the integrity of information) more complex.

1.2 Information Assurance

Information assurance is the practice of managing risks related to information and information systems by ensuring confidentiality, integrity, authenticity, availability, and non-repudiation of said information. Elaborating on these properties: confidentiality is the property that information is only made available to authorized individuals, entities, and processes. For example, if two parties, Alice and Bob, are having a confidential conversation, they want to have some assurance that any third parties, such as Eve, cannot eavesdrop. Integrity is the property that data is complete and accurate and that no unauthorized or unintended modifications can or could have been made to the data. For example, if Alice sends a letter to Bob, she wants to be sure that not only does Eve not modify the contents, but also that no incidental damage occurs to the letter, such as water damage. Availability is the property that information is readily accessible to authorized individuals, entities, or processes whenever it is

needed. Maintaining a website is a good example of this: the site needs to be available for it to be of any use. Non-repudiation is the property that actions on or changes to data can be associated with the party responsible for the actions or changes. For example, if an individual sends a transaction, that individual cannot later deny having sent it.

Within the purview of information assurance are many of the systems in place on the internet. Online shopping, online banking, secure browsing; all of these require encryption and authentication mechanisms for them to operate securely. Two of the most fundamental mechanisms to build the security for these processes are symmetric key cryptosystems and asymmetric key cryptosystems.

1.2.1 Symmetric Key Cryptosystems

An important example of an information assurance process in practice are symmetric key cryptosystems. Symmetric key cryptosystems are used to allow two parties to communicate securely. Both parties share a common secret (the key) which allows one party to encrypt a message and allows the second party to decrypt the message with the same key. Without the key, the message cannot be read.

1.2.1.1 One-Time Pad

An important example of a symmetric key cryptosystem is the One-Time Pad cryptosystem. In this case, the two parties, which are typically referred to as Alice and Bob, respectively, share a randomly generated string of characters which is of the same length (or longer) than the message which will be encrypted. These characters can be from a standard alphabet such as Latin characters: {a, b, c, ..., x, y, z}. The alphabet can also be extended to additional characters: {a, b, c, ... , x, y, z, 0, 1, 2, ..., 8, 9}. Two important aspects of the choice of alphabet are that it needs to include all of the symbols which will appear in the unencrypted (or plaintext) message and the alphabet must be an ordered set of characters. This means, for example, that the sets {a, b, c, ... , x, y , z} and {z, y, x, ... ,c, b, a} would be two different alphabets in this context.

With their shared key, Alice can encrypt a message of equal or lesser length than the key.

Table 1.1 Example cipher

	A	T	T	A	C	K	plaintext
	0 (A)	19 (T)	19 (T)	0 (A)	2 (C)	10 (K)	plaintext: integer translation
+	25 (Z)	9 (J)	22 (W)	7 (H)	18 (S)	5 (F)	key: integer translation
=	25	28	41	7	20	15	plaintext + key
= (mod 26)	25 (Z)	2 (C)	15 (P)	7 (H)	20 (U)	15 (P)	plaintext $+_{26}$ key
	Z	C	P	H	U	P	cipher text

To do this, she creates a map between the integers and her alphabet using the ordering of the alphabet. So, for example, in the alphabet {a, b, c, ... ,x, y ,z}, a $\rightarrow$ 1 and z $\rightarrow$ 26. She can then combine her key with her plaintext message by using modular addition. To do this, she adds the corresponding integer values of each character in her plaintext message with its corresponding character in the key mod the length of the alphabet. So, for example, if she is using the alphabet A, B, C, ... X, Y, Z, has plaintext message: "ATTACK" and has key "ZJWHSF" she would produce "ZCPHUP". To decypt this message, Bob simply uses his copy of the shared key to perform the operation in reverse.

While symmetric cryptosystems such as this offer desirable security properties and facilitate two way communication, the distribution of keys is, in practice, problematic. If Alice wants to share a key with Bob, not only does she have the problem of transmitting the key securely to Bob so that no one else can read the key, but she also must verify that she is indeed transmitting it to Bob and not to someone else. There are several solutions to this problem, one of which is asymmetric key cryptography.

1.2.2 Asymmetric Key Cryptosystems

A second form of encryption which allows one party to send messages securely to a second party is asymmetric key encryption. In contrast to symmetric key encryption, this is one way communication and is based on one party having a set of keys: a public key and a private key. The public key is published for anyone to see and can be used to encrypt a message. The private key, on the other hand is kept secret and is used to decrypt messages. Although symmetric

key encryption may at first seem superior in that it allows two parties to communicate back and forth, asymmetric key encryption has a few advantages. First, since the private key is kept secret, it allows the key pair to be reused. Additionally, in certain situations (for example, online transactions) asymmetric key cryptography can be used to protect a symmetric key. An important example of asymmetric key encryption for quantum information theory which is widely used for internet security is RSA.

1.2.2.1 RSA Algorithm

RSA encryption is based on the assumption that factoring large numbers is difficult. Classically, there is no known algorithm to factor an arbitrary integer number into its prime factors in polynomial time.

Alice wants to create a key pair using the RSA algorithm. She does the following:

1. Pick two large primes: p, q and calculate $n = pq$.
2. Find the totient of n as $\phi(n) = (p-1)(q-1)$.
3. Pick an integer e such that $1 < e < \phi(n)$ such that e is relatively prime to $\phi(n)$.
4. Compute the modular inverse d of e (i.e. find d such that $de = 1(mod\ \phi(n))$
5. n and e are made public and comprise the public key while d is kept private and is the private key (p, q, and $\phi(n)$ must also be kept private).

Now, if Bob wants to send a message to Alice, he can use her published private key to encrypt a message as follows:

6. Write message M in the form of an integer (a natural language message can be translated into an integer using ASCII, for example)
7. Calculate $E = M^e(mod\ n)$

Bob sends this message E to Alice, who can decrypt it as follows:

8. $E^d = (M^e)^d = M^{ed} = M^{N\phi(n)+1} = M(mod\ n)$

The security of the scheme is based on the fact that to find d, one must factor n. While there is no known classical algorithm which can do this efficiently, there is an algorithm known as Shor's algorithm which can only be run on a quantum computer which can factor an arbitrary integer in polynomial time.

RSA in particular, is used in the TLS protocol which allows web sites to communicate securely between their web servers and web browsers. The security of this algorithm can be broken by a quantum computer.

1.3 Quantum Computation

In a traditional computer, electricity is used to encode information. A two state system is used wherein a high state (electricity on) is represented by a 1 and a low state (electricity off) is represented by a 0. We call a unit of information in this system a bit. Computation takes place on a collection of input bits by running electricity though a system of gates which represent logical operations on the input bits and gives corresponding outputs.

In contrast, in a quantum computer, information is encoded on two state quantum systems. For example, the polarization states of a photon are one such system. Exploring this example in more detail can give insight into how such systems work.

1.3.1 Polarization States of the Photon

Say Alice has a photon source and which can emit single photons. She directs this source at a linear polarization filter rotated to only allow horizontally polarized light through and places a detector after the filter. If she emits one photon, then she can detect whether it passed through the screen or not. If she does this multiple times, she can measure the probability that a photon from her source will pass through the screen. Say that there is a probability p_0 that a photon from her source passes through the screen and a probability p_1 that it will not.

Alice adds a second polarization filter behind the first filter and before the detector. She wants to try three different orientations for the second filter: rotated to only allow vertically polarized light through, rotated to only allow horizontally polarized light through, and rotated

to allow diagonally polarized light through. For the first configuration, where the second polarization filter will only allow vertically polarized light through, Alice finds that no photons make it to the detector. In the second configuration where the second filter will only allow horizontally polarized light through, she finds that the probability that a photon makes it through remains p_0. For the third configuration, where the second filter will only allow diagonally polarized light through, she finds that the probability that a photon makes it through is $\frac{1}{2}p_0$.

These results may seem odd at first glance, but they can be made sense of by using linear algebra: let the vector $|0\rangle = \begin{pmatrix} 1 \\ \\ 0 \end{pmatrix}$ represent the horizontal polarization state and the vector $|1\rangle = \begin{pmatrix} 0 \\ \\ 1 \end{pmatrix}$ represent the vertical polarization state. Then the general polarization state of a photon can be written as:

$$|\psi\rangle = a|0\rangle + b|1\rangle, \text{ where } a, b \in \mathbb{C} \text{ and } |a|^2 + |b|^2 = 1$$

(such a linear combination is referred to as a superposition of the states $|0\rangle$ and $|1\rangle$). For example, diagonal and anti-diagonal polarization are represented (respectively) by the states:

$$|+\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \text{ and } |-\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle).$$

Even circular polarization states can be represented. For example, the right-circular polarization and left-circular polarization state are represented (respectively) by:

$$|R\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \text{ and } |L\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle - i|1\rangle).$$

We can use these quantities to find projection operators. Such operators can be used to find the expected values of a measurement. For example:

$$|0\rangle\langle 0| = \begin{pmatrix} 1 \\ \\ 0 \end{pmatrix} (1\ 0) = \begin{pmatrix} 1 & 0 \\ \\ 0 & 0 \end{pmatrix}.$$

The expectation value that a photon in a state $|\psi\rangle$ will pass through a horizontal polarization filter is then:

$$\langle 0\rangle_{|\psi\rangle} = \langle\psi|0\rangle\langle 0|\psi\rangle \ .$$

Consider a photon in the state $|\psi\rangle = \sqrt{p_0}|0\rangle + \sqrt{p_1}|1\rangle$. If this photon is directed at a polarization filter which only lets horizontally polarized light through, then it will pass through the filter with an expected value of:

$$\langle 0\rangle_{|\psi\rangle} = \langle\psi|0\rangle\langle 0|\psi\rangle = \begin{pmatrix}\sqrt{p_0} & \sqrt{p_1}\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}\sqrt{p_0}\\ \sqrt{p_1}\end{pmatrix} = p_0.$$

Say the photon passes through the filter. Since the filter only allows horizontally polarized light through, immediately after it passes through, the photon is in state $|0\rangle$. Now, if the photon comes up against a filter which only lets vertically polarized light through, it will pass through with expected value:

$$\langle 1\rangle_{|0\rangle} = \langle 0|1\rangle\langle 1|0\rangle = \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = 0.$$

Similarly, if the photon encounters another horizontal polarizer immediately after passing through the first, it will make it through the second with expected value:

$$\langle 0\rangle_{|0\rangle} = \langle 0|0\rangle\langle 0|0\rangle = \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = 1.$$

Finally, if the photon encounters a diagonal polarizer immediately after the first, (i.e. a polarization filter which only allows light in the $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ state pass through), then it will pass through this filter with expected value:

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

Notice that these results correspond with the results Alice obtained from her experiments: there is a probability of $p_0 \cdot 0 = 0$ that a photon will pass through a horizontal polarizer followed by a vertical polarizer to the detector, there is a probability of $p_0 \cdot 1 = p_0$ that it will pass through a horizontal polarizer followed by a horizontal polarizer to the detector, and a probability of $p_0 \cdot \frac{1}{2}$ that the photon will pass through a horizontal polarizer followed by a diagonal polarizer to the detector (it is important to note that Alice's photon source will most likely not emit only photons in the state $|\psi\rangle = \sqrt{p_0}|0\rangle + \sqrt{p_1}|1\rangle$ but the ensemble of photons emitted will have a distribution of states which has this state as an average).

1.3.2 Quantum Computers

Quantum computers use two-state quantum systems, such as the polarization states of the photon to encode information. Now, rather than the fundamental unit of information being a bit, which is either a 0 or 1, the fundamental unit of information is called a qubit. Measurement

in such a system is done with respect to a chosen basis with basis vectors represented by $|0\rangle$ and $|1\rangle$ and a general state in this system is a linear combination (called a superposition in this context) of the states $|0\rangle$ and $|1\rangle$:

$$|\psi\rangle = a|0\rangle + b|1\rangle$$

where

$$|a|^2 + |b|^2 = 1$$

Similarly to traditional computers, quantum computers run algorithms on a system of gates which perform logical operations on the inputs. Because of the quantum nature of the system, there are often additional steps which need to be taken in a computation. For example, errors can occur due to imperfections in the gates, etc. and error correction may need to be done on the system.

CHAPTER 2. QUANTUM COMPUTATION

2.1 introduction

Quantum computers employ two-state quantum systems to encode information. By making use of quantum mechanical properties of such systems, quantum computers can perform computations beyond the scope of traditional computers. This chapter gives information on quantum mechanical properties relevant to quantum computation and begins to build concepts which are integral to Shor's algorithm, quantum key distribution, and quantum error correction; the postulates of quantum mechanics, the no-cloning theorem, entanglement, and quantum parallelism (which will be integral to Shor's algorithm) are covered.

2.2 Quantum Mechanics

Quantum mechanics is a branch of physics concerned with quantized systems, such as atoms and photons. The postulates of quantum mechanics describe how these systems behave.

2.2.1 Postulates of Quantum Mechanics

There are many different (yet equivalent) mathematical formulations of quantum mechanics. All of these formulations give a mathematical description of states (i.e. a mathematical description of what conditions describe a physical system at a given time) and the time evolution of states (i.e. how a state changes with time). Additionally, a mathematical formulation of quantum mechanics must also cover properties relating to measurement. The state of a quantum system cannot be directly observed and only certain properties of a quantum system can be found. These properties are called obeservables. A process which allows an observer to find an observable is called "measurement" and after measurement, a phenomena known

as wave-function collapse occurs. Wave-function collapse is of great importance to quantum key-distribution and quantum error correction (and to quantum computing in general).

States: The state of a physical system S is completely described by a unit vector $|\psi\rangle$ in a Hilbert space H_S (a complete metric space with norm $\langle\cdot|\cdot\rangle$).

Time Evolution: The time evolution of a state $|\psi\rangle$ is governed by the Schrödinger equation

$$i\hbar\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle$$

where H, the Hamiltionian, is a self-adjoint linear operator and $\hbar = \frac{h}{2\pi}$ where h is Planck's constant.

Observables: Any observable $\tilde{A}$ has an associated self-adjoint operator A on the Hilbert space H_S. The only possible outcome of a measurement of the observable $\tilde{A}$ is one of the eigenvalues of A (for convenience, we use A for both the observable and the operator).

Measurement and Wave-function Collapse: If a system S is described by the vector $|\psi\rangle$ and a measurement of $|\psi\rangle$ is taken for observable A giving outcome a_n, then immediately after the measurement, the state of the system is:

$$\frac{P_n|\psi\rangle}{\sqrt{\langle\psi|P_n|\psi\rangle}}$$

where P_n is the projection operator on the subspace corresponding to a_n.

The concept of wave-function collapse is of particular importance in quantum information theory since measuring a state changes the state. To put this another way, if Alice passes a qubit to Bob, but Eve looks at the qubit on the way to Bob, Bob will not, in general, receive a qubit in the same state which Alice sent. This fact can be used to detect eavesdroppers.

2.2.2 Two State Quantum Systems and Qubits

Two-state quantum systems are of special significance in quantum computation since such systems are used as qubits. Two-state quantum systems are represented by vectors in a two-dimensional complex Hilbert space (for example, the polarization states of the photon). Qubits are represented by unit vectors in the vector space $\mathbb{C}^2$, with orthonormal basis:

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

.

What these vectors correspond to physically depends on the physical set up of the system. For example, in the polarization example given earlier, these states represented horizontal and vertical polarization, respectively.

2.2.3 No-Cloning Theorem

In the case of a classical bit, one can make always make a copy of a bit. This is not the case with qubits. The No-Cloning theorem states that it is impossible to make a copy of an arbitrary unknown state. For example, it would be impossible to take a state:

$$|\psi\rangle = a|0\rangle + b|1\rangle$$

where a and b are unknown, and generate two copies of $|\psi\rangle$. This is of great importance to quantum-error correction as it makes creating error correction codes more difficult. Additionally, this theorem, together with the measurement postulate (wave-function collapse), is fundamental to quantum-key distribution.

2.2.4 Entanglement

Entanglement is a phenomenon which can occur when a quantum system has more than one component (for example, when a system has two or more photons instead of one). If a quantum system has n components, then its Hilbert space will be:

$$H = H_1 \otimes H_2 \otimes ... \otimes H_n$$

where $H_1, H_2, ...H_n$ are the Hilbert spaces of the n components, respectively. A state $|\psi\rangle$ in this system is said to be *seperable* if it can be written as:

$$|\psi\rangle = c(|\psi\rangle_1 \otimes |\psi\rangle_2 \otimes ... \otimes |\psi\rangle_n)$$

where $|\psi\rangle_1 \in H_1, |\psi\rangle_2 \in H_2, ...|\psi\rangle_n \in H_n$ and $c \in \mathbb{C}$ is a normalization constant. The general form for states in this system, however. For example, let $|0\rangle, |1\rangle$ be the standard basis in $\mathbb{C}^2$. Then in $H = \mathbb{C}^2 \otimes \mathbb{C}^2$, $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |1\rangle - |1\rangle \otimes |0\rangle)$ is not separable. Such an inseparable state is called *entangled*. The correlation between two entangled particles is irrespective of distance, so if Alice generates an entangled state:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

she can send one of the qubits to Bob. If Alice makes a measurement on her qubit, whatever the result of her measurement, Bob will obtain the same result upon measuring his qubit. This fact is used in some quantum key distribution protocols such as the E91 protocol. Additionally, entangled states play a crucial role in quantum error correction as a means to work around the constraints imposed by the no-cloning theorem.

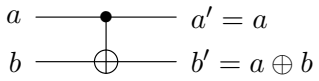
2.2.4.1 Entanglement Generation

In general, the evolution of a quantum state from one state to another is described by a unitary operator (i.e. an operator $U : H \to H$ such that $UU^\dagger = U^\dagger U = I$). The action of quantum gates is thus also described by unitary operators. An important example of a quantum gate is the CNOT gate. This gate takes two-qubits as inputs and flips the second qubit (called the target bit) when the first (called the control) is $|1\rangle$ but does nothing to the first when the first qubit is $|0\rangle$. If the first input bit is in state $|a\rangle$ and the second input bit in state $|b\rangle$ then the respective outputs $|a'\rangle$, $|b'\rangle$ are described in table 2.1. It can also be useful to describe this gate using a circuit diagram as in figure 2.1 where closed dots represent taking a qubit as an input, open dots represent a "not" operation and the symbol "$\oplus$" is addition modulo 2.

Table 2.1 CNOT gate

$\|a\rangle$	$\|b\rangle$	$\|a'\rangle$	$\|b'\rangle$
$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$
$\|0\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$

Figure 2.1 CNOT



Here $\oplus$ is addition mod 2

This gate is described by the unitary operator:

$$U_{CNOT} = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10|$$

Or, as a matrix:

$$U_{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

This gate can be used to generate entangled qubits. Consider the state $|\psi\rangle = a|0\rangle + b|1\rangle$. If $|0\rangle$ is input as the control qubit and $|\psi\rangle$ is input as the target qubit:

$$U_{CNOT}|\psi\rangle|0\rangle = a|00\rangle\langle 00|00\rangle + b|11\rangle\langle 10|10\rangle = a|00\rangle + b|11\rangle$$

Note that $a|00\rangle + b|11\rangle$ is inseperable.

2.2.4.2 Quantum Parallelism

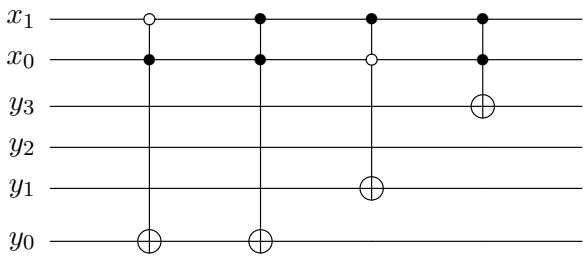
Quantum computers offer capabilities beyond the scope of traditional computers. One of the most powerful capabilities of a quantum computer arises from the fact that qubits in a quantum computer are in a superpostition of states. This linear property allows for a phenomenon known as quantum parallelism in which multiple computations are performed simultaneously.

Say, for example, that Alice uses to two qubits to encode numbers in binary: $|00\rangle \rightarrow 0, |01\rangle \rightarrow 1, |10\rangle \rightarrow 2, |11\rangle \rightarrow 3$. Let U_{x^2} be the unitary operator describing the operation of the circuit 2.2, which takes 2 qubits as input and gives 4 as output. Alice calculates:

$$U_{x^2}\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)|0000\rangle = \frac{1}{2}(|00\rangle|0000\rangle + |01\rangle|0001\rangle + |10\rangle|0100\rangle + |11\rangle|1001\rangle)$$

With only one computation, Alice is able to calculate $0^2 = 0$, $1^2 = 1$, $2^2 = 4$, $3^2 = 9$. There is one catch with this calculation, however: Alice generated a state containing the answers to $0^2, 1^2, 2^2, 3^2$, but she cannot see what these answers are without measuring the state, and when she does, the state will collapse and give only one output (so she effectively losses any information about the other measurement possibilities). The challenge with using quantum parallelism effectively is extracting useful information from the output of a computation.

Figure 2.2 x^2 on two qubits



CHAPTER 3. SHOR'S FACTORIZATION ALGORITHM

3.1 Introduction

Shor's factorization algorithm efficiently factors an integer N into its prime factors. This is of critical importance in the context of cryptography where algorithms such as RSA use the idea that factoring a large number into prime components is computationally difficult. The algorithm relies on quantum parallelism to find the period of a discrete logarithm.

3.2 Shor's Factorization Algorithm

For a simplified version of the algorithm where the integer to be factored is a product of only two primes, let p and q be (odd) primes and let $N = pq$.

1. Randomly pick $m \in \mathbb{Z}^+$, $(m < N)$ and calculate $gcd(m, N)$ with the Euclidean algorithm. If $gcd(m, N) \neq 1$ then $m = p$ or $m = q$ and we are done. Otherwise continue.

2. Define $f_N : \mathbb{N} \to \mathbb{N}$ by $a \mapsto m^a \ mod_N$. Find the smallest $P \in \mathbb{N}$ such that $m^P = 1(mod_N)$ (P is called the order or period of f_N).

3. If P is odd, go back to 1 and repeat with a different m until an even P is found, then proceed to 4.

4. Since P is even, $(m^{\frac{P}{2}} - 1)(m^{\frac{P}{2}} + 1) = m^P - 1 \equiv 0(mod_N)$. If $m^{\frac{P}{2}} + 1 = 0(mod_N)$ then $gcd(m^{\frac{P}{2}} - 1, N) = 1$; go back to 1 and try a different m. If $m^{\frac{P}{2}} + 1 \neq 0(mod_N)$ then $m^{\frac{P}{2}} - 1$ contains either p or q. Go to 5.

5. The number $d = gcd(m^{\frac{P}{2}} - 1, N)$ is either p or q and factorization is done.

In this algorithm, all of the steps except for step 2 can be done efficiently on a traditional computer. Step 2 is the only part which requires a quantum computer. This step is covered in more detail in the subsequent sections, but first some information on the quantum Fourier transform must be given.

3.2.1 Quantum Fourier Transform

Discrete Fourier transforms are often used to transform a function into a series of complex sinusoids. It transforms a series of complex components $x_0, x_1, ...x_{N-1}$ into a new complex series $\tilde{x}_0, \tilde{x}_1, ...\tilde{x}_{N-1}$ by:

$$\tilde{x}_i = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i \frac{jk}{N}} x_j$$

This, in effect, transforms the function from its original domain into the frequency domain and can make periodicity in the function more apparent. The quantum Fourier transform is quantum analogue to the discrete Fourier transform.

If $|Reg1\rangle$ is a quantum register with n qubits and $|0\rangle, |1\rangle, ...|2^n - 1\rangle$ is the computational basis for the space H, then the quantum Fourier transform of $|Reg1\rangle$ is defined as a unitary operator F which acts on the jth vector of the basis as:

$$F(|j\rangle) = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2\pi i \frac{jk}{2^n}} |k\rangle$$

. Thus F acts on a state $|\psi\rangle = \sum_j x_j |j\rangle$ as:

$$F|\psi\rangle = \sum_{k=0}^{2^n-1} \tilde{x}_k |k\rangle$$

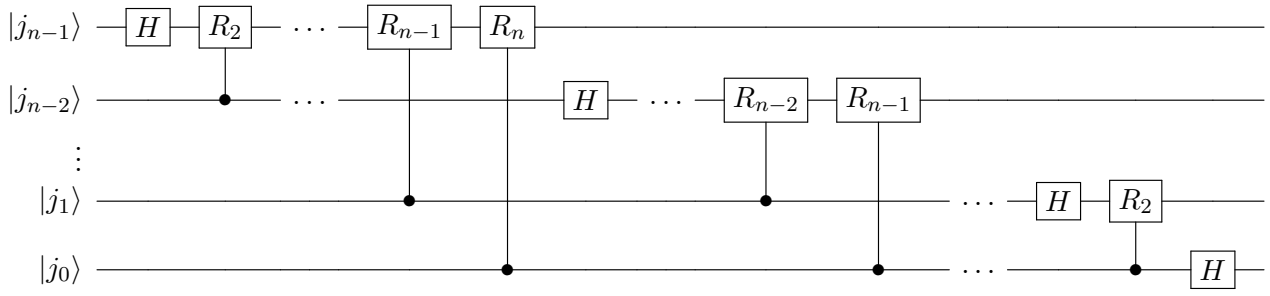
where the coefficients $\tilde{x}_0, \tilde{x}_1, ...\tilde{x}_{2n-1}$ are the discrete Fourier transform of the coefficients $x_0, x_1, ...x_{2n-1}$.

Using two unitary operators,

$$R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^k}} \end{pmatrix} \quad \text{and} \quad H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

the quantum Fourier transform can be implemented in a quantum circuit as in figure 3.1 where the inputs $|j_i\rangle$ are the qubits of the quantum register $|Reg1\rangle$ (for more on this circuit, see **(5)**).

Figure 3.1 Quantum Fourier transform circuit



One important aspect of the quantum Fourier transform is that:

$$F|0\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} |j\rangle$$

That is, the quantum Fourier transform acts on the state $|0\rangle$ to produce a superposition of all of the basis vectors of the space H.

3.2.2 Shor's Algorithm: The Quantum Part

Going back to the second step of Shor's algorithm, N = pq is the integer to be factored with p and q (odd) primes. First, find $n \in \mathbb{N}$ such that $N^2 \leq 2^n < 2N^2$ and let $Q = 2^n$. Restrict $f : a \mapsto m^a \ mod_N$ to $S_n = \{0, 1, 2, \ldots Q-1\}$. The algorithm requires two n-qubit registers, $|Reg1\rangle$ and $|Reg2\rangle$.

1. Set the initial state:

$$|\psi_0\rangle = |Reg1\rangle|Reg2\rangle = \underbrace{|00\ldots0\rangle}_{\text{n qubits}} \underbrace{|00\ldots0\rangle}_{\text{n qubits}}$$

2. Apply a quantum Fourier transform to the first register to gain a superposition of all of the states $|x\rangle$ where $0 \leq x \leq Q-1$:

$$|\psi_1\rangle = F|0\rangle \otimes I|0\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle|0\rangle$$

3. Define $f : S_n \to \mathbb{Z}/N\mathbb{Z}$ by $f(x) = m^x mod\ N$ for all $x \in S_n$. Let U_f be a unitary operation such that $U_f|x\rangle|0\rangle = |x\rangle|f(x)\rangle$. Apply U_f to $|\psi_1\rangle$:

$$|\psi_2\rangle = U_f|\psi_1\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle|f(x)\rangle$$

4. Apply a quantum Fourier transformation on $|Reg1\rangle$:

$$|\psi_3\rangle = (F \otimes I)|\psi_2\rangle = \frac{1}{Q}\sum_{x=0}^{Q-1}\sum_{y=0}^{Q-1} e^{2\pi i \frac{xy}{Q}}|y\rangle|f(x)\rangle$$

5. A measurement of the state $|\psi_3\rangle$ will yield a state $|y\rangle|f(x)\rangle$ with probability:

$$\left|\frac{1}{Q}\sum_{a:f(a)=f(x)} e^{2\pi i \frac{ay}{Q}}\right|$$

.

Make a measurement of $|\psi\rangle$ and suppose the measurement outcome is $|y\rangle|f(x)\rangle$.

6. Find the continued fraction expansion $[a_0, a_1, ..., a_M]$ of $\frac{y}{Q}$.

7. Let:

   - $p_0 = a_0$ and $q_0 = 1$
   - $p_1 = a_1 p_0 + 1$ and $q_1 = a_1 q_0$
   - $p_i = a_i p_{i-2}$ and $q_i = a_i q_{i-1} + q_{i-2}$ for $2 < i < M$.

8. Find the smallest k $(0 \leq k \leq M)$ such that $\left|\frac{p_k}{q_k} - \frac{y}{Q}\right| \leq \frac{1}{2Q}$.

9. If $\frac{p_k}{q_k}$ is in lowest terms and p_k is relatively prime to q_k, then the period is $P = q_k$.

Shor proved in **(1)** that step 5 will result in a measurement which will successfully give the period (after following the rest of the steps) with probability at least $\frac{\phi(P)}{3P}$ where ϕ is the totient function.

CHAPTER 4. QUANTUM ERROR CORRECTION

4.1 introduction

During the transmission of a bit or a qubit, there is an inherent chance that interference or noise in the channel through which the information is being transmitted will cause errors to occur. Error correction, in general, is a process through which the probability of an error occurring is diminished. This process can be broken into four parts: encoding, transmission, error detection, and error correction. The entire process is dependent on the encoding. Typically, this step duplicates the data being transmitted in some way. For a qubit, this step will be more difficult due to the no-cloning theorem. After the data is transmitted, during the error detection step, the information received and measured. The measurement is then compared to what could be received and based on the encoding, which (if any) errors have occurred are determined. This step is again more difficult in a quantum system due to the fact that measurement changes the state being measured. Finally, in the error correction step, based on which errors are detected, the data is modified based on the encoding to recover a corrected state.

4.2 Classical Error Correction

An example from classical error correction which elucidates the ideas behind error correction is a repetition encoding: say Alice transmit a series of classical bits through a noisy channel to Bob and each bit has a probability p to flip. That is when Alice sends a 1 through the channel, there is a probability p that Bob receives a 0 instead, and similarly when Alice sends a 0 through the channel, there is a probability p that Bob receives a 1. Instead of simply sending each bit though the channel, Alice instead encodes each bit in triplicate: 0 is encoded as 000

Table 4.1 Probabilities for recieved states for sent state 000

Recieved State	Probability
000	$(1-p)^3$
001	$p(1-p)^2$
010	$p(1-p)^2$
011	$p^2(1-p)$
100	$p(1-p)^2$
101	$p^2(1-p)$
110	$p^2(1-p)$
111	p^3

Table 4.2 Probabilities for recieved states for sent state 111

Recieved State	Probability
000	p^3
001	$p^2(1-p)$
010	$p^2(1-p)$
011	$p(1-p)^2$
100	$p^2(1-p)$
101	$p(1-p)^2$
110	$p(1-p)^2$
111	$(1-p)^3$

and 1 is encoded as 111 (these triplicates are called logical bits). Then when transmitting 000 Bob receives states with probabilities as given in table 4.1 and similarly, when transmitting state 111 Bob receives states with probabilities given in table 4.2.

Bob uses a majority vote for error detection. That is, if a logical bit contains more $0's$ than $1's$, Bob determines that the original bit was a 0. Similarly, if a logical bit contains more $1's$ than $0's$, Bob determines that the original bit was a 0. Using this detection scheme, for a single received state, Bob then has probability that he will correctly detect the error (or lack thereof) of the state he receives with probability $p_0 = (1-p)^3 + 3p(1-p)^2 = (1-p)^2(1+2p)$ and incorrectly detect the error with probability $p_1 = 3p^2(1-p) + p^3 = p^2(3-2p)$. For sufficiently small p, $p_0 >> p_1$. For example, if $p = 0.1$, then $p_0 = 0.972$ and $p_1 = 0.028$.

4.3 Bit-Flip Error Correction Code

Quantum error correction aims to do the same thing as classical error correction: encode information in a way in which errors can be detected and corrected. If Alice sends a stream of qubits to Bob, however,any encoding she wants to use is subject to limitations imposed by the No-Cloning theorem. For example, she cannot, in general, employ a method which duplicates a state $|\psi\rangle = a|0\rangle + b|1\rangle$ as $|\psi'\rangle = |\psi\rangle|\psi\rangle|\psi\rangle$. Instead, if information is to be duplicated, Alice will use entanglement.

The first example of a quantum error correcting code is analogous to the classical correction code where bits are encoded in triplicate and a majority vote is taken. Suppose Alice wants to send a qubit $|\psi\rangle = a|0\rangle + b|1\rangle$ to Bob and wants to detect and correct errors of the form:

$$|\psi\rangle \rightarrow |\psi'\rangle = b|0\rangle + a|1\rangle$$

(such an error is called a "bit-flip"). Recall from 2.2.4.1 that a CNOT gate acts on a state $|j0\rangle$ $(j = 0, 1)$ as:

$$U_{CNOT}|j0\rangle = |jj\rangle$$

Alice can use this fact to duplicate information in her state $|\psi\rangle$ by employing two CNOT gates and using two additional (ancillary) qubits. She encodes $|\psi\rangle$ as:

$$|\psi\rangle_L = U_{CNOT2}U_{CNOT1}|\psi\rangle = U_{CNOT2}(a|00\rangle + b|11\rangle) = a|000\rangle + b|111\rangle$$

Here U_{CNOT1} and U_{CNOT2} are the two respective unitary operations. Alice sends $|\psi\rangle_L$ to Bob through a noisy channel. Say the channel has a probability p of producing bit-flip errors. Then the probabilities of Bob receiving a given state are given in table 4.3.

Bob still has the issue of detecting which error has occurred. Bob does not want to measure the state directly as this will change the state. This issue is typically solved by the use of a technique called "Error Syndrome Detection." To do this, Bob employs four CNOT gates and two ancillary qubits of his own. Bob uses the first CNOT gate, taking the first qubit he received from Alice as the control and acting on his own first ancillary qubit as the target. This is followed by using his second CNOT gate with Alice's first ancillary qubit as control and his

Table 4.3 Bit-flip error transmission probabilities

Recieved State	Probability
$a\|000\rangle + b\|111\rangle$	$(1-p)^3$
$a\|001\rangle + b\|110\rangle$	$p(1-p)^2$
$a\|010\rangle + b\|101\rangle$	$p(1-p)^2$
$a\|011\rangle + b\|100\rangle$	$p^2(1-p)$
$a\|100\rangle + b\|011\rangle$	$p(1-p^2)$
$a\|101\rangle + b\|010\rangle$	$p^2(1-p)$
$a\|110\rangle + b\|001\rangle$	$p^2(1-p)$
$a\|111\rangle + b\|000\rangle$	p^3

own first ancillary qubit as the target. Bob does a similar process with his second ancillary qubit, but for the fourth CNOT gate, he takes Alice's second ancillary qubit as the control rather than her first. In this way, Bob will produce states as in table 4.4. Notice that he has produced separable states and can thus make measurements without affecting the qubits he has received from Alice.

Bob can make measurements on his ancillary qubits without affecting the other three qubits. In particular, he measures his qubits to determine which of $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ he has. Note that he does not make any measurement to determine a or b so while Bob can determine if an error has occurred, he gains no other information about the state.

Table 4.4 Bit-flip error after syndrome detection

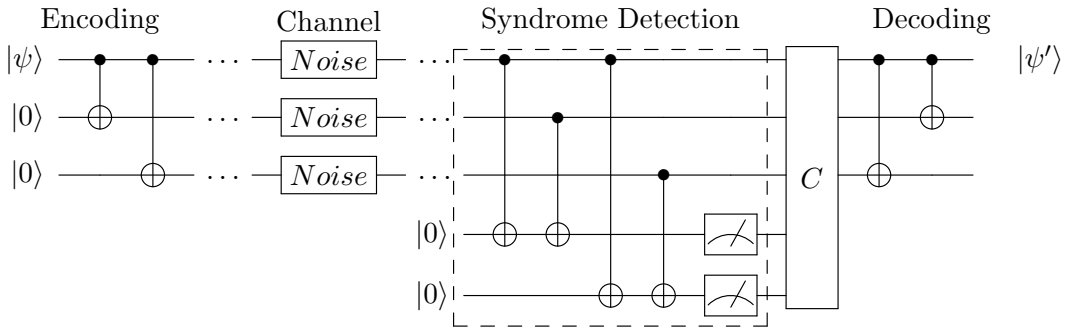
Recieved State	Probability
$(a\|000\rangle + b\|111\rangle)\|00\rangle$	$(1-p)^3$
$(a\|001\rangle + b\|110\rangle)\|01\rangle$	$p(1-p)^2$
$(a\|010\rangle + b\|101\rangle)\|10\rangle$	$p(1-p)^2$
$(a\|011\rangle + b\|100\rangle)\|11\rangle$	$p^2(1-p)$
$(a\|100\rangle + b\|011\rangle)\|11\rangle$	$p(1-p^2)$
$(a\|101\rangle + b\|010\rangle)\|10\rangle$	$p^2(1-p)$
$(a\|110\rangle + b\|001\rangle)\|01\rangle$	$p^2(1-p)$
$(a\|111\rangle + b\|000\rangle)\|00\rangle$	p^3

Let $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. Using the information he obtained from his measurement, Bob determines which correction needs to be made as follows:

- If the error syndrome is $|00\rangle$, do nothing.
- If the error syndrome is $|01\rangle$, apply $I \otimes I \otimes X$ (where I is the two dimensional identity matrix).
- If the error syndrome is $|10\rangle$, apply $I \otimes X \otimes I$.
- If the error syndrome is $|11\rangle$, apply $X \otimes I \otimes I$

Note that Bob does not correct all of the possible errors. Just as in the classical example, he reduces the probability that his final state does not match the state which Alice sent. The circuit diagram is given in figure 4.1.

Figure 4.1 Bit-flip error correction circuit



[1]

4.3.1 Phase-Flip Error Correction Code

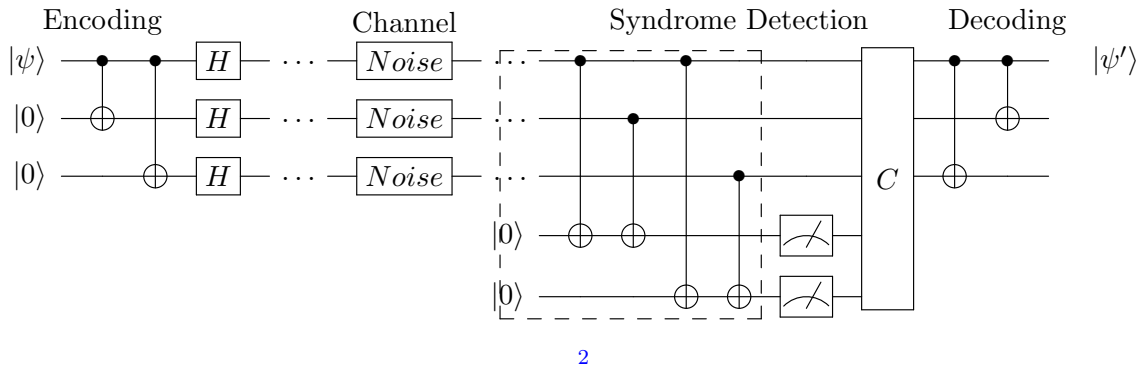
Another error which can occur in a qubit is a phase-flip. This takes $|x\rangle \mapsto (-1)^x|x\rangle$ where $x \in \{0, 1\}$. In particular, this will take:

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \qquad \mapsto \qquad \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle$$

and similarly:

$$|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \qquad \mapsto \qquad \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle$$

Figure 4.2 Phase-flip error correction circuit



This shows that the phase-flip error in the $\{|0\rangle, |1\rangle\}$ basis is bit-flip error in the $\{|+\rangle, |-\rangle\}$ basis. The circuit for this error correction is then as in figure 4.2 where H is the same unitary operation as in 3.2.1 (the unitary operation H is known as the Walsh-Hadamard transform). These and other error correction schemes are described in **(6)**.

(4) showed that the error syndrome detection step is not necessary. It can, in general, be replaced with a unitary operator (which will depend on the particular error code). Not only does this result greatly streamline the error correction process, but it reduces the number of qubits needed for error codes.

Error correction is critical to reliable transmissions. Moreover, since quantum states are not generally stable, error correction is imperative for reliable computation. Additionally, quantum error correction has application to quantum key distribution. For example, in **(2)**, Shor and Preskill used error correction code to prove that the BB84 protocol is secure.

CHAPTER 5. QUANTUM KEY DISTRIBUTION

5.1 Introduction

Symmetric key cryptosystems are used to allow two parties to communicate securely. Both parties share a common secret (the key) which they can use to encrypt information. The challenge with this type of cryptosystem is key distribution. The two parties need a secure means of establishing their shared secret in the first place.

A second type of cryptosystem is the asymmetric key cryptosystem. In this case, a party has a public and private key. The private key is kept secret and can be used to decrypt information encrypted with the public key. This type of cryptosystem can allow for the secure distribution of a symmetric key. For example, in the RSA algorithm, a symmetric key is encrypted with the public key of the other party. Thus both parties can share a symmetric key.

Quantum key distribution creates and distribute symmetric keys. To do this with authentication built into the protocol, Trusted Servers are used to facilitate communication. Quantum key distribution also allows for additional desirable properties which are unique to the quantum domain: quantum key distribution algorithms allow for eavesdropper detection and the security of a quantum key is based on physical properties rather than mathematical properties.

5.2 BB84 Protocol

The BB84 protocol is a foundational example of a quantum key distribution protocol. If Alice wants to generate a key with Bob, using the BB84 protocol:

1. Alice generates a random sequence of 0's and 1's.

2. For each bit, Alice randomly chooses between the $\{|0\rangle, |1\rangle\}$ and $\{|+\rangle, |-\rangle\}$ bases.

   (a) If she chooses the $\{|0\rangle, |1\rangle\}$ basis and the bit is a 0, she prepares a qubit in the $|0\rangle$ state, while if it a 1, she prepares a qubit in the $|1\rangle$ state.

   (b) Similarly, if she chooses the $\{|+\rangle, |-\rangle\}$ basis and the bit is a 0, she prepares a $|+\rangle$ qubit, while if it a 1, she prepares a $|-\rangle$ qubit.

3. Alice sends the string of qubits she has generated to Bob.

4. For each qubit he recieves from Alice, Bob randomly chooses a basis ($\{|0\rangle, |1\rangle\}$ or $\{|+\rangle, |-\rangle\}$) in which to measure the qubit. Note that on average Bob will select the same basis which Alice used to generate a given qubit half of the time.

   (a) If Bob measures a qubit he receives from Alice in the same basis which she used to generate it, the probability (not accounting for errors in transmission or eavesdroppers) that the measurement outcome will give the same state which Alice prepared is 1.

   (b) If Bob measures a qubit he receives from Alice in the opposite basis from the one in which it was prepared, there is still a probability of $\frac{1}{2}$ that the state he measures will correspond to the correct bit which Alice encoded (e.g. if Alice encodes a 0 as $|0\rangle$, and Bob measures this state in the $\{|+\rangle, |-\rangle\}$ basis, there is a probability of $\frac{1}{2}$ that he measures the state as $|+\rangle$ corresponding to 0).

5. Bob communicates to Alice (over a classical channel) in which bases he measured each qubit (but not the measurement outcome).

6. Alice communicates back to Bob in which basis she prepared each qubit (but not the state prepared).

7. Alice and Bob delete all the bits corresponding to cases in which they used different bases. The remaining qubits constitute the "raw key".

8. Alice and Bob then announce and compare a part of their raw key. Using this comparison, they can determine if an eavesdropper (or other noise) is present in their quantum channel by looking at the rate of errors in transmission. If the error rate is too high, they start over

Alice's Bits	1	1	1	0	1	0	0	1	0
Alice's Basis	β	β	α	β	α	α	β	α	β
Transmitted Qubits	$\|-\rangle$	$\|-\rangle$	$\|1\rangle$	$\|+\rangle$	$\|1\rangle$	$\|0\rangle$	$\|+\rangle$	$\|1\rangle$	$\|+\rangle$
Bob's Basis	β	α	α	α	β	α	β	β	β
Bob's Outcomes	$\|-\rangle$	$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|+\rangle$	$\|0\rangle$	$\|1\rangle$	$\|-\rangle$	$\|+\rangle$
Bob's Bits	1	1	1	1	0	0	1	1	0
Raw Key	1		1			0			0

(possibly using a different channel) otherwise they can use error correction and privacy amplification to generate a stronger key from their raw key.

While the BB84 protocol offers eavesdropper detection, it is not immune to man-in-the-middle attacks. This is the term for an attack in which, rather than passively eavesdropping, Eve instead intercepts all communication between Alice and Bob. She then performs the protocol with Alice while masquerading as Bob and does the same with Bob while masquerading as Alice. This allows Eve to establish two keys: one between herself and Alice and one between herself and Bob. To avoid this problem, a protocol needs to authenticate its participants. Conceptually, this is done by having a participant prove that they know something which only they can know. For example, in RSA, only Alice knows Alice's private key, but anyone can send a message to Alice using her public key to encrypt the message. That way, only Alice can decrypt the message, and thus prove she is Alice (or at least has Alice's key). For quantum key distribution, trusted servers are used to facilitate key exchange.

5.3 Three Party Quantum Key Distribution with Authentication

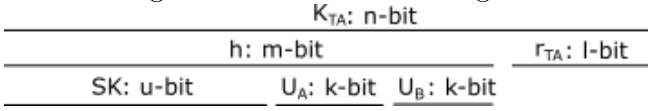
Authentication is a necessary part of any secure protocol. Without it, man-in-the-middle attackers are possible, as was the case with the BB84 protocol. The basic idea behind authentication is that one party holds a secret which either only they know or is only shared with one other party and that by proving they know the secret, they prove their identity. There have been several protocols which offer some form of authentication which relies on a pre-shared secret. **(3)** gives two protocols for quantum key distribution with authentication using a trusted server. The first of their protocols is described in more detail below.

5.3.1 Three Party Quantumm Key Distribution with Implicit Authentication

Trusted server models use a central server (the Trusted Server or TC) to facilitate communication between two or more parties. In such a model, each user shares a secret key with the trusted server which the trusted server can use to authenticate users and set up session keys between parties. Like in the BB84 protocol, the models proposed in **(3)** have eavesdropper detection as a property eliminating the need for time-stamps which are necessary in classical trusted server protocols to prevent a number of attacks.

- Let U_i be the k-bit identity of a participant. Denote Alice's identity as U_A and Bob's as U_B.
- Let $h(.)$ be a one way cryptographic has function mapping $\{0,1\}^* \rightarrow \{0,1\}^m$.
- Let r_{TU} denote an n-bit random string (salt for the hash function) chosen by the TC (where the subscript TU represents this quantity being relevant to the trusted server T and a user U).
- Let K_{TU} be the n-bit secret key (where $n = l + m$) shared between the server and a user U.
- Let SK be the u-bit session key (where $m = u + 2k$) shared between two participants.

Figure 5.1 Relative bit lengths



If Alice and Bob want to generate a session key:

1. TC generates l-bit random number r_{TA} and a u-bit session key SK. TC uses these to compute

$$R_{TA} = h(K_{TA}, r_{TA}) \oplus (SK||U_A||U_B)$$

for Alice and

$$R_{TB} = h(K_{TB}, r_{TB}) \oplus (SK||U_B||U_A)$$

for Bob.

2. TC creates a string of qubits, Q_{TA}, for Alice where the ith qubit is determined based on the ith bits $(r_{TA}||R_{TA})_i$ and $(K_{TA})_i$ of $(r_{TA}||R_{TA})$ and (K_{TA}), respectively, for $i = 1, 2, \ldots, n$.

   - If $(r_{TA}||R_{TA})_i = 0$ and $(K_{TA})_i = 0$, then $(Q_{TA})_i = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.
   - If $(r_{TA}||R_{TA})_i = 1$ and $(K_{TA})_i = 0$, then $(Q_{TA})_i = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.
   - If $(r_{TA}||R_{TA})_i = 0$ and $(K_{TA})_i = 1$, then $(Q_{TA})_i = |0\rangle$.
   - If $(r_{TA}||R_{TA})_i = 1$ and $(K_{TA})_i = 1$, then $(Q_{TA})_i = |1\rangle$.

   Similarly, TC creates Q_{TB} for Bob.

3. TC sends Q_{TA} to Alice and Q_{TB} to Bob.

1. Alice measures Q_{TA} based on K_{TA}.

   - If $(K_{TA})_i = 0$, then the ith qubit of Q_{TA} is measured in the $|+\rangle, |-\rangle$ basis.
   - If $(K_{TA})_i = 0$, then the qubit is measured in the $|0\rangle, |1\rangle$ basis.

   Similarly, Bob measures the qubits of Q_{TB} based on K_{TB}.

2. Alice obtains results $r'_{TA}||R'_{TA}$ from measurement which she uses to compute

$$SK'||U_A||U_B = h(K_{TA}, r'_{TA}) \oplus R'_{TA}$$

   from which the session key SK' can be obtained and U_A, U_B can be verified. Similarly, Bob computes

$$SK''||U_B||U_A = h(K_{TB}, r'_{TB}) \oplus R'_{TB}$$

CHAPTER 6. SUMMARY AND DISCUSSION

6.1 Introduction

Quantum computers have the power to break cryptosystems critical to the security of the internet. Shor's algorithm is capable of breaking the RSA algorithm which is widely used for computer security.

On the other hand, quantum key distribution algorithms offer new cryptographic primitives which do not rely on computationally difficult mathematical problems, but rather physics for its security. These new systems have potential applications which are already being explored.

This chapter looks at some of the technology which is emerging due to quantum information theory as well as some of the holes left by the power which quantum computers will have.

6.2 Emergent Technologies

Technologies employing quantum key distribution are already commercially available. According to **(7)**, as early as 2007, the Swiss government has been using quantum key distribution technology, developed by ID Quantique, to protect its data centers. The United States government is also investing quantum key distribution technology with a network of fiber cables and relay stations from Ohio to DC which will be part of a network connecting the Federal Reserve banking system. Similarly, China has invested in over 2000 km of cable connecting Beijing to Shanghai as part of a quantum network. **(7)** also discusses a quantum router being developed at Los Almos National Lab. The router, called a QKarD, will allow cell phones, computers, and anything else that can connect to the internet to exchange quantum keys through a secure, central server. These technologies are certainly amazing, but they all rely on a central server model, so the threats posed by quantum computers are not fully addressed.

6.3 Security Holes

There are two practical problems with the new key distribution algorithms offered by quantum information theory. The first is that implementation of such systems on a wide scale would require infrastructure change. With technologies like the QKarD, this problem is approachable, however, there is an issue of employing and maintaining the central servers.

The second problem is that, while quantum information theory does offer new key distribution algorithms, they are all symmetric-key cryptosystems. Shor's algorithm breaks asymmetric-key cryptosystems such as RSA. To reconcile these differences would require not only the implementation of new infrastructure, but drastic changes in internet protocols. Additionally, the servers used for authentication would effectively need to replace certificate authorities (or be run by certificate authorities). While quantum key distribution algorithms can offer improved security properties over their classical counterparts, a more tenable solution to the problems which Shor's algorithm incurs would be development of asymmetric-key cryptosystems which are implementable on a traditional computer and which can not be broken by quantum computers. Such cryptosystems (known as quantum resistant cryptosystems) will use new cryptographic primitives. The field of quantum resistant cryptography is currently very active. The National Institute of Standards and Technology list several of these methods in **(8)** including lattice-based cryptography, code-based cryptography, and multivariate polynomial cryptography. Developing and employing quantum resistant protocols to replace the soon to be vulnerable systems is likely to be an enormous challenge, but one which must be undertaken.

BIBLIOGRAPHY

[1] Shor, Peter W. (1996). Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer *http://arxiv.org/abs/quant-ph/9508027*,

[2] Shor, Peter W. & Preskill, John (2000). Simple Proof of Security of the BB84 Quantum Key Distribution Protocol *http://arxiv.org/abs/quant-ph/0003004v2*.

[3] Hwang, Tzonelih., Lee, Kuo-Chang., & Li, Chuan-Ming (2007). Provably Secure Three-Party Authenticated Quantum Key Distribution Protocols *IEEE Transaction on Dependable and Secure Computing*, *4*(1), 71–80.

[4] Li, Chi-Kwong., Nakahara, Mikio., Poon, Yiu-Tung., Sze, Nung-Sing., & Tomita, Hiroyuki (2011). Recovery in Quantum Error Correction for General Noise without Measurement, Quantum Information and Computation, 12, 149-158.

[5] Benenti, Giuliano., Casati, Giulio., & Strini, Giuliano (2004). *Principles of Quantum Computation and Information. Volume I: Basic Concepts*. Signapore: World Scientific Publishing Co. Pte. Ltd.

[6] Nakahara, Mikio & Ohmi,Tetsuo (2008). *Quantum Computing: From Linear Algebra to Physical Realizations*. Boca Raton: Taylor and Francis Group.

[7] Folger, Tim (2016, February 1). Quantum Hack. *Scientific American*, 49-55.

[8] Lily Chen, Lily., Jordan, Stephen., Liu, Yi-Kai., Moody, Dustin., Peralta, Rene., Perlner, Ray., & Smith-Tone, Daniel (2016). Report on Post Quantum Cryptography. *NISTR8105*.