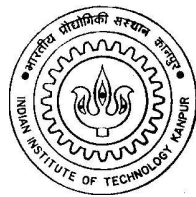


Quantum Cryptography (CHM 696 - Quantum Computing)

A Term Paper Report

Submitted by

D. V. Janardhan Rao
Sagarmoy Dutta
Abhkas Pratam Sakhiya



Department of Computer Science & Engineering
Indian Institute of Technology, Kanpur

March, 2005

Chapter 1

Overview of Cryptography - Quantum and Classical

1.1 Abstract

Quantum cryptography could well be the first application of quantum mechanics at the individual quanta level. The very fast progress in this area and its effect on both classical cryptography and Public Key cryptosystems is explained in detail with emphasis on open questions and technological issues. The Quantum analogs of the classical concepts like Digital Signatures, Authentication of messages, Key distribution are covered. Uncloneable encryption based on the no cloning theorem is also dealt in detail.

1.2 Introduction

The idea of QC was first proposed only in the 1970s by Wiesner (1983) and by Charles H. Bennett from IBM and Gilles Brassard from Montreal University (1984, 1985). It is certainly not a coincidence that QC and, more generally, quantum information has been developed by a community including many computer scientists and more mathematics oriented young physicists.

1.3 What is Cryptology ?

The science of secure communication is called cryptology derived from Greek word kryptos meaning hidden and logos meaning word. Cryptology embodies cryptography, the art of code-making and cryptanalysis, the art of code-breaking.

Cryptography is the art of rendering a message unintelligible to any unauthorized party. To achieve this goal, an algorithm (also called a cryptosystem or cipher) is used to combine a message with some additional information known

as the key and produce a cryptogram. This technique is known as encryption. For a cryptosystem to be secure, it should be impossible to unlock the cryptogram without the key. In practice, this demand is often softened so that the system is just extremely difficult to crack. The idea is that the message should remain protected at least as long as the information it contains is valuable. Although confidentiality is the traditional application of cryptography, it is used nowadays to achieve broader objectives, such as authentication, digital signatures and non-repudiation.

1.4 What are the various aspects of a Cryptosystem ?

The various aspects of a cryptosystem are enforcing policies and mechanisms to achieve,

1. Confidentiality
2. Integrity
3. Availability

Confidentiality : concealment of information or resources from unauthorized persons. It may be important to even conceal existence of data, conventional encryption cannot achieve this. Stegnography is the art of sending data using images.

Integrity : Trustworthiness of information goal is to prevent or detect improper or unauthorized change.

Availability : Ability to use the information or resource by authorized users.

1.5 Policy and Mechanism

A security policy is a statement of what is and what is not allowed . A security mechanism is a method or tool or procedure For enforcing the security policy. Mechanisms need to be technical for example, ID card is needed to get password changed by operator in computer centre. Policies often require procedural mechanisms.

1.6 What are threats and attacks ?

A threat is a potential violation of security. An actual violation of security is called an attack.

1.6.1 Attack classification :

Based on what is intended to be achieved by the attack they are broadly classified into 4 categories. They are,

1. Disclosure unauthorized access to information
2. Deception acceptance of false data
3. Disruption interruption of correct operation
4. Usurpation unauthorized control of some part of the system.

Some attacks will fall in multiple categories.

1.7 Evolution of cryptosystems :

The various cryptosystems evolved with time till now are listed in order,

1. Classical Cryptosystems
2. Public Key Cryptosystems
3. Quantum Cryptosystems

1.7.1 Classical Cryptosystem :

These are also known as conventional cryptosystems or symmetrical (secret key) cryptosystems. Symmetrical ciphers require the use of a single key for both encryption and decryption. These systems can be thought of as a safe, where the message is locked by sender with a key. The receiver in turn uses a copy of this key to unlock the safe.

A General Cryptosystem :

An encryption function E , a decryption function D , a key space K , a plain text space M , and a cipher text space C .

$$E : K \times M \mapsto C$$

$$D : K \times C \mapsto M$$

Assume the adversary knows E and D , then the adversary wishes to find out the plain text for some cipher text or more generally the key being used.

The strength of a classical cryptosystem depends on the length of the key and the computational complexity of the encryption algorithm. The systems in use for routine applications such as e-commerce employ rather short keys. In the case of Data Encryption Standard (also known as DES) a 56 bit key is combined with plain text divided in blocks in a rather complicated way, involving permutations and non-linear functions to produce the cipher text blocks . Other cryptosystems (ex : IDEA, AES, MD5, SHA, Triple DES) follow similar principles.

1.7.2 Public Key Cryptosystems :

These systems involve the use of different keys for encryption and decryption, hence also known as Asymmetrical cryptosystems. Their principle was first proposed in 1976 by Whitfield Diffie and Martin Hellman at Stanford University which is popularly known as Diffie Hellman Key exchange . The first actual implementation was developed by Ronald Rivest, Adi Shamir, Leonard Adleman at MIT in 1978. It is known as RSA and is still widely used.

A public key cryptosystem consists of :

- a key generation algorithm that generates a key pair : a decryption key (the secret key) and the encryption key (the public key).
- an encryption algorithm, taking public key and plain text as input and producing cipher text as output.
- a decryption algorithm, taking a private key and cipher text as input and producing plain text as output.

Satisfying :

Decipherability : for a key pair, the decryption transformation must be inverse of the encryption transformation.

Security : computationally infeasible to compute plain text for given cipher text without knowledge of secret key.

RSA Algorithm :

Key generation

Select p, q (p and q both prime $p \neq q$)

Calculate $n = pq$

Calculate $\Phi(n) = (p-1)(q-1)$ (means the no of numbers less than n which are relatively prime to n)

Select integer e such that $\gcd(\Phi(n), e) = 1$; $1 < e < \Phi(n)$

Calculate $d, d \equiv e^{-1} \pmod{\Phi(n)}$

Public Key : $K_U = (e, n)$

Private Key : $K_R = (d, n)$

Encryption

Plain Text : $M < n$

Cipher Text : $C = M^e \pmod{n}$

Decryption

Cipher Text : C

Plain Text : $M = C^d \pmod{n}$

Security of Public Key Cryptosystems :

The security of public key cryptosystems is based on computational complexity. The idea is to use mathematical objects called one-way functions. By definition it is easy to compute the function $f(x)$ given the variable x , but difficult to reverse the calculation and x from $f(x)$. In the context of computational complexity, the word *difficult* means that the time to do a task grows exponentially with the no of bits in the input, while *easy* means that it grows polynomially. Intuitively, it is easy to understand that it takes only a few seconds to work out 67×71 , but it takes much longer time to find the prime factors of 4757. However, factoring has a trapdoor which means that it is easy to do calculation in the difficult direction provided that you have some additional information. For example, if you were told that 67 was one of the prime factors of 4757, the calculation would be relatively simple. The security of RSA is actually based on the factorization of large integers.

In spite of its elegance suffers from a major flaw. Whether factoring is difficult or not could never be proven. This implies that the existence of a fast algorithm for factorization cannot be ruled out. (The popular work *Primes are in P* by Dr.Manindra Agarwal, Neeraj Kayal, Nitin Saxena, (Dept of computer science, IIT Kanpur) adds fuel to this argument. In addition, the discovery in 1994 by Peter Shor of a polynomial algorithm allowing fast factorization of integers with a quantum computer puts additional doubts on the non-existence of a polynomial algorithm for classical computers).

Similarly, all public-key cryptosystems rely on unproven assumptions for their security, which could themselves be weakened or suppressed by theoretical or practical advances. So far, no one has proved the existence of any one-way function with a trap door. In other words, the existence of secure asymmetric cryptosystem is not proven. This casts an intolerable threat on these cryptosystems. However, for a given key length, symmetrical systems are more secure than their asymmetrical counterparts.

In practical implementations, asymmetrical algorithms are not so much used for encryption, because of their slowness, but used to distribute session keys for symmetrical cryptosystems such as DES. Because the security of those algorithms is not proven, the security of the whole implementation can be compromised. If they were broken by mathematical advances, QC would constitute the only way to solve the key distribution.

Chapter 2

Quantum Digital Signature

2.1 What is digital signature

Suppose you have some documents which you want to send someone else. Your signature on that documents assures the recipient that the document is indeed sent by you. Digital signature is a natural extension of this idea to cryptographic domain. Its importance in e-commerce is unquestionable. Any kind of electronic payment system requires it as the analog of handwritten signature. Here instead of a document you have some message (classical or quantum) coded as a string of bits or qubits. We will discuss only about the classical message here. The signature here is another string of bits or qubits which is called a key. Signing is done by computing a function of the message and the key, that results the signed message. In classical digital signature scheme the key is string of classical bits. Using quantum keys instead of classical leads us to Quantum Digital Signature. The protocol we are going to discuss here is due to Gottesman and Chuang [?]

Before going to full detail some qualitative features of digital signature should be noted that.

The original message should be retrievable from the signed message

There should be some easy way to verify the identity of the sender from the signed message.

In presence of eavesdropping if the recipient Bob infers that message is sent by Alice then the message Bob retrieves should be the same as that was originally sent by Alice. It implies if the retrieved message is not same with the original one the sender identity verification process must not infer that the sender is Alice. This constraint cannot be satisfied deterministically so there should high probabilistic guarantee.

Another subtle point is, if there are multiple recipients say Bob and Charlie, the protocol must not allow Alice to send some signed message such that Bob concludes that Alice has sent and Charlie concludes that Alice has not. Again there can be only probabilistic guarantee.

2.2 Classical Digital Signature

In the classical scheme Alice and Bob share a private key $k \in K$. To send a message $m \in M$ Alice computes $f(m, k)$ where $f : M \times K \mapsto D$ and sends this as signed message to Bob. Suppose Eve is eavesdropping and she changes the signed message to $h(f(m, k))$, where $h : D \mapsto D$ Bob calculates $g(h(f(m, k)), k)$ where $g : D \mapsto M \times \{0, 1\}$. If there is no eavesdropping then h is identity function. If Bob obtains (m', b) it signifies that the retrieved message is m' and $b = 1$ iff the sender is Alice. Then we must have,

$$g(f(m, k), k) = (m, 1)$$

And for any h ,

$$Pr(m = m' \wedge b = 1 \vee m \neq m' \wedge b = 0) \geq 1 - \delta$$

,where δ is a small number which depends on $|m|$ and $|k|$

2.3 Computational Vs Physical limitation

Now why should we at all use a quantum key instead of classical. The answer to this question is the classical scheme relies on computational intractability of some problems whereas quantum scheme gives us the guarantee that the very laws of nature will prevent breaking the security thus making it fullproof.

To make it clear see that for fixed m ; f is one to one function of k otherwise there exists two keys k, k' ($k \neq k'$) such that $f(m, k) = f(m, k') = (m, 1)$. If Bob has k, k' as keys for two different senders he has no way to tell who has sent the message. Hence there exists an inverse function applying which Eve can retrieve k from (m, k) . So f is very carefully chosen that one cannot calculate its inverse easily. Say for example in order to calculate k he has to factor a number which is the product of two large prime numbers. There is no known classical algorithm for doing this in reasonable (I.e. polynomial of input size) time.

But certain quantum algorithm can do such thing in polynomial time, for example Shor's algorithm can do factorisation in polynomial time. So we need an alternative which works without assuming intractability of certain mathematical problem. Quantum digital signature actually does this. The proofs of security is based on impossibility of doing certain things such as measuring a state without disturbing it which by the laws of quantum mechanics can never be done.

2.4 The classical approach : Bit Commitment

It is a general trend to build complex protocols using existing simpler protocols of similar kind. Classically people built digital signature protocol using another concept called bit commitment. In bit commitment Alice sends Bob an encrypted

bit which Bob cannot verify unless Alice provide him a key. But once Alice has sent the encrypted bit she cannot make him conclude (by sending a cleverly chosen wrong key) that the bit sent is \bar{b} , where it is actually b .

Example : Suppose Alice writes the bit 1 in a paper and send it to Bob in a locked box. When she sends the key of the box Bob can see what the bit was, and at the same time she can do nothing to convince Alice that she sent a 0.

2.4.1 Failure of bit-comitment in quantum cryptography

Initially people thought that there should be fully secure bit comitment protocol for quantum information. But later it was proved to be impossible as staed below.

Theorem - For any protocol which prevents Bob from reading the encrypted bit whitout the proper key will allow Alice to alter the bit (by sending some wrong key) without Bob knowing that.

It forces us to look for another line of attack. We notice that bit-comitment is somewhat more secure version of signaturing a single bit. As in the former case we can allow the possibility of recovery of the bit without the key. In [?] authors have picked a classical algorithm by Lamport [?] and came up with a quantum version of it. So we first present Lamport's algorithm.

2.4.2 One-way function

Before discussing lamport's algorithm we would like to remind the property of the previously discussed signing function that the function itself can be easily computed but its inversed cannot be easily computed. Such functions are known as one way function. Later this idea will be extended to quantum way function.

2.5 Lamports protocol

Here Alice chooses two random number k_0 and k_1 .

Alice announces a one-way function f , $(0, f(k_0))$ and $(1, f(k_1))$ publicly so that Bob can have it.

Now if Alice wants to send a bit b she sends (b, k_b) as the signed message.

If Bob receives the signed message as (b', k) he computes $f(k)$. If $f(k) = f(k_{b'})$ he concludes that Alice has sent b' otherwise it rejects message.

If Alice want to send another bit she have to start all over again i.e. have to choose new pair of keys and repeat the process.

2.5.1 One Time Pad

Note that here the keys are discarded after one bit transfer, we cannot reuse them. There are classical algorithms which reuses the key. However there is an upper bound on how many times it is secure to use the same key which depends on the size of the key. It is a well known result of classical cryptography that given sufficiently large encrypted data to Eve she can always retrieve the key with high probability. So the most secure scheme is to use a key only once. This is known as one time pad. Lamports algorithm is an example of that.

2.6 Quantum one way-function

In order to extend Lamport's algorithm to the quantum situation we need quantum one way function which given a classical key will provide us a quantum state to act as a quantum key. These functions should be easily computable and their inverse should not be computable. In addition to that as a requirement of the quantum protocol we have to compare two keys for their equality. But as we shall show now due to no cloning theorem equality test of arbitrary states are not possible.

Theorem - If $|\psi_1\rangle$ and $|\psi_2\rangle$ are two arbitrary quantum states there does not exist any measurement which can check for their equality.

Proof : Suppose a unitary transformation U exists such that $U|\psi_1\rangle|\psi_2\rangle$ and $U|\psi_1\rangle|\psi_1\rangle$ are distinguishable. We know that in order to be distinguishable two states must be orthogonal (See Nielsen and Chuang Page - 87 for a proof). Therefore we can take them as computational basis $|0\rangle$ and $|1\rangle$. Now applying fanout gate we can clone the outcome of U . Since U^{-1} is also a unitary transformation we can apply them to all these copies to get several copies of $|\psi_1\rangle$ and $|\psi_2\rangle$. According to no cloning theorem it is possible if and only if $|\psi_1\rangle$ and $|\psi_2\rangle$ are orthogonal. Hence such U cannot exist.

Now the problem is in a n qubit system there can only be n orthonormal states. So if the classical keys are k bits long then $n = 2^k$. To avoid this we introduce the notion of near orthogonality.

2.6.1 Near Orthogonality

Definition : Two states $|\psi_1\rangle$ and $|\psi_2\rangle$ are near orthogonal if $|\langle\psi_1|\psi_2\rangle| \leq \delta$ for some constant $\delta < 1$.

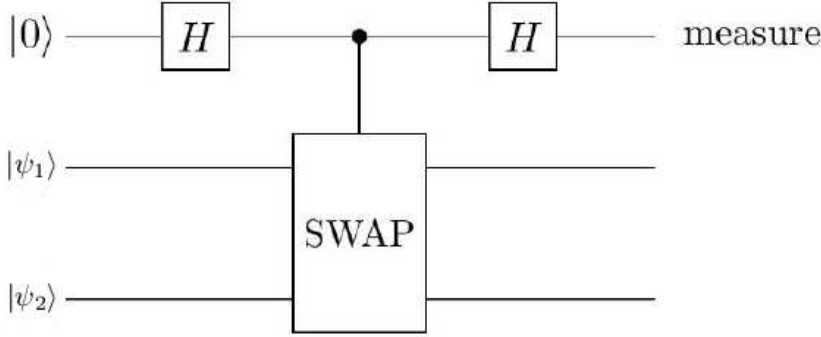
Note that if $\delta = 0$ then near orthogonality is same as orthogonality. If we allow $\delta > 0$ we can have much larger set of nearly orthogonal states.

Example : For $\delta = \cos(\pi/2^k)$ take the set of near orthogonal states as $K = \{\cos(\pi j/2^k)|0\rangle + \sin(\pi j/2^k)|1\rangle |j \in N\}$

There are other examples also. But in this protocol we don't have to worry about the exact form of the function. We can pick up any of them.

2.7 Verifying near orthogonality : Swap Test

Now how does near orthogonality helps us to verify the equality of key. Consider the following circuit. Here C-Swap is the controlled swap gate which performs the transformation $|0\rangle |\psi_1\rangle |\psi_2\rangle \mapsto |0\rangle |\psi_1\rangle |\psi_2\rangle$, $|1\rangle |\psi_1\rangle |\psi_2\rangle \mapsto |1\rangle |\psi_2\rangle |\psi_1\rangle$.



Claim : The measurement is always 0 if $|\psi_1\rangle = |\psi_2\rangle$. If $|\langle \psi_1 | \psi_2 \rangle| \leq \delta$ measurement is 0 with probability at most $(1 + \delta^2)/2$.

Proof : The whole transformation can be expressed as $U = (H \otimes I)(C - Swap)(H \otimes I)$. So the final state is,

$$\begin{aligned}
 \psi &= (H \otimes I)CNOT(H \otimes I) |0\rangle |\psi_1\rangle |\psi_2\rangle \\
 &= \frac{1}{\sqrt{2}}(H \otimes I)(C - Swap)(|0\rangle |\psi_1\rangle |\psi_2\rangle + |1\rangle |\psi_1\rangle |\psi_2\rangle) \\
 &= \frac{1}{\sqrt{2}}(H \otimes I)(|0\rangle |\psi_1\rangle |\psi_2\rangle + |1\rangle |\psi_2\rangle |\psi_1\rangle) \\
 &= \frac{1}{2}((|0\rangle + |1\rangle) \otimes |\psi_1\rangle |\psi_2\rangle + (|0\rangle - |1\rangle) \otimes |\psi_2\rangle |\psi_1\rangle) \\
 &= |0\rangle \otimes (|\psi_1\rangle |\psi_2\rangle + |\psi_2\rangle |\psi_1\rangle)/2 + |1\rangle \otimes (|\psi_1\rangle |\psi_2\rangle - |\psi_2\rangle |\psi_1\rangle)/2
 \end{aligned}$$

Now probability of obtaining 0 is $|\langle 0 | \psi \rangle|^2 = (1 + \langle \psi_1 | \psi_2 \rangle^2)/2$ which is 1 if $|\psi_1\rangle = |\psi_2\rangle$, and at most $(1 + \delta^2)/2$ if $|\langle \psi_1 | \psi_2 \rangle| \leq \delta$.

This test is called swap test and introduced in in the context of quantum digital fingerprinting [?].

2.8 Verifying the key

Like Lamport's protocol given the signed message as $(b, |f(b)\rangle)$ and the classical bitstring k_b we have to verify indeed k_b maps to $|f(k_b)\rangle$. Let $|f|$ denote the number of qubits in $|f(k_b)\rangle$ and $|0_{|f|}\rangle = |0\rangle^{\otimes |f|}$. Then we can express $|f(k_b)\rangle$ as a unitary transform f such that $f |k_b\rangle |0_{|f|}\rangle = |k_b\rangle |f(k_b)\rangle$. f^{-1} is also a unitary transform so given k_b and $|f(k_b)\rangle$ the reciever can prepare $|k_b\rangle |f(k_b)\rangle$ and apply

f^{-1} on that if the key is right measuring the output should always give the classical bit string $0^p k_b$.

If the key is substituted by Eve with another key $|\psi\rangle$ then the second slot will give non-zero on measurement with probability $1 - |\langle\psi|f(k_b)\rangle|^2$. Which combined with the constraint of near orthogonality reduces to $1 - \delta^2$.

Note that given both $|k\rangle$ and $|f(k)\rangle$ one can get $|k\rangle|0_p\rangle$ but in general that does not assure the existence of function g which can map $|f(k)\rangle|0_{|k|-|f|}\rangle$ to $|k\rangle$. f is chosen such that g does not exist.

2.9 Problems of naive protocol - Holevo's theorem

We will consider the case of multiple recipients. The naive extension of Lamport's algorithm would be to choose k_0 and k_1 , send a bit b along with $|f(k_b)\rangle$ and then announce k_0, k_1 publicly such that the receivers can verify it using the technique just described. But there are certain issues which are unique to the quantum version of the problem.

Unlike classical keys quantum keys cannot be duplicated arbitrarily, due to no cloning theorem. So if there are T recipients, the sender has to make T copies of $f(k)$.

If some or all copies fall into the hand of an eavesdropper he can extract some information about k . So the protocol has to be designed in such a way that even if all the copies are available no one can get much information about k . But to do that we need to know what is the theoretical upper bound on the classical information that can be extracted from an n -qubit state. A theorem due to Holevo tells us that it can be at most n bits.

We should have a scheme to distribute the keys and make a provision for the recipients to compare their keys among themselves. And that is where a swap test will be used. A detailed key distribution scheme can be found in [?].

Since both swap test and key verification are probabilistic we should make the probability high enough. For that instead of choosing one pair k_0, k_1 , M such pairs denoted by $k_b^1, k_b^2, \dots, k_b^M$ will be chosen where $b = 0, 1$. Here M is a security parameter.

At the same time it turns out that the recipient cannot simply accept or reject the key unconditionally. There will be two kinds of acceptance, 1-ACC means message is accepted and can be transferred to others, 0-ACC means message is accepted but cannot be transferred to others. REJ means message is rejected.

2.10 The protocol

We are now ready to go into the protocol. It is given bellow.

1. Alice sends the signed message $(b, k_b^1, k_b^2, \dots, k_b^M)$ over an insecure classical channel.
2. Each recipient checks whether $k_b^i \mapsto |f(k_b^i)| >$ for all $1 \leq i \leq M$ using the quantum key verification method. Let the number of incorrect key counted by recipient j is s_j
3. Recipient j concludes the acceptance status is 1-ACC if $s_j \leq c_1M$, REJ if $s_j \geq c_2M$. If $c_1M < s_j < c_2M$ then the status is 0-ACC. Here c_1 and c_2 are some constants such that $0 \leq c_1 < c_2 \leq 1$.
4. Discard all used and unused keys.

This procol ensures following claim.

Claim : If $c_2M < (1 - \delta^2)(M - 2^{-(k-Tn)}(2M))$ then either each receives the correct message or rejects with high probability, where T is number of recipients, k is length of each classical keys that is k_b^i s, n is the number of qubits in quantum keys and δ is the near orthogonality parameter of quantum keys.

The probability with which the sender can cheat that he makes two recipients disagree about the acceptance of a message depends on the quantity $c_2 - c_1$. The probability decreases with the difference and can be made very small choosing c_1 suitably. The proof of this statement is too involved to accommodate in this paper. Interested readers can see the reference. However we give the proof of claim 1 which is quite simple to understand.

2.11 Proof of Claim1

If all T copies of each key is available to the evesdropper, as stated in the Holevo's theorem, he can extract at most Tn bit of classical information per classical key k_b^i . Given Tn bits of k_b^i rest of the $k-Tn$ bits (k_b^i contains k bits) have to guessed with success probability $1/2$ (either 0 or 1) for each bit. Thus guessing a key correctly occurs with probability $2^{-(k-Tn)}$. So expected number of correctly guessed keys is $2^{-(k-Tn)}(2M)$ since there are total $2M$ number of keys (M keys for 0 and 1 each). So if the evesdropper wants to falsly sign a message the expected number wrong keys that are used by him is $M - 2^{-(k-Tn)}(2M)$. From the verification test we know that for a wrong key the probability that it will be discovered by the test is $1 - \delta^2$. So the expected number of incorrect keys counted by the recipient is $(1 - \delta^2)(M - 2^{-(k-Tn)}(2M))$.

Chapter 3

Quantum Authentication

3.1 What is Quantum Authentication

So far we have seen how to secure the classical information using quantum key. We can also talk of all sorts of cryptographic tasks when the information itself is quantum. The key can be classical or quantum. The protocol we are going to discuss ([?]) uses classical key. The advantage of classical key over quantum is, they are easy to handle and in this case they suffices to ensure unconditional security.

The difference between signature and authentication is that in signature if the message is accepted it is necessary to retrieve the identity of the sender. Authentication is more simple as we only need to ensure that if the message is accepted then it is valid. We will give the formal definition of quantum authentication later. However it is almost exactly same as described in the classical digital signature section. Only the functions used to encode or decode information, in stead of mapping bit strings to bit strings, maps a quantum state to another quantum state.

3.2 Formal Definition

Quantum authentication was first formally defined in [?]. We present them in exactly the same words as given in the paper and suppliment it with the motivation and clarification. **Definition:** A quantum authentication scheme (QAS) is a pair of polynomial time quantum algorithms A and B together with a set of classical keys K such that,

1. A takes as input an m -qubit message system M and a key $k \in K$ and outputs a transmitted system T of $m + t$ qubits.
2. B takes as input the (possibly altered) transmitted system T' and a classical key $k \in K$ and outputs two systems: a m -qubit message state M ,

and a single qubit V which indicates acceptance or rejection. The classical basis states of V are called $|ACC\rangle, |REJ\rangle$ by convention.

Comparing to the classical digital signature section we can see that A and B correspond to f and g . So they have similar properties. Given $|\psi\rangle \in M$ as input to A , B should produce either $|\psi\rangle |ACC\rangle$ or if the message is tampered so that $|\psi\rangle$ can not be retrieved it should produce $|\psi'_k\rangle |REJ\rangle$. More over in absense of evesdropping it should always accept and retrieve correctly. This property is called *completeness*. From now on we will use density matrix $|\psi\rangle \langle\psi|$ to represent a quantum state, in stead of ket vector $|\psi\rangle$. So the definition of completeness is given by,

Completeness : For a state $|\psi\rangle$ and for all keys $k \in K$, $B_k(A_k(|\psi\rangle \langle\psi|)) = |\psi\rangle \langle\psi| \otimes |ACC\rangle \langle ACC|$

Here $A_k(\cdot)$ and $B_k(\cdot)$ means $A(k, \cdot)$ and $B(k, \cdot)$ respectively.

Now like h in the classical case assume there are some evesdropping which is represented by another quantum function O . We don't want B to make mistakes even in presence of O . Let us design an operator $P^{|\psi\rangle}$ which return 0 if B makes a mistake and 1 if it does not. $P^{|\psi\rangle}$ represented by a set $\{P_0^{|\psi\rangle}, P_1^{|\psi\rangle}\}$ of two projectors. The projectors are given explicitly by,

$$\begin{aligned} P_1^{|\psi\rangle} &= |\psi\rangle \langle\psi| \otimes |ACC\rangle \langle ACC| + I_M \otimes |REJ\rangle \langle REJ| \\ P_0^{|\psi\rangle} &= (I_M - |\psi\rangle \langle\psi|) \otimes (|ACC\rangle \langle ACC|) \end{aligned}$$

Where I_V and I_M are identity matrices of spaces V and M respectively.

If B_k outputs $|\psi'_k\rangle$ and $P^{|\psi\rangle}$ results in 1 then,

$$\begin{aligned} P_1^{|\psi\rangle} |\psi'_k\rangle &= |\psi'_k\rangle \\ \text{Tr} \left(P_0^{|\psi\rangle} |\psi'_k\rangle \langle\psi| \right) &= \text{Tr} (|\psi'_k\rangle \langle\psi|) \\ &= 1 \end{aligned}$$

Otherwise if P returns 0, in that case

$$\begin{aligned} P_1^{|\psi\rangle} |\psi'_k\rangle &= 0 \\ \text{Tr} \left(P_0^{|\psi\rangle} |\psi'_k\rangle \langle\psi'_k| \right) &= 0 \end{aligned}$$

If we choose all values of $k \in K$ then the total number of not making mistakes by B_k is given by,

$$\sum_{k \in K} \text{Tr} \left(P_1^{|\psi\rangle} |\psi'_k\rangle \langle\psi| \right) = \text{Tr} \left(P_1^{|\psi\rangle} \sum_{k \in K} |\psi'_k\rangle \langle\psi'_k| \right)$$

So we have,

$$\Pr\{B_k \text{ does not make mistake}\} = \frac{1}{|K|} \text{Tr} \left(P_1^{|\psi\rangle} \sum_{k \in K} |\psi'_k\rangle \langle\psi'_k| \right)$$

We would like to bound this probability above a constant value near to 1 say $1 - \epsilon$. That leads us to the definition of another property *Soundness*.

Soundness : For a state $|\psi\rangle$, for all keys $k \in K$ and for all super operator O let ρ_{Bob} be the state output by Bob when the adversary's intervention is characterized by O the soundness property with error ϵ holds if,

$$\text{Tr} \left(P_1^{|\psi\rangle} \left(\frac{1}{|K|} \sum_{k \in K} B_k(O(A_k(|\psi\rangle \langle\psi|))) \right) \right) \geq 1 - \epsilon$$

We now finally arrive at the definition of secure quantum authentication scheme.

Definition A QAS is secure with error ϵ for a state $|\psi\rangle$ if it satisfies completeness for $|\psi\rangle$ and soundness for $|\psi\rangle$ with error ϵ .

A QAS is secure with error ϵ if it is secure with error ϵ for all states.

3.3 Quantum Error Correcting Code

The protocol uses Quantum Error Correcting Code, more precisely a particular class of error correcting code known as *Stabilizer Purity Testing Code*. The theory of QECC or stabilizer formalism is a vast field of study in itself ([?] has a nice discussion on them). In the next section we will just illustrate a simple example of that.

We can assume the act of a potential evesdropper as some channel noise. It is expressed as a unitary transform which may or may not apply to a qubit with some probability p and $1 - p$ respectively. In our example the noise is of particular kind that it flips the qubits i.e. maps $|0\rangle \mapsto |1\rangle$ and vice versa. Such channels are called bit-flip channel. Further assume that the error rate is such that in three consecutive qubits no more than one qubit is flipped.

To cope up with the noise we will use $|000\rangle \equiv |0_L\rangle$ and $|111\rangle \equiv |1_L\rangle$ to represent 0 and 1 respectively instead of using just $|0\rangle$ and $|1\rangle$. It can be viewed as a map $f : D \mapsto C$ where D is your original data space and C is the code space. f is known as coding function. In our example D the two dimensional hilbert space and $C = D^{\otimes 3}$.

Due to noise a $|0_L\rangle$ may be changed to $|100\rangle$ or $|010\rangle$ or $|001\rangle$ and a $|1_L\rangle$ may be changed to $|011\rangle$ or $|101\rangle$ or $|110\rangle$. Now consider the measurement $P = P_0, P_1, P_2, P_3$ which tells us on which bit position the error has accured. The result of the measurement is known as **syndrome**.

$P_0 = 000\rangle \langle 000 + 111\rangle \langle 111 $	no error
$P_1 = 100\rangle \langle 100 + 011\rangle \langle 011 $	error in qubit 1
$P_2 = 010\rangle \langle 010 + 101\rangle \langle 101 $	error in qubit 2
$P_3 = 001\rangle \langle 001 + 110\rangle \langle 110 $	error in qubit 3

Note that the measurement P does not destroy the coded data. So knowing the syndrome we can modify them to get back the error free code. Then we can apply f^{-1} to obtain correct decoded data.

Here we have talked of only bit-flip noise but coding scheme for sufficiently general type of noise also exist. Sometimes it is only needed to detect whether an error has accrued but not necessarily correct them. The code used for this purpose are base on similar principles and known as error detection code. In the theory of quantum error correcting code there is a general framework called *stabilizer formalism* which can be used to mechanically obtain a family of QECC. They are called *stabilizer code*. What is used in this protocol is a class stabilizer codes for detecting error in ERP states. It is called *stabilizer purity testing code*.

3.4 The protocol

A protocol can be *Interactive* or *Non-interactive*. The later is obviously easy to implement. In [?] authors have first designed a interactive protocol based on quantum teleportation. And then transformed it preserving the security through two intermediate forms to finally get a non-interacting protocol. Here we only present the final protocol.

1. Alice and Bob agree on some stabilizer purity testing code $\{Q_k\}$ and some private and random classical keys x, y and z .
2. Alice encrypts ρ as τ using key x . Alice encodes τ according to code Q_k for the code Q_k with syndrome y to produce σ . Alice sends the result to Bob.
3. Bob receives n qubits. Denote the received state by σ' . Bob measures the syndrome y' of the code Q_k on his qubits. Bob compares y to y' and abort if error is detected. Bob decodes his n -qubit words according to Q_k obtaining τ' . Bob decrypts τ' using x and obtains ρ .

Authors of [?] have given a particular stabilizer purity testing code. If that one is used in the protocol the following theorem holds.

Theorem *With key length $2m + s + \log_2(2^s + 1)$ the soundness error of the protocol is $2n / [s(2^s + 1)]$, where m is the length of the message in qubit and Alice sends a total of $n = m + s$ qubits.*

3.5 Two Important Results

In course of proving the security of the protocol in [?] the authors also prove two very important results which we would like to mention

3.5.1 Good Authentication Implies Encryption

From definition authentication and encryption are two independent tasks. In encryption one has to take care that the evesdropper should not read the message, but in authentication no matter the message is visible to Eve or not the only guarantee that should be provided that eve cannot change the message so that Bob accepts a wrong message. There are separate classical algorithms for both and it is not necessary to encrypt the message in order to ensure authentication.

But in quantum authentication whichever protocol is used, it turns out that if the message can be read by Eve she can always change the encoded message such that Bob will accept it after decoding and the decoded message will be different from the original. That means in quantum situation authentication implies encryption.

3.5.2 Impossibility Of Signing Quantum Information

Since encryption is necessary for authentication digitally signing quantum information is impossible. To understand why let us assume such a scheme exists. If there are two receivers, Bob and Charlie, and Alice sends a message to Bob, he can modify the content and send it to Charlie so that Charlie falsely conclude that the message has been sent by Alice.

Chapter 4

Uncloneable Encryption

4.1 What is Uncloneable Encryption

Quantum states cannot be cloned. This important property is extended to classical messages encoded using quantum states which is called uncloneable encryption. An uncloneable encryption scheme has the property that an eavesdropper Eve not only cannot read the encrypted message, but she cannot copy it for later decoding. She could steal it, but then the receiver Bob would not receive the message, and would thus be alerted that something was a miss. Any authentication scheme for quantum states acts as a secure uncloneable encryption. Quantum encryption is also closely related to quantum key distribution, demonstrating a close connection between cryptographic tasks for quantum states and for classical messages. Uncloneable encryption remains secure with a pseudorandom key. In this case to defeat the scheme, Eve must break the computational assumption behind the pseudorandom sequence before Bob receives the message, or her opportunity is lost. So, Uncloneable encryption can be used in a non-interactive setting, where quantum key distribution is not available, allowing Alice and Bob to convert a temporary computational assumption into a permanently secure message.

One problem with computationally secure encryption is that Eve can simply copy down the encrypted message and then try to break it at her leisure. For instance, she might apply brute-force methods, running many computers for a long time, or wait for faster or better computers. For example, a quantum computer could break many of today's standard codes. Or Eve might wait for a breakthrough in cryptanalysis techniques which might make breaking the code easy. Alternatively, Eve might try to steal Bob's key or hope that he will carelessly leave it unguarded. Even the one-time pad is vulnerable to this last problem, meaning it is critical to utterly destroy a one-time pad key after using it.

However, quantum states have an interesting property: they cannot be copied (a result known as the "no-cloning theorem"). It turns out to be possible

to give this property to encrypted classical messages, creating something known as uncloneable encryption. An unencrypted classical message can always be copied, so there is no such thing as quantum copy protection; indeed, reading a message (or watching it, listening to it, etc.) is in a very real sense making a copy. However, uncloneable encryption is possible, meaning the eavesdropper Eve, who cannot read the message, can also not copy it for later decoding. Thus, information protected by an uncloneable encryption scheme remains secure even if the scheme itself later becomes unreliable – for instance, if Eve learns how to solve the computational problem it relies on, or even if she steals Bob’s decryption key.

One way to perform uncloneable encryption is to use a quantum authentication scheme. A quantum authentication scheme is normally used to protect quantum information from being changed, but could also protect classical information. Not only must a quantum authentication scheme encrypt the information it protects, but it also provides uncloneable encryption.

An uncloneable encryption scheme, in turn, can be used to perform quantum key distribution, that is, to generate new classical secret key over an insecure quantum channel. Alice simply chooses a random number r and uses this as a key for an uncloneable encryption scheme, with which she transmits another random number k to Bob. At this point, neither Eve nor Bob knows r , so neither can learn k . Bob lets Alice know when he has received the transmitted message, however, and by the uncloneability property, Alice and Bob know that Eve cannot have copied the message in transmission. It is therefore safe for Alice to publicly announce r . Eve has lost her chance at decrypting the message: she cannot learn k , but Bob can. Alice and Bob therefore both know the random number k , but Eve does not, so k becomes their new secret key.

4.2 Uncloneable Encryption and Quantum Authentication

Assume that Alice and Bob share a secret classical key $k \in K$ which they will use to send just one message. If Alice wants to send a classical message m to Bob, she will use some encoding that depends on k ; in general this could be a message state $\phi_k(m)$. In order for this to be a good encryption scheme, the transmitted density matrix, averaged over possible values of the key, should not depend on the message.

Definition : Let $\phi(m) = (\sum_k \phi_k(m))/|K|$. Then $\phi_k(m)$ is an (unconditionally secure) encryption scheme with error ϵ if the trace distance $D(\phi(m), \phi(m')) = 1/2 \text{Tr}|\phi(m) - \phi(m')| \leq \epsilon$ for $m \neq m'$.

That is someone who does not know the key has essentially no information about the message. The above definition only addresses the secrecy of the message; for a useful protocol, we also require that someone who does know the key is able to read the message.

In order to have an uncloneable encryption scheme, we need an additional

condition. A general attack by Eve is a super operator l acting on $\phi_k(m)$. This represents the action Eve performs when she first gets the encrypted state before she learns the key, so l cannot depend on k . The output of l can be divided into two parts, a density matrix $\phi_{Bob,k}(m)$ which is sent on to Bob and the remainder which is kept by Eve.

To take the next step, we must assume Bob has some efficient check procedure $\phi_{Bob,k}(m, k) \mapsto \{ACC, REJ\}$ which allows him to detect Eve's tampering. They may need to protect the key k especially well, for instance, or act to neutralize any damage caused if Eve learns m . We let $\rho_k(m)$ be Eve's residual density matrix conditioned on the case that Bob gets outcome ACC, and let $P_k(m)$ be the probability that Bob accepts the message m . In general, $\rho_k(m)$ and $P_k(m)$ can depend on the attack l .

Definition An encryption scheme $\phi_k(m)$ with error ϵ is an uncloneable encryption scheme with error ϵ if, for any two messages $m \neq m'$ and all attacks l by Eve, for fraction of at least $1 - \epsilon$ of the possible values of the key k , the trace distance $D(P(m)\rho_k(m), P(m')\rho_k(m')) \leq \epsilon$

From the above definition one can easily prove two useful properties: that $|P(m) - P(m')|$ is small and that, except when $P(m)$ is very small, $D(\rho_k(m), \rho_k(m'))$ is small. That is, Eve's chance of being caught does not depend much on the message being sent, and unless she has a large chance of being caught, she has little information about the message, even after learning the key. In particular, Eve cannot tell whether the message was m or m' .

4.3 Uncloneable Encryption and Quantum Key Distribution

Uncloneable encryption is closely related to quantum key distribution. In fact, any uncloneable encryption can be used to perform secure quantum distribution. In QKD, Alice and Bob share authenticated classical channels and an insecure quantum channel, and use just those resources to create a shared key. Alice and Bob do not (generally) use any pre-existing secret key beyond whatever is used in classical authentication.

Given these same resources and an uncloneable encryption scheme, Alice can perform QKD with the following protocol:

Alice generates random strings k and x .

Alice sends the message x to Bob using the uncloneable encryption scheme with key k .

Bob announces (on the authenticated classical channel) that he received the message.

Alice announces k (again using the authenticated classical channel).

Bob checks if the message is valid, and reports the result. If it is, Alice and Bob use x as their new secret key.

The properties of uncloneable encryption guarantee that this is a secure QKD scheme : Eve gets the quantum state and then later learns the key , but we know that her residual density matrices , conditioned on Bobs accepting the transmission , are very similar. Therefore , she almost always (for most values of k) has little information about the established key x .

Quantum authentication is slightly stronger than uncloneable encryption , which is in turn slightly stronger than quantum key distribution. Nevertheless, the differences are really quite small, meaning quantum authentication and quantum key distribution are closely related. This is rather surprising , give that the tasks of authenticating quantum information and encrypting classical information at first sight appear completely unrelated.

4.4 Computational Security

In classical cryptography, we frequently use a computational assumption to encrypt long messages with a short key. Does uncloneable encryption still work if the key is not truly random, but is instead a pseudorandom sequence generated from a much shorter secret key shared by Alice and Bob? A similar question arises in the context of QKD. Alice must make a lot of random choices when preparing the qubits to send to Bob. Generating truly random numbers can be a difficult task. If she instead generates a long pseudorandom sequence, what does that do to the security of QKD?

In both cases, the answer is that Eve still cannot learn the secret message, provided she has no quantum algorithm to break the pseudorandom sequence. Furthermore, even if she can eventually break the computational assumption, it will do her no good : in order to defeat Alice and Bob , Eve must break the pseudorandom sequence before Bob receives the quantum transmission from Alice. Intuitively , this makes a lot of sense : unless she can defeat the scheme during transmission, the uncloneability property holds, preventing her from copying the message down to work on it later.

The definition of a pseudorandom sequence is one which a computationally-bounded Eve cannot distinguish from a truly random sequence.

Theorem : *If Eve can break uncloneable encryption scheme S (which is derived from a quantum authentication scheme) with a pseudorandom key (from oracle K) using an attack of low complexity during transmission, then she has an efficient quantum algorithm that can distinguish K from a truly random sequence.*

Chapter 5

Quantum Key Distribution

5.1 What is Quantum Key Distribution?

We have already discussed that the rise of quantum computing had created a loop hole in the existing Public Key Cryptographical Protocols and as well as multiple usable key Protocols of Private Key Cryptosystem. The state of a quantum computer is a superposition of exponentially many basis states, each of which corresponds to a state of a classical computer of the same size. By taking advantage of interference and entanglement in this system, a quantum computer can perform in a reasonable time some tasks that would take ridiculously long on a classical computer. Based on this knowledge Peter Shore proposed an quantum algorithm for efficient factoring of large numbers in 1994, which clearly showed that the considerably secure and widely used public Key Cryptosystem RSA can easily be broken in near future once a good enough quantum computer is made. It is because RSA rely on complexity of factorization of big integers. A motivating work on Quantum Computing was further published by Lov Grover in 1996. It was the Quantum Searching algorithm which increased the rate of searching from $O(n)$ to $O(N^{1/2})$. Hence limiting the powers of Data Encryption Standard (DES) algorithm, the best amongst the known multiple usable key Protocols. It is also likely that it breaks down in near future.

Now we are left with the single time usable Private key Cryptosystem that is one-time pad protocols. But main disadvantage with this scheme is need of keys as long as the message is each time a new message is to be sent. Hence here comes the need of secure transmission of large number of keys. Even if we don't consider one time pad other protocols also need to transmit keys, but there since the number of keys are small we can assume that they can be transmitted with absolute security. But in case of one time pad due to large amount of data we are forced to use insecure channel. So we have to design protocols which can transmit keys securely through an insecure channel. This is a well known problem in classical cryptography and is called Key Distribution Problem.

This is one of the area around which Quantum Cryptography first started

to develop and have got remarkable success in both theoretical and practical ground. By using the fragile and random quantum states of elementary particles, like photons, quantum cryptography allows a key to be generated by and distributed to separated parties in a way that allows them to both transmit the key secretly and also tell with a certainty that approaches perfect if an eavesdropper has attempted to intercept part or all of the key. This whole process is called Quantum Key Distribution. So far it has been possible to perform QKD over distances of the order of kilometers.

The mostly known Quantum Key Distribution(QKD) protocols are **1. BB84** and **2. B92**. Here we shall have a detailed description of BB84.

5.2 BB84 Protocol

The BB84 protocol is named after its two inventors, Charles Bennett and Gilles Brassard, who published it in 1984. The key emerges as the protocol proceeds starting from an random bit sequence. There are two channels of a different nature separating Alice and Bob. One channel is private and is made to carry quantum bits; for this reason it is termed the quantum channel. The other channel is classical and it is publicly accessible: it could equally well be a telephone line or a radio broadcast. In fact, any data exchanged over the second channel is insecure and open to everyone. Curiously this feature does not affect the overall security of BB84. None of the information exchanged over this channel is of use to an eavesdropper. BB84 takes place in two phases, the first over the quantum channel, the second over the public, classical channel. In the first phase, Alice and Bob exchange a random set of qubits (typically photons over a optical fibre). In the second phase, they discuss some of the measurements that they made. An eavesdropper, Eve, is assumed to have access to both the quantum channel and the classical channel. Furthermore, Eve is assumed to have equipment just as powerful as Alice and Bob have. Due to properties of quantum bits, Eves presence will not go completely unnoticed. Alice and Bob will take measures to minimize the amount of valid information that Eve can obtain about their secret key. In the first phase of the protocol, Alice chooses a basis with which to encode a bit of data at random. When Bob receives the photon sent by Alice, he has to decode each using another basis chosen at random. In many cases, Bobs choice of basis will be different from Alices, and then he is bound to get an incorrect bit of data. There are going to be many cases where Bobs choice of basis is wrong and so does the measurements. The second phase of the protocol allows Alice and Bob to compare their measurements and to detect if eavesdropping has occurred. In those cases where Bob made a wrong measurement, he and Alice discard the corresponding bits. The final step of BB84 protocol is the most important part of the protocol. In this step, a section of bits for which both Alice and Bob have used the same basis are sent by Alice over a public channel. Normally, the corresponding bits in Bobs sequence should agree. In other words, since Alice and Bob used the same basis for encoding and decoding the bits in $TFK(A)$ and $RCK(B)$ it should be that $TFK(A)=TFK(B)$. If not

,it means that an eavesdropper has measured the photons on the channel,and sent random substitutes to Bob.An eavesdropper cannot measure the photon and simultaneously send an identical copy of it to Bob.This is the property of non-clonability of quantum states and is what makes eavesdropper detectable in BB84. The BB84 protocol is shown detail bellow. The version of protocol shown below is capable to deal with errors.

5.2.1 Eavesdropping Check

This is a procedure Bob and Alice go through after they have the key bits. Amongst the bits they are left with after they have gone through all earlier steps, Alice chooses a subset of half the bits that serve as check bits on Eves interference .She tells Bob which bits she selected . Then they announces and compare the the value of check bits. If more than an acceptable number disagree ,they abort the protocol.

5.2.2 Privacy Amplification

Privacy amplification is basically the procedure followed after error correction to get a secure key .Here we shall go through a few basic principles regarding it . Privacy amplification is basically performed with the help of class of functions so called Universal Hash Functions G ,which maps a set of n bit string X to a set of n bit string Y , such that for any distinct $a_1, a_2 \in X$,when g is chosen uniformly at random from G ,then probability that $g(a_1) = g(a_2)$ is at most $1/|Y|$,

The collision entropy of the random variable X with probability distribution $p(x)$ is defined as

$$H_c(X) = -\log \left[\sum p(x)^2 \right]$$

This H_c is important in the following theorem about about universal Hash functions.

Theorem : *Let X be a random variable on alphabet X with probability distribution $p(x)$ and collision entropy $H_c(x)$,and let G be a random variable corresponding to a random choice (with uniform distribution) of a member of universal class of hash functions from X to $\{0,1\}^m$.Then,*

$$H(G(X)/G) \geq m - 2^{(m-H_c(X))}$$

This theorem applies directly to Privacy Amplification. Alice and Bob publicly selects $g \in G$ and each apply it to their secret key after ERROR CORRECTION giving a new string which they use as the secret key. Now if Eves uncertainty about X given her knowledge $Z = z$ is known in terms of the collision entropy to be bounded below by some number ,say $H_c(X/Z = z) > d$,then it follows from the above theorem that,

$$H_c(S/G, Z = z) \geq m - 2^{(m-d)}$$

We can choose m as small as we can and then H_c is nearly equal to m minimizing possible eavesdropping chances, making it a secure key.

PHASE 1. COMMUNICATION OVER A QUANTUM CHANNEL

BB84.1.1 (Generate Initial Bits) Alice generates a random sequence $D = \{d_i | 0 \leq i \leq n-1\} = [d_0, d_1, \dots, d_{n-1}]$ of n bits.

BB84.1.2 (Choose encoding basis) Alice chooses at random whether to use the rectilinear or diagonal basis to encode each of the bits in D . This can be expressed as a sequence of bases $B = \{x_i | 0 \leq i \leq n-1\} = [x_1, x_2, \dots, x_{n-1}]$

BB84.1.3 (Encode and Transmit Qubits.) Alice encodes each of the bits in D with corresponding basis in B and transmits the resulting qubit over the quantum channel.

BB84.1.4 (Receive Qubits.) Bob retrieves qubits one by one from the channel.

BB84.1.5 (Choose Decoding Basis and Decode Qubits) Bob decodes each qubit with the basis x_i ; thus Bob obtains a sequence of $D = \{d_i | 0 \leq i \leq n-1\} = [d_0, d_1, \dots, d_{n-1}]$

PHASE 2. COMMUNICATION OVER CLASSICAL CHANNEL

BB84.2.1 (Disclose Decoding Bases) Bob tells Alice which basis x_i is used to obtain each bit d_i in D .

BB84.2.2 (Discard Invalid Bits) Alice tells Bob, for each bit d_i in D , whether she used the same basis for encoding. She discards those bits in D for which a different basis was used than Bob. Similarly, Bob discards those bits in D for which he used different basis than Alice. The d_i, d_j for which $x_i = x_j$ are sorted in tentative final key TFK(A) and TFK(B).

BB84.2.3 (EVESDROPPING CHECK) Alice and Bob engage a eavesdropping check to get information regarding amount of eavesdropping.

BB84.2.4 (Privacy Amplification) Alice chooses an upper bound u and a security parameter s and performs privacy amplification.

5.3 An Example

Here is an example of the protocol in use, using $+$ and X as rectilinear and diagonal basis and $- = \backslash$ and $/ = /$ these bases respectively. This is a simple example without taken into consideration of eavesdropping.

Alice calculates with basis:	+	X	+	+	X	X	+	+	X	X	+	+	X
Alice sends to Bob:		/		-	/	\		-	\	\	-		/
Bob measures with basis:	+	X	X	+	+	X	+	X	X	+	X	+	X
Bob's results:		/	/	-		\		\	\	-	\		/
Valid data:		/		-		\			\				/
Translated to key:	1	0		0		1	1		1			1	0

Chapter 6

Conclusion and References

6.1 Real Security : Technology, Cost and Complexity

Despite the elegant and generality of security proofs, the dream of a QC system whose security relies entirely on quantum principles is unrealistic. The technological implementation of the abstract principles will always be questionable. It is likely that they will remain the weakest point in all systems. Moreover, one should remember the obvious equation:

Infinite security \Rightarrow Infinite cost \Rightarrow Zero practical interest

On the otherhand , however, one should not under-estimate the following two advantages of QC. First, it is much easier to forecast progress in technology than in mathematics: the danger that QC breaks down overnight is negligible ,contrary to public-key cryptosystems . Next, the security of QC depends on the technological level of the adversary at the time of the key exchange, contrary to complexity based systems whose coded message can be registered and broken thanks to future progress. The latter point is relevant for secrets whose value last many years. One often points at the low bit rate as one of the current limitations of QC . However ,it is important to stress that QC must not necessarily be used in conjunction with one-time pad encryption. It can also be used to provide a key for a symmetrical cipher such as AES whose security is greatly enhanced by frequent key changes. To conclude this part we briefly elaborate on the differences and similarities between technological and mathematical complexity and on their possible connections and implications. Mathematical complexity means that the number of steps needed to run complex algorithms explodes exponentially when the size of the input data grows linearly. Similarly, one can define technological complexity of a quantum computer by an exploding difficulty to process coherently all the qubits necessary to run a (non-complex) algorithm on a linearly growing number of input data. It might be interesting to consider the possibility that the relation between these two concepts of complexity is deeper. It could be that the solution of a problem requires either a

complex classical algorithm or a quantum one which itself requires a complex quantum computer .

6.2 Summary

Quantum cryptography is a fascinating illustration of the dialog between basic and applied physics . It is based on a beautiful combinations of concepts from quantum physics and information theory and made possible thanks to the tremendous progress in quantum optics and in the technology of optical fibers and of free space optical communication. Its security principle relies on deep theorems in classical information theory and on a profound understanding of the Heisenbergs uncertainty principle . We also emphasize the important contributions of QC to classical cryptography : privacy amplification and classical bound information are examples of concepts in classical information whose discovery were much inspired by QC. Moreover, the fascinating tension between quantum physics and relativity, as illustrated by Bells inequality, is not far away . Now , despite the huge progress over the recent years, many open questions and technological challenges remain. One technological challenge at present concerns improved detectors compatible with telecommunication fibers. Two other issues concern free space QC and quantum repeaters. The first is presently the only way to realize QC over thousands of kilometers using near future technology. The idea of quantum repeaters is to encode the qubits in such a way that if the error rate is low , then errors can be detected and corrected entirely in the quantum domain. The hope is that such techniques could extend the range of quantum communication to essentially unlimited distances. Indeed , Hans Briegel et al , showed that the number of additional qubits needed for quantum repeaters can be made smaller than the numbers of qubits needed to improve the fidelity of the quantum channel . One could thus overcome the decoherence problem. However, the main practical limitation is not decoherence but loss (most photons never get to Bob, but those which get there, exhibit high fidelity).

On the open questions side, we emphasize three main concerns. First, complete and realistic analyses of the security issues are still missing . Next , figures of merit to compare QC schemes based on different quantum systems (with different dimensions for example) are still awaited . Finally, the delicate question of how to test the apparatuses did not yet receive enough attention . Indeed, a potential customer of quantum cryptography buys confidentiality and secrecy ,two qualities hard to quantify. Interestingly, both of these issues have a connection with Bell inequality . But , clearly , this connection cannot be phrased in the old context of local hidden variables, but rather in the context of the security of tomorrow's communications. QC could well be the first application of quantum mechanics at the single quanta level. Experiments have demonstrated that keys can be exchanged over distances of a few tens of kilometers at rates at least of the order of a thousand bits per second. There is no doubt that the technology can be mastered and the question is not whether QC will find commercial applications , but when. Indeed , presently QC is still very limited in distance and

in secret-bit rate. Moreover, public key systems occupy the market and, being pure software, are tremendously easier to manage. Every so often, the news is that some classical cipher system has been broken. This would be impossible with properly implemented QC .

6.3 Acknowledgements

We express our deep sense of gratitude to our mentor Dr. Debabrata Goswami for teaching Quantum mechanics from basics to prepare the students from computer science and other non-physics and chemistry disciplines to learn this exciting field of Quantum computing . We are very thankful to him for his continuous help and support in doing this project .

6.4 References

1. M.A.Nielsen and Isaac.L.Chuang, Quantum Computation and Quantum Information, Cambridge University press, 2000.
2. J.Preskill, Quantum Computing: Pro and Con arXiv:quant-ph/9705032.
3. Bennett , Ch.H , 1992 , Quantum cryptography using any two non orthogonal states , Phys . Rev . Lett . 68 , 3121-3124 .
4. Bennett , Ch.H and G.Brassard , 1984 , Quantum cryptography : public key distribution and coin tossing , Int. conf . Computers , Systems Signal Processing , Bangalore , India , December 10-12,175-179 .
5. Bennett , Ch.H. and G.Brassard , 1985 , Quantum public key distribution system IBM Technical Disclosure Bulletin , 28 , 3153-3163 .
6. J.Smolin , 1992a , Experimental Quantum Cryptography , J. Cryptology 5 , 3-28 .
7. Bennett , Ch.H , G.Brassard and Mermin N.D . , 1992b , Quantum Cryptography without Bells theorem Phys. Rev . Lett. 68 , 557-559 .
8. Bennett , Ch.H , G.Brassard and A. Ekert , 1992c , Quantum Cryptography , scientific Am. 267 , 26-33 (int. ed.)
9. Biham,E. and T.Mor , 1997a , Security of Quantum Cryptography against collective attacks , Phys.Rev. Lett. 78 , 2256-1159 .
10. Biham,E . and T.Mor , 1997b , Bounds on Information and the Security of Quantum Cryptography , Phys.Rev.Lett. 79 , 4034-4037
11. Uncloaneable Encryption by Daniel Gottesman, 0210062, 3 Sep 2004
12. Authentication of Quantum Messages by Howard Barnum et al, quant-ph/0205128. 20 May 2002
13. Quantum Digital Signatures by Daniel Gottesman , Isaac L. Chuang, quant-ph/ 0105032 15 Nov 2001.
14. Quantum Fingerprinting, by Harry Buhrman et al, quant-ph/01022001 1 Feb 2001.