Quantum Private Comparison under noise with CSS-Protection layer

Rishabh Singh

A dissertation submitted for the partial fulfilment
of BS-MS dual degree in Science

Under the guidance of

Prof. Arvind

Indian Institute of Science Education and Research Mohali

April 2019

Certificate of Examination

This is to certify that the dissertation titled **"Quantum Private Comparison under noise with CSS-Protection layer"** submitted by **Rishabh Singh** (Reg. No. MS14065) for the partial fulfillment of BS-MS dual degree programme of the Institute, has been examined by the thesis committee duly appointed by the Institute. The committee finds the work done by the candidate satisfactory and recommends that the report be accepted.

Dr. Kavita Dorai | Dr. Sandeep K. Goyal | Prof. Arvind (Supervisor)

Dated: 26.04.2019

Declaration

The work presented in this dissertation has been carried out by me under the guidance of Prof. Arvind at the Indian Institute of Science Education and Research Mohali.

This work has not been submitted in part or in full for a degree, a diploma, or a fellowship to any other university or institute. Whenever contributions of others are involved, every effort is made to indicate this clearly, with due acknowledgement of collaborative research and discussions. This thesis is a bonafide record of original work done by me and all sources listed within have been detailed in the bibliography.

Rishabh Singh (Candidate)

Dated: April 26, 2019

In my capacity as the supervisor of the candidate's project work, I certify that the above statements by the candidate are true to the best of my knowledge.

Prof. Arvind
(Supervisor)

Acknowledgement

First and foremost, I would like to express my gratitude to my thesis supervisor Prof. Arvind, without whose help and supervision, this thesis would have never been possible. I would also like to thank him for giving me space and freedom to explore the subject. The discussions that I had with him has enhanced my capabilities as a researcher.

I also thank Dr. Kavita Dorai and Dr. Sandeep K. Goyal for being my committee members. The discussions I had with Jaskaran Singh and other members of our lab helped me to gain an in-depth understanding of the subject. I would like to thank them as well.

I'm grateful to the Department of Science and Technology, Government of India for providing me financial support through DST-INSPIRE-SHE grants. And above all, I wholeheartedly thank IISER Mohali for providing me the work-space and a sound environment to pursue my research.

Last but not atleast, I would like to thank my friends and family for their love and support, throughout my entire journey.

Rishabh Singh

MS14065

IISER Mohali.

List of Figures

Notations

$\|0\rangle$	$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$
$\|1\rangle$	$\begin{pmatrix} 0 \\ 1 \end{pmatrix}$
$\|+\rangle$	$\begin{pmatrix} 1 \\ 1 \end{pmatrix}$
$\|-\rangle$	$\begin{pmatrix} 1 \\ -1 \end{pmatrix}$
σ_x or X	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
σ_y or Y	$\begin{pmatrix} 0 & -\iota \\ \iota & 0 \end{pmatrix}$
σ_z or Z	$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
H (Hadamard matrix)	$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
$\mathbb{F}_2^n$	n-tuple space with entries $\in \{0, 1\}$
$M^{\otimes n}$	n-tensor product of M

Contents

Abstract

Quantum Cryptography allows us to do secure communication by exploiting the properties of quantum mechanics. The basic idea behind the security of quantum cryptography comes from no-cloning theorem. Whenever an eavesdropper tries to gain information by attacking the quantum channel, one would end up disturbing the state. The communicating parties can easily detect this error by introducing some check bits.

There are various applications of Quantum Cryptography, such as, Quantum Key Distribution (QKD), Quantum Coin Flipping, Quantum Private Comparison (QPC), Quantum Voting, etc. Here, we analyze the security of specifically QKD and QPC.

To render the security of our Quantum Cryptographic protocols, high fidelity of shared quantum states is required. But in real world, quantum channels can be noisy (in addition to the noise caused due to eavesdropping). Quantum Error-Correction allows us to overcome the effects of noise and achieve very high fidelities, given the error rate is below a certain threshold.

We first develop the formalism of error-correction, starting from classical linear codes; the properties of which are exploited in several Quantum Error-Correcting codes. We look at a particular class of such codes, known as CSS Codes; which we then use to prove the security of BB84 QKD Protocol. Some QPC protocols under noiseless, as well as, noisy conditions, are discussed. We then propose a three-party entangled state QPC Protocol which uses CSS Codes to encode our state, and is unconditionally secure.

Chapter 1

Classical Error Correction

Noise in communication systems is inevitable. We try to build our systems so as to avoid noise from being acted on them. But whenever it's not possible, we need to employ certain strategy to overcome the effects of noise. This can be done by adding some redundant information to the data, which can protect the encoded data while being transmitted, and performing error correction to get the original data back. In this chapter, first we develop the formalism for error-correction and then we'll describe it's working.

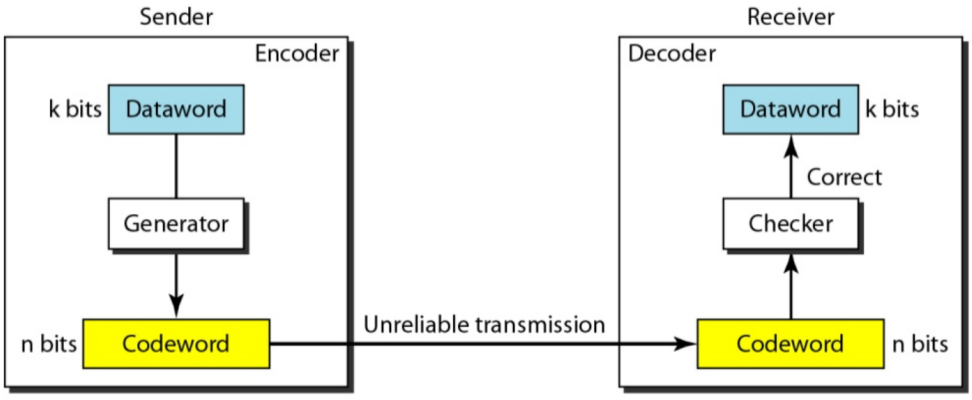


Figure 1.1: The basic idea behind error correction[Sain 00]

1.1 Formalism of Classical Linear Coding

A classical linear[] code can be represented in 2 different formalism:

1.1.1 Generator matrix formalism

A classical linear code C encoding k-bit of information into an n bit code space can be represented by an $n \times k$ generator matrix G with elements which belong to $\mathbb{Z}_2$

$$G: \{0,1\}^k \mapsto \{0,1\}^n \quad ; \quad n \geq k \tag{1.1}$$

Here, we call C a $[n, k]$ code.

For example, consider a 6-bit repetition code, where $k = 2$ and $n = 6$;

Gx encodes x as follows

$$00 \mapsto 000000, \quad 01 \mapsto 000111,$$
$$10 \mapsto 111000, \quad 11 \mapsto 111111$$

In matrix form, G can be written as

$$G = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

The set of all possible codewords are spanned by the columns of G. So for unique encoding, we require columns of G to be linearly independent.

Here we can also see that encoding k bits in $\{0,1\}^n$ space would require $n.2^k$ bits, whereas, in linear encoding, we need only $n.k$ bits by defining it with a $n \times k$ generator matrix.

1.1.2 Parity Check Formalism

By looking at a generator matrix, one can easily visualise the connection between a message and its encoding. However, to perform error-correction, we require Parity-Check formalism. The relation between an $n \times k$ generator matrix and $n - k \times n$ parity-check matrix is such that if we write $G \equiv [\frac{I_k}{A}]$, then $H \equiv [A|I_{n-k}]$.

The interesting property of H for being used in error-correction is that if $y\ (= Gx)$ is a codeword of $[n, k]$ linear code, then $Hy = 0$.

Hence, we can also say that codewords of a linear code $[n, k]$ are kernel of H. We will see

in the next section that how it is useful for error detection and correction.

In matrix form,

$$\begin{pmatrix} r_1 & . & . & . & . & . & . \\ r_1 & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ r_{n-k} & . & . & . & . & . & . \end{pmatrix}_{n-k \times n} \begin{pmatrix} c_1 \\ c_2 \\ . \\ . \\ . \\ . \\ c_n \end{pmatrix}_{n \times 1} = \begin{pmatrix} 0 \\ 0 \\ . \\ . \\ 0 \end{pmatrix}_{n-k \times 1} \tag{1.2}$$

Here, r_i represents a row of H and column c is a codeword generated by G.

1.2 Error detection and correction

For $x \in \{0,1\}^k$ to be encoded by a linear code $[n,k]$ defined by $n \times k$ generator matrix G, we can write $y = Gx$, for y in G. Suppose a bit flip occurs on the j^{th} bit, its representation in matrix form is

$$e_j = \begin{pmatrix} 0 \\ 0 \\ . \\ . \\ 1 \\ 0 \\ . \\ . \\ 0 \end{pmatrix}_{n \times 1}$$

where 1 is at j^{th} position. Now because of error, the encoded bit y transforms to $y' = y + e$. The error syndrome can determined using H as follows

$$\begin{aligned} Hy' &= H(y + e_j) \\ Hy' &= He_j \end{aligned}$$

From the knowledge of He_j, we can know at which position the error has occurred, which can be rectified to get y back.

Definition [Hamming Distance] Let $x, y \in \{0,1\}^n$, then the hamming distance d between x and y is the number of places at which they differ.

For example, $d(10010, 01110) = 3$

Definition [Hamming Weight] For $x \in \{0,1\}^n$, hamming weight w is the number of places at which x is 1. In other words, it is the hamming distance between x and 0.

For example, $wt(10010) = 2$

It can be easily shown that $d(x, y) = wt(x + y)$.

Definition [Distance of a code] The distance of a code $d(C)$ is the minimum hamming distance between any two codewords $x, y\ ; x \neq y$. Or in other words, it is the minimum possible hamming weight of a codeword.

$$\begin{aligned} d(C) &= \min_{x,y \in C, x \neq y} d(x, y) \\ &= \min_{x,y \in C, x \neq y} wt(x + y) \\ &= \min_{x' \in C, x' \neq 0} wt(x') \end{aligned}$$

A code C with distance d is written as $[n, k, d]$ code. A code to be able to correct t errors must have distance $d = 2t + 1$, where t is an integer. Suppose $d(y_i, y_j) \geq 2t + 1 \; \forall \; x, y$; and y' has a maximum distance of t with a codeword y, it can be corrected back to y.

Lemma 1.1 A code with parity check matrix H has distance d if H has $d - 1$ linearly independent columns.

Proof If $y \in C$, then $Hy = 0$,

$$Hy = \sum_i h_i y_i = 0$$

where h_i are columns of H and y_i are elements of y.

$$\sum_{For\ d\ values\ of\ i} h_i = 0$$

Hence, these particular d columns are linearly independent.

If $\sum_{For\ d-1\ values\ of\ i} h_i = 0$, it would imply that there exist a codeword which has hamming weight $d - 1$, which contradicts the fact that the code has distance d. Therefore, any set of $d - 1$ columns of H are linearly independent.

1.3 Dual of a code

Suppose C is an $[n, k]$ which has generator matrix G and parity check matrix H. Then we define a dual of C, denoted by $C^{\perp}$, which has generator matrix H^T and parity check matrix G^T. C^T is an $[n, n-k]$ code.

$$C^{\perp} \ : \ \{y \in C^{\perp};\ y.x = 0 \ \forall \ x \in C\}$$

If g_i denotes a row of G, and h_j denotes a column of H, then $g_i.h_j = 0 \ \forall \ i, j$. This implies that all columns of C and $C^{\perp}$ are orthogonal to each other.
C is called strictly self dual if $C = C^{\perp}$, and weakly self-dual if $C \subseteq C^{\perp}$.

Lemma 1.2: A code C denoted by $[n, k]$ is weakly self dual iff $G^T G = 0$.

Proof For $y \in C$, we have $Gx = y$; and since $C \subseteq C^{\perp}$, all such $y \in C^{\perp}$.
Also since G^T is the parity check matrix of $C^{\perp}$, we have

$$\begin{aligned} G^T y &= 0 \\ G^T(Gx) &= 0 \\ G^T G x &= 0 \end{aligned} \tag{1.3}$$

And for any $x \in C$, there is a y, hence above condition holds for all values of y. Hence, $G^T G = 0$.
Now, for its converse, we have

$$\begin{aligned} G^T G x = G^T y = 0 \\ \Rightarrow x \in C^{\perp} \end{aligned}$$

But we also have $Gx = y$, so $x \in C$, hence, $C \subseteq C^{\perp}$.

Lemma 1.3 If $x \in C$, where C is a linear code, then

$$\sum_{y \in C} (-1)^{x.y} = \begin{cases} |C|, & \forall \ x \in C^{\perp} \\ 0\ , & \forall \ x \in C^{\perp} \end{cases} \tag{1.4}$$

Proof For $y \in C$ and $x \in C^T$, we have

$$\begin{aligned}\sum_{y\in C, x\in C^{\perp}} (-1)^{x.y} &= \sum_{y\in C}(-1)^0 \\ &= \sum_{y\in C} 1 \\ &= |C| \qquad\qquad ;\ x \notin C^{\perp}\end{aligned}$$

And for $x \notin C^T$,

For a particular x, the number of values of y for which $x.y = 0(mod2)$ is equal to that of for which $x.y = 1(mod2)$. Thus,

$$\sum_{y\in C}(-1)^{x.y} = \sum c(1 + (-1))$$

where c is a constant. Hence,

$$\sum_{y\in C}(-1)^{x.y} = 0 \qquad\qquad ;\ x \notin C^{\perp}$$

Chapter 2

Quantum Error-Correction

In real world, the existence of a perfectly closed quantum system is almost impossible. As such, real quantum systems end up having unwanted interactions with the environment, which shows up as noise. To control that noise, we need to understand how noise processes occurs, and need to develop the formalism to correct those errors. In this chapter, we look at the formalism of quantum operations, bit flip, phase flip and depolarizing channel. We then illustrate how a general theory of quantum error-correction can be constructed. Then we see how classical error correcting codes can be incorporated into quantum error correction, and illustrate a particular class of quantum error-correcting codes, known as CSS codes.

2.1 Quantum Operations

Interaction between physical quantum systems and environment causes the system to undergo completely arbitrary time evolution. We use Quantum Operation Formalism to model this interaction.

Consider the state of a system is ρ. Upon interaction with the environment, our quantum state transforms to

$$\rho' = \varepsilon(\rho) \tag{2.1}$$

Suppose the (environment + system) together form a closed quantum system. Assume the environment to be in initial state $|e_0\rangle\langle e_0|$, then their combined evolution can be represented by an unitary transformation, which can be written as

$$U(\rho \otimes |e_0\rangle\langle e_0|)U^\dagger \tag{2.2}$$

To know the final state of the system, we just trace out the state of the environment as follows [Nielsen 00]:

$$\begin{aligned}
\rho' &= Tr_{env}[U(\rho \otimes |e_0\rangle\langle e_0|)U^\dagger] && (2.3)\\
&= \sum_i \langle e_i|U(\rho \otimes |e_0\rangle\langle e_0|)U^\dagger|e_i\rangle && (2.4)\\
&= \sum_{i,j,k} \langle e_i|U|e_k\rangle\langle e_k|(\rho \otimes |e_0\rangle\langle e_0|)|e_j\rangle\langle e_j|U^\dagger|e_i\rangle && (2.5)\\
&= \sum_{i,j,k} \langle e_i|U|e_k\rangle(\rho \otimes \langle e_k|e_0\rangle\langle e_0|e_j\rangle)\langle e_j|U^\dagger|e_i\rangle && (2.6)\\
&= \sum_{i,j,k} \langle e_i|U|e_0\rangle\rho\langle e_0|U^\dagger|e_i\rangle\delta_{k0}\delta_{j0} && (2.7)\\
&= \sum_i \langle e_i|U|e_0\rangle\rho\langle e_0U^\dagger|e_i\rangle && (2.8)\\
& && (2.9)
\end{aligned}$$

This can also be written as

$$\varepsilon(\rho) = \sum_i M_i \rho M_i^\dagger \tag{2.10}$$

where $M_i \equiv \langle e_i|U|e_0\rangle$. M_i are called operation elements of the quantum operation ε. In general, $\sum_i M_i M_i^\dagger \leq 1$, where the equality holds if the quantum operation ε is trace-preserving. If the system is in a pure state, and the quantum operation is trace-preserving, then the action of U on $|\psi\rangle|e_0\rangle$ can be written as

$$U|\psi\rangle|e_0\rangle = \sum_i M_i|\psi\rangle|e_0\rangle \tag{2.11}$$

It can easily be shown that U preserves the norm, even when the system is in mixed state.

2.1.1 Bit flip and Phase flip channels

[Nielsen 00] Consider a bit flip occurs on a qubit with probability p which is exposed to the environment. In quantum operation formalism, we can say that bit flip has operation elements, $M_0 = \sqrt{p}I$ and $M_1 = \sqrt{1-p}X$. In the operator sum representation, we have

$$\varepsilon \xrightarrow{\mathbf{X}} \varepsilon(\rho) = (1-p)\rho + pX\rho X \tag{2.12}$$

Similarly, for a phase-flip operation, we have

$$\varepsilon \xrightarrow{\mathbf{Z}} \varepsilon(\rho) = (1-p)\rho + pZ\rho Z \tag{2.13}$$

2.1.2 Depolarizing Channel

Suppose a qubit having state ρ is depolarized with probability p, *i.e.* replaced by a completely mixed state $I/2$. The resultant state of the system after this noise has occurred is

$$\varepsilon(\rho) = (1-p)\rho + p\frac{I}{2} \tag{2.14}$$

In operator-sum representation, we can write it as

$$\varepsilon(\rho) = (1-\frac{3p}{4})\rho + \frac{p}{4}(X\rho X + Y\rho Y + Z\rho Z) \tag{2.15}$$

Considering that the state remains unchanged with probability $1-p$, by re-parameterizing p we can also write the final state as

$$\varepsilon(\rho) = (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z) \tag{2.16}$$

Thus the depolarizing channel has the operation elements $\{\sqrt{1-p}I, \sqrt{p/3}X, \sqrt{p/3}Y, \sqrt{p/3}Z\}$.

2.2 Theory of Quantum-Error Correction

[Nielsen 00]This basic idea in theory of quantum-error correction is to protect the quantum states by encoding into a *quantum error-correcting code* by applying an unitary operation. To develop a general theory of quantum error correction, we make 2 broad assumptions:

- Noise $\xrightarrow[\text{by}]{\text{described}}$ ε (a quantum operation)
- Error-Correction operation $\xrightarrow[\text{by}]{\text{described}}$ R (a trace-preserving quantum operation)

In order to perform error-correction successfully, we require:

$$(R \circ \varepsilon)\rho \propto \rho \tag{2.17}$$

where ρ is a quantum state, encoded by C, and that is to be transmitted.
The only condition for *quantum-error correcting code* to be able to protect the state from a particular noise ε , is as follows:
Theorem 2.1 (Quantum-Error Correction condition) [Nielsen 00] Suppose C is a quantum code, P is a projector onto C. Consider ε to be a quantum operation which has operation

elements $\{E_i\}$. A sufficient and necessary condition for the existence of error-correction operation R correcting error ε on C:

$$PE_i^\dagger E_i P = \alpha_{ij} P$$

where α_{ij} is an element of a Hermitian matrix.

Discretization of errors

In Theorem 2.1, we illustrated the condition for protection of the encoded quantum information against a specific noise operation ε, but in reality, we don't have the knowledge of what type of error has occurred to the quantum system. We want the quantum information to be protected against an entire class of noise operations. Luckily, the *linearity of quantum mechanics* allows the condition to be adapted to provide this sort of protection.

Theorem 2.2 [Nielsen 00] Let C be a quantum code , and R is the error-correction operation from Theorem 2.1, which can recover from noise operation ε with operation elements $\{E_i\}$. Let us define the combination of all classes of noise processes with a quantum operation F, having operation elements $\{F_j\}$, where F_j is a linear combination of E_i, i.e., $F_j = \sum_i m_{ji} E_i$, where m_{ji} are the elements of a matrix with complex entries. Then R can also correct the state against the effects of noise operation F on code C.

2.3 Quantum-Error Correcting Codes

In some respects, quantum error-correcting code are quite related to classical linear codes; encoded state undergoes noise, then error is recognized by measuring the error syndrome, and then correcting it as appropriate.

Assume that we want to encode our quantum information into n qubits. We define an error-correcting code space which is a subspace of Hilbert space $\mathbb{C}^{2^n}$ which can protect a small number (t) of qubits against any arbitrary error by measuring the error and subsequently correcting it, without disturbing the encoded state.

2.3.1 CSS Codes

CSS codes [Calderbank 96] is one of the prominent example of large class of quantum error-correcting codes. CSS codes exploits the properties of classical linear codes to detect the quantum errors and correct it.

Suppose C_1 and C_2 are $[n, k_1]$ and $[n, k_2]$ classical linear codes such that $C_2 \subset C_1 \subset F_2^n$, where C_1 and C_2 both can correct upto t errors. We define $CSS(C_1, C_2)$, which is an $[n, k_1 - k_2]$ quantum code, which can correct errors on upto t qubits, through following construction:

Suppose $x \in C_1$, we define our quantum state $|x + C_2\rangle$ by

$$|x + C_2\rangle \equiv \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x \oplus y\rangle \tag{2.18}$$

Suppose $x' \in C_1$, such that $x - x' = y' \in C_2$, then

$$\begin{aligned} |x' + C_2\rangle &\equiv \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x' \oplus y\rangle \\ &= \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |(x - y') \oplus y\rangle \quad ; \; y' \in C_2 \\ &= \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x \oplus y''\rangle \quad ; \; y'' = y - y' \in C_2 \\ &\equiv |x + C_2\rangle \end{aligned}$$

So if x and x' belong to same coset of C_2, that is, $x - x' = y' \in C_2$, then $|x' + C_2\rangle$ and $|x + C_2\rangle$ represent the same code state. And if x and x' belong to different cosets of C_2, then $x + y \neq x' + y' \; \forall \; y, y' \in C_2$, and hence $|x' + C_2\rangle$ and $|x + C_2\rangle$ are orthonormal states. The total number of different code states is the total number of cosets of C_2 in C_1, which is equal to $|C_1|/|C_2| = 2^{k_1 - k_2}$, and we say that $CSS(C_1, C_2)$ is an $[n, k_1 - k_2]$ code. Now let's look at how our encoded state can be used to detect and correct the errors.

Suppose e_1 is an n-bit string which have $1's$ in places where bit-flips have occurred and similarly, e_2 is an n-bit string which have $1's$ in places where phase-flips have occurred.

If $|x + C_2\rangle$ was the original encoded state, then the corrupted state would be

$$\frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{(x \oplus y).e_2} |x \oplus y \oplus e_1\rangle \tag{2.19}$$

To detect the error-syndrome, we introduce the ancillary qubits, and correct it to our original state. Given that $wt(e_1) \leq t$ and $wt(e_2) \leq t$.

Bit-flip correction

We take ancillary qubits, initially all in state $|0\rangle$. We apply unitary operation corresponding to the parity check matrix H_1 of C_1, and since, $x + y \in C_1$, our resultant state is

$$\frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{(x \oplus y).e_2} |x \oplus y \oplus e_1\rangle \mapsto \sum_{y \in C_2} (-1)^{(x \oplus y).e_2} |x \oplus y \oplus e_1\rangle |H_1.e_1\rangle \tag{2.20}$$

With the knowledge of the error syndrome, $H_1.e_1$, we can deduce e_1. By applying σ_z gates to qubits at positions where e_1 is 1, we remove the bit-flips occurred and get the following state

$$\frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{(x \oplus y).e_2} |x \oplus y\rangle \tag{2.21}$$

Phase-flip correction

To detect the phase flips occurred to the encoded state, we apply Hadamard transformation to each qubit, and get the state

$$\begin{aligned} H^{\otimes n}[\frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{(x \oplus y).e_2} |x \oplus y\rangle] &= \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{(x \oplus y).e_2} \frac{1}{2^{n/2}} \sum_z (-1)^{(x \oplus y).z} |z\rangle \\ &= \frac{1}{2^{n/2}\sqrt{|C_2|}} \sum_{y \in C_2} \sum_z (-1)^{(x \oplus y).(e_2 \oplus z)} |z\rangle \\ \text{Suppose } z' \equiv z \oplus e_2 & \\ &= \frac{1}{2^{n/2}\sqrt{|C_2|}} \sum_{y \in C_2} \sum_z (-1)^{(x \oplus y).z'} |z' \oplus e_2\rangle \end{aligned}$$

From Lemma 1.3, we have

$$\sum_{y \in C_2} (-1)^{y.z'} = \begin{cases} |C_2|, & \forall\ z' \in C_2^{\perp} \\ 0\ , & \forall\ z' \notin C_2^{\perp} \end{cases}$$

Using this, we can rewrite the state as

$$\frac{\sqrt{|C_2|}}{2^{n/2}} \sum_{z' \in C_2^{\perp}} (-1)^{x.z'} |z' \oplus e_2\rangle \tag{2.22}$$

This is similar to the case of bit-flip errors with error e_2. Similarly, here also we introduce an unitary operation corresponding to parity check matrix H_2 for $C_2^{\perp}$ (H_2^T is the generator matrix for C_2), to get the error-syndrome $H_2.e_2$ and correct it to get the following state

$$\frac{\sqrt{|C_2|}}{2^{n/2}} \sum_{z' \in C_2^{\perp}} (-1)^{x.z'} |z'\rangle \tag{2.23}$$

We apply Hadamard transformation to each qubit, and get back

$$\frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x \oplus y\rangle$$

which is our original encoded state.

Example [*The Steane Code*] Let's look at an important example of CSS codes, where we have $C_1 (\equiv C)$ which is a $[7, 4, 3]$ Hamming code, whose parity check matrix is

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \tag{2.24}$$

and generator matrix is

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix} \tag{2.25}$$

Suppose $C_2 \equiv C^{\perp}$. Then by definition of a dual code, we have

$$H[C_2] = G[C_1]^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{2.26}$$

We also notice that $C_2 \subset C_1$. We can now write the encoded states of $[7, 1, 3]$ CSS code as

$$\begin{aligned} |0\rangle_L = \frac{1}{2\sqrt{2}} [&|0000000\rangle + |0011101\rangle + |0101011\rangle + |0110110\rangle \\ &+ |1000111\rangle + |1011010\rangle + |1101100\rangle + |1110001\rangle] \end{aligned} \tag{2.27}$$

$$\begin{aligned} |1\rangle_L = \frac{1}{2\sqrt{2}} [&|1111111\rangle + |1100010\rangle + |1010100\rangle + |1001001\rangle \\ &+ |0111000\rangle + |0100101\rangle + |0010011\rangle + |0001110\rangle] \end{aligned} \tag{2.28}$$

Here, the classical bit 0 is encoded as $|0\rangle_L$ and 1 as $|1\rangle_L$.

2.3.2 Generalized CSS Codes

Let's generalize the discussion for $CSS[C_1, C_2]$ quantum code using two n-bit strings x and z, where C_1 and C_2 are $[n, k_1, d]$ and $[n, k_2, d]$ classical codes respectively. The code state can be written as

$$|v_k + C_2\rangle \equiv \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{y.z} |v_k \oplus y \oplus x\rangle \tag{2.29}$$

We denote this quantum code as Q_s with $s \equiv (x, z)$. Here the m-bit string $(= k_1 - k_2)$ is indexed by an n-bit string $v_k \in C_1$. It can easily be seen that if we put $x = 0$ and $z = 0$, we get back our simple $CSS(C_1, C_2)$ code.

From equation(2.21), we have the error-syndromes $H_1.e_1$ $(H_2.e_2)$ for bit-flips (phase-flips) for the simple $CSS(C_1, C_2)$; similarly, for the quantum code Q_s, notice that errors syndromes are $H_1.(x + e_1)$ $(H_2.(z + e_2))$ for bit-flips (phase-flips). So after measuring the error syndrome, we just subtract $H_1.x$ and $H_2.z$, to get the knowledge of e_1 and e_2, and correct it subsequently.

We use these code states in various Quantum Cryptographic protocols.

Chapter 3

Quantum Key Distribution

3.1 Introduction

Quantum Key distribution (QKD) protocols allows two separated parties (say, Alice and Bob) to share secure private key over a public channel. The security of the key is guaranteed by the laws of quantum mechanics, given that the error rate is below a certain threshold. Following is the basic idea behind QKD: An eavesdropper (Eve) cannot gain any information from the quantum state transmitted by Alice to Bob without causing a disturbance in their state.

Proposition 3.1 (*Information gain implies disturbance*) In order to distinguish between 2 non-orthogonal states, any information gain is possible only at the cost of introducing disturbance to the state.

Proof Suppose $|\psi_1\rangle$ and $|\psi_2\rangle$ are two non-orthogonal quantum states, and Eve is trying to gain their information. Without any loss of generality, we can assume that while obtaining the information, Eve unitarily interacts its own system with the states $|\psi_1\rangle$ and $|\psi_2\rangle$ by introducing an ancilla. And assuming it does not disturb the states, we have

$$\begin{aligned} |\psi_1\rangle|u\rangle &\rightarrow |\psi_1\rangle|v\rangle \\ |\psi_2\rangle|u\rangle &\rightarrow |\psi_2\rangle|v'\rangle \end{aligned}$$

In order to gain any information, Eve would want $|v\rangle$ and $|v'\rangle$ to be different. Now since unitary transformation will preserve norm, we can write

$$\begin{aligned}\langle\psi_2|\psi_1\rangle\langle u|u\rangle &= \langle\psi_2|\psi_1\rangle\langle v|v'\rangle \\ \langle v|v'\rangle &= \langle u|u\rangle \\ &= 1\end{aligned}$$

and hence, v and v' must be identical. Thus, distinguishing between two non-orthogonal will certainly disturb atleast one of the two states.

We use this idea of transmitting non-orthogonal states from Alice to Bob. And by checking the disturbance caused in the transmitted states, an upper bound can be established on the noise (or eavesdropping) that's being occurring in the channel, such that, if the noise is below a certain threshold, they perform information reconciliation and privacy amplification to get a shared secret key; and if it's above that, they discard the key and start it over again. The threshold for the maximum amount of tolerable error depend upon the efficacy of our information reconciliation and privacy amplification protocols.

3.2 QKD Protocols

We discuss here two conventional QKD Protocols of each type, that is, prepare and measure (BB84 protocol) and entanglement based (The EPR protocol).

3.2.1 BB84 Protocol

This protocol was the first quantum cryptographic protocol. It was developed by Bennett and Brassard in 1984, as the name suggests. Photon polarization states are used to transmit the information. The protocol[Nielsen 00] is as follows:

- Alice creates two strings a and b each of $4n$-random classical bits. She encodes the strings as a block of $4n$ qubits.

$$\psi = \bigotimes_{i=1}^{4n} |\psi_{a_i b_i}\rangle \tag{3.1}$$

where $a_i(b_i)$ is the i^{th} bit of $a(b)$, and each of the qubits is one of these four states:

$$|\psi_{00}\rangle = |0\rangle \tag{3.2}$$

$$|\psi_{10}\rangle = |1\rangle \tag{3.3}$$

$$|\psi_{01}\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \tag{3.4}$$

$$|\psi_{11}\rangle = |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \tag{3.5}$$

$$\tag{3.6}$$

- After encoding these bits, she sends the resulting state to Bob.
- Upon receiving the qubits, Bob announces this fact, and measures the qubits randomly in X or Z basis.
- Alice announces b.
- Over the public channel, they check and discard the bits in which Bob measured in different basis than the one in which Alice prepared. Given n is very large, they are left with $2n$ bits.
- Alice randomly selects n bits that serve as check bits. They check their values corresponding to the that. If the error rate is above a certain threshold, then they'll abort the protocol, otherwise they'll continue.
- Alice and Bob together performs information reconciliation and privacy amplification on their remaining n bits to get an m-bit shared key.

3.2.2 The EPR Protocol

This scheme uses entangled pairs of photons, and are distributed such that one pair of photon is with Alice, and other pair with Bob. The entangled states should be perfectly correlated. The protocol is as follows:

- Alice prepares $4n$ pairs of the following EPR state,

$$|\psi_{AB}\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) \tag{3.7}$$

(We use this state because it's rotationally invariant, and will give perfect correlations no matter in which basis they are being measured.)

- Alice sends the second pair of the qubit to Bob over a quantum channel.
- Alice and Bob measures their qubits randomly in S_z $\{|+\rangle, |-\rangle\}$ or S_x $\{|0\rangle, |1\rangle\}$ basis.
- Over the public channel, they do basis reconciliation. Given that n is very large, they are left with $2n$ bits.
- Alice randomly selects n bits that will serve as check bits. Then they'll check their values corresponding to them. If the error rate is above a certain threshold, then they'll abort the protocol, otherwise they'll continue.
- Alice and Bob together performs information reconciliation and privacy amplification on their remaining n bits to get an m-bit shared key.

3.3 Privacy and coherent information

We have discussed about the basic QKD protocols and argued that it's secure. Now let's look at the quantitative bounds, in terms of quantitative measures of quantum information, and it's connection with obtainable security of Quantum Cryptography.

[Nielsen 00] The lower bound ability to send private information through a quantum channel is given by quantum coherent information $I(\rho, \varepsilon)$. Let's consider the most general case, where Alice prepares states ρ_k^A; k is a non-negative integer, which indicates different possible states that Alice can send with respective probabilities p_k. Bob receives the state $\rho_k^B = \varepsilon(\rho_k^A)$ which may differ from ρ_k^A because of the channel noise that might have occurred because of an eavesdropper or any other environmental effects. The mutual information $H_{Alice:Bob}$ between Alice and Bob is bounded by Holevo's bound.

$$\begin{aligned} H_{Alice:Bob} &\leq \chi^B && (3.8)\\ &= S(\rho^B) - \sum_k p_k S(\rho_k^B) && (3.9) \end{aligned}$$

where $\rho^B = \sum_k p_k \rho_k^B$ and χ^B is the Holevo's quantity. Similarly, mutual information between Alice and Eve is bounded above,

$$\begin{aligned} H_{Alice:Eve} &\leq \chi^E && (3.10)\\ &= S(\rho^E) - \sum_k p_k S(\rho_k^E) && (3.11) \end{aligned}$$

Any excess information that Bob has in relative to Eve can be exploited by Alice and Bob to obtain a shared a secret key through techniques like privacy amplification. Let's define the quantity

$$S = sup[H_{Alice:Bob} - H_{Alice:Eve}] \tag{3.12}$$

This is the guaranteed privacy of the channel, where the supremum is over all strategies that Alice and Bob may utilize. Alice and Bob can use a strategy so that $H_{Alice:Bob} = \chi^B$, and for any strategy that Eve can use, $H_{Alice:Eve} \leq \chi^E$. This implies, $S \geq \chi^B - \chi^E$ when a suitable strategy is employed.

Lower bound on S can be obtained by transmitting pure states $\rho_k^A = |\psi_k^A\rangle\langle\psi_k^A|$. Assuming all interactions that occur are due to Eve, to give her the greatest possible advantage.

Now since the combined state of Eve and the one that Alice sends, is a pure state, ρ_k^B and ρ_k^B have same non-zero eigenvalues, and so the entropies, $S(\rho_k^B) = S(\rho_k^B)$. We have

$$\begin{aligned} S &\geq \chi^B - \chi^E && (3.13)\\ &= S(\rho^B) - \sum_k p_k S(\rho_k^B) - S(\rho^E) + \sum_k p_k S(\rho_k^E) && (3.14)\\ &= S(\rho^B) - S(\rho^E) && (3.15)\\ &= I(\rho, \varepsilon) && (3.16) \end{aligned}$$

This is the lower bound for the guaranteed privacy of channel ε.

3.3.1 The security of QKD

The fact that an eavesdropper causes disturbance in order to gain information, is the basis of security of QKD. Let's quantify the security if the final key that Alice and Bob share.

Criterion 3.1 [Nielsen 00] A QKD protocol is *secure*, if for security parameters $s \geq 0$ and $l \geq 0$ that's being chosen by Alice and Bob; and any eavesdropping strategy that Eve may employ; either the protocol aborts, or succeeds with probability of atleast $1 - O(2^{-s})$, and assure that Eve's mutual information with the key is less than 2^{-l}. Also, the key string should be completely random.

It explicitly bounds the knowledge that Eve may have with the final key, given Alice and Bob employ several strategies to achieve suitable values of s and l.

In the next chapter, we specifically look at the security of BB84 and also prove it.

Chapter 4

Security proof of BB84 Protocol

In this chapter, we analyze the BB84 protocol under noisy conditions, and look at its security proof[Shor 00]. We first discuss the Modified Lo-Chau protocol, where Alice and Bob share high fidelity EPR states. Given that Alice and Bob agree to use the protocol when the error rate is below a certain threshold, these high fidelity states inhibit Eve from gaining more than exponentially small amount of information. Then we see connection between Modified Lo-Chau Protocol and the CSS-based Protocol. We then modify the CSS-Based Protocol and get the protocol which is equivalent to, we say, a modified version of BB84 protocol.

4.1 Modified Lo-Chau Protocol

We define the following four Bell states

$$|\psi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|01\rangle \pm |10\rangle$$
$$|\phi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle$$

Recall from Section 2.3.1, we have, C_1 and C_2 as $[n, k_1, d]$ and $[n, k_1, d]$ classical linear codes, where both can correct upto $t[= (d-1)/2]$ errors. Also H_1 is the parity check matrix for C_1, and H_2 is the parity check matrix for $C_2^{\perp}$. $CSS(C_1, C_2)$ is an $[n, k_1 - k_2]$ quantum code which can correct errors on upto t qubits.

Protocol 4.1

1. Alice creates $2n$ EPR pairs in the state $|\phi^+\rangle^{\otimes 2n}$.

2. Alice creates a random $2n$-bit string b, and apply Hadamard transform on second half on those EPR pair corresponding to which $b = 1$.

3. Alice randomly selects n of the $2n$ pairs which would serve as check bits for Eve's interference.

4. Alice sends each of the second half of EPR pair to Bob.

5. Upon receiving, Bob announces this fact publicly.

6. Alice announces the bit string b and position of EPR pairs which are to be used as check bits.

7. Bob performs Hadamard transform on qubits wherever $b = 1$.

8. Alice and Bob measure their check bits in computational $\{|0\rangle, |1\rangle\}$ basis. If the error rate turns out to be above a certain threshold (t), they abort the protocol, otherwise they continue.

9. Alice and Bob perform entanglement purification using CSS codes, to correct the states nearest to the codeword for $CSS(C_1, C_2)$. They use ancillary qubits to perform syndrome measurement corresponding to H_1, and similarly for H_2. Suppose Alice gets bit and phase flip syndromes x and z , and because of noise in the channel, Bob gets different bit and phase syndromes, say, x' and z'. Assume Alice's syndrome define the CSS Code Q_s where $s \equiv (x, z)$. Alice and Bob publicly share the results.

10. Bob computes the syndromes he got with respect to Q_s and correspondingly transforms his state so to obtain $m = k_1 - k_2$ almost perfect EPR pairs.

11. Alice and Bob measure their respective halves of EPR pairs in computational basis to obtain an m-bit shared secret key.

In step 8, since they only use the protocol if the error rate is below t and from Section 2.3.1, error-correction can be preformed to overcome the effects of noise and get back the original encoded state whenever error rate is less than t; we say, the m-bit key shared between Alice

and Bob has very high fidelity.

Now we show that sharing a high fidelity key also implies its security.

Lemma 4.1 (*High fidelity implies low entropy*) [Nielsen 00] If $F^2 =^{\otimes m} \langle\phi|\rho|\phi\rangle^{\otimes m} \geq 1 - 2^{-s}$, where s is a parameter and F is the fidelity between states ρ and $|\phi\rangle^{\otimes m}$, then $S(\rho) < (2m + s + 1/ln2)2^{-s}$.

Proof Since $F^2 =^{m\otimes} \langle\phi|\rho|\phi\rangle^{\otimes m} \geq 1 - 2^{-s}$, the largest eigenvalue of ρ is greater than $1 - 2^{-s}$. Also

$$S(\rho) = -tr(\rho log_2(\rho)) \tag{4.1}$$

Suppose a diagonal matrix ρ_{max} which has maximum entropy, has diagonal entries $\{1 - 2^{-s}, 2^{-s}/(2^{2m} - 1), 2^{-s}/(2^{2m} - 1), ..., 2^{-s}/(2^{2m} - 1)\}$, we have

$$\begin{aligned} tr(\rho_{max}) &= 1 - 2^{-s} + (2^{2m} - 1)\frac{2^{-s}}{2^{2m} - 1} \\ &= 1 \end{aligned} \tag{4.2}$$

Now $S(\rho) \leq S(\rho_{max})$,

$$S(\rho) \leq -(1 - 2^{-s}) \log_2(1 - 2^{-s}) - (2^{-s}) \log_2 2^{-s} + 2^{-s} \log_2(2^{2m} - 1) \tag{4.3}$$

$$\leq -(1 - 2^{-s})(-2^{-s}) + s2^{-s} + 2^{-s} \log_2(2^{2m} - 1) \ \text{[Using } \log(1 + x) \leq x] \tag{4.4}$$

$$\leq 2^{-s}(1 + s + \log_2(2^{2m} - 1)) - 2^{-2s} \tag{4.5}$$

$$\leq 2^{-s}(2m + s + 1) + O(2^{-2s}) \tag{4.6}$$

$$\tag{4.7}$$

By Holevo's Bound, we know that the accessible information to Eve is upper bounded by $S(\rho)$. Hence, the mutual information that Eve has with key is exponentially small, in the case where Alice and Bob agree to use the protocol (when error rate is less than t).

4.2 CSS based Protocol

Now we'll see how the Modified Lo-Chau protocol can be made equivalent to quantum error correction protocol. We see that Alice can also measure her check bits and then send the encoded state to Bob through the channel. Also note that Alice could have performed syndrome measurements before sending the states.

To measure the check bits before sending the pair to Bob is equivalent to the fact that Alice choosing randomly from states $|0\rangle$ and $|1\rangle$. Also, measuring the error syndrome first is equivalent to sending m halves of EPR pairs, encoded in CSS Code Q_s, where $s \equiv (x, y)$, and $H_1.x$ and $H_2.z$ are bit and phase syndromes respectively.
Now in Step 11 of protocol 4.1, Alice and Bob measure their halves of EPR pairs. Since it doesn't matter in which order their halves are measured, Alice can instead measure her halves before sending the EPR pairs, which is like, Alice choosing a random m bit key and encoding it using Q_s. Thus we have following protocol equivalent to the Modified Lo-Chau protocol.

Protocol 4.2 [Nielsen 00]

1. Alice creates a random m-bit key k, a random check bit string of length n and a random $2n$-bit string b.
2. Alice also creates 2 random n-bit strings x and z to generate $s \equiv (x, z)$.
3. Alice encodes the key $|k\rangle$ using CSS Code Q_s.
4. Alice randomly chooses n positions (out of $2n$ positions) and puts the code bits in these positions and check bits in remaining positions.
5. Alice applies Hadamard transform to the qubits corresponding to which $b = 1$.
6. Alice send the resulting state to Bob. Upon receiving, Bob announces this fact.
7. Alice announces the positions of check bits, b and s.
8. Bob performs Hadamard transform wherever b is 1.
9. Alice and Bob perform measurement on check bits and if the error rate is above a particular threshold (t), then they abort the protocol, otherwise they continue.
10. Bob decodes the remaining n qubits using Q_s.
11. Bob measures the remaining qubits in computational basis, to get an m-bit shared secret key.

So as long as the error rate is below t, Alice and Bob can share very high fidelity states, which results in the security of the shared key.

4.3 Modified BB84 Protocol

We now see how the CSS based Protocol can be converted into a modified version of BB84 protocol. Note that Bob does not care about the phase values, but only about the bit values. So performing phase correction is not required, and as such, Alice doesn't need to send z. We can say, Bob would receive a mixed state averaged over all z. From equation 2.29, we have

$$|v_k + C_2\rangle \equiv \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{y.z} |v_k \oplus y \oplus x\rangle \tag{4.8}$$

where $v_k \in C_1$, where m-bit key k is indexed by n-bit key v_k. Now the state that Bob receives can be written as

$$\frac{1}{2^n |C_2|} \sum_z \sum_{y_1, y_2 \in C_2} (-1)^{(y_1+y_2).z} |v_k \oplus y_1 \oplus x\rangle\langle v_k \oplus y_2 \oplus x| \tag{4.9}$$

On solving, the expression becomes

$$\frac{1}{|C_2|} \sum_z \sum_{y \in C_2} (-1)^{(y.z} |v_k \oplus y_1 \oplus x\rangle\langle v_k \oplus y_2 \oplus x| \tag{4.10}$$

The above state is equivalent to a mixture of states $|v_k \oplus y \oplus x\rangle$ with y being chosen randomly from C_2. Alice tells Bob x (the error-correction information) and sends the state $|v_k \oplus y \oplus x\rangle$ through a quantum channel. Given the value of n is very large, these are just random variables in F_2^n with $v_k + y \in C_1$. Bob would receive the state $v_k + y + x + e$, where e is bit flip error. Bob subtracts x from the value he gets and corrects e from $H_1.e$ and gets $v_k + y \in C_1$. Alice has the knowledge of v_k, but not of y. To share the same key, Alice and Bob calculate $v_k + y + C_2$, which is the coset of C_2 in C_1, and both will get the same coset. This is equivalent to getting an m-bit string, using which we can have 2^m, which is also equal to the number of cosets of C_2 in C_1.

Now let $c = v_k + y + x,\ d = v_k + x\ \in C_1$, so that, $c + d = x$. Alice sends $|d\rangle$ to Bob, with $x = c + d$ being the error correction information. Bob obtains $d + e$ and subtracts $c + d$ to correct the error and get $d \in C_1$. The final key is a coset of $c + C_2$.

Protocol 4.3 [Nielsen 00]

1. Alice creates $(4 + \delta)$ random bits.

2. Alice also creates a $(4 + \delta)$-bit string b. For which, if $b = 0$, she creates a state randomly in $\{|0\rangle, |1\rangle\}$ basis, otherwise in $\{|+\rangle, |-\rangle\}$ basis.

3. Alice sends the resulting state to Bob. Upon receiving, Bob announces this fact.

4. Bob measures the qubits randomly in $\{|0\rangle, |1\rangle\}$ or $\{|+\rangle, |-\rangle\}$ basis.

5. Alice announces b.

6. Bob keeps only those results where he measured in the same basis in which Bob prepared. With high probability, they'll be left with $2n$ bits, and if not, they'll abort the protocol.

7. Alice randomly selects the positions for check-bits and announces it. They publicly compares their values, and if error threshold is more than t, they abort the protocol, otherwise they continue.

8. Alice announces $c + d$, where c is the string of remaining non-check bits, and d is a random codeword in C_1.

9. Bob subtracts $c + d$ from $d + e$ (code qubits), and subsequently corrects $c + e$ to get a codeword in C_1.

10. Alice and Bob computes the coset of $c + C_2$ to get a shared secret key k.

We have systematically proven the security of BB84 protocol.

Chapter 5

Quantum Private Comparison

5.1 Introduction

Secure multi-party computation [Yao 82] allows multiple parties to compute a function, without disclosing their private information to any other party. Let's look at a particular case, where two parties want to compare their private information without sharing it. Quantum Private Comparison(QPC) allows us to do that.
Suppose Alice and Bob have private information M_A and M_B respectively. In QPC, we compute the following function $f(M_A, M_B)$ where

$$f(M_A, M_B) = \begin{cases} 0, & \text{if } M_A = M_B \\ 1, & \text{if } M_A \neq M_B \end{cases} \tag{5.1}$$

Also, Alice and Bob don't want to share their private information with each other. Lo [Lo 97] pointed out that this is possible only if this process is facilitated through a third party (Charlie). Now the problem is that Alice and Bob don't want to share their information with anyone, including Charlie. For our purpose, we assume that Charlies is semi-honest, such that she may try to gain the information about Alice and Bob's information, but cannot be corrupted by the adversary (i.e. Alice or Bob).

In this chapter, we first discuss QPC protocols under noiseless conditions, then we look at a CSS-based QPC protocol which works under noisy conditions. Then we propose a protocol which uses three-party entangled state and is robust against noise (upto a certain threshold). The high fidelity of these states, that's being achieved by encoding our states

with CSS codes; also guarantees the security of the QPC protocol.

5.2 QPC Protocols under noiseless conditions

5.2.1 EPR based QPC Protocol

Suppose Alice and Bob are connected by a quantum channel, which is vulnerable to eavesdropping, as well as a public classical channel. Alice and Bob have private information M_A and M_B each of length The QPC protocol using EPR pairs, is as follows:

Protocol 5.1 [Tseng 12]

1. Charlie creates a random n bit string C_T. For each bit, he prepares a quantum state. If the bit value is 0, he prepares anyone of the states $|\phi^{\pm}\rangle$. If bit value is 1, he prepares anyone of the states $|\psi^{\pm}\rangle$. First half of the sequence is labelled by T_A, and the second half by T_B.

2. To check error rate, Charlie prepares n qubit decoy states D_A and D_B randomly in the following states: $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$. He randomly arranges qubits corresponding D_A and D_B in between of T_A and T_B respectively, to form sequences S_A and S_B, and sends them to Alice and Bob respectively.

3. Alice and Bob receives the qubits and announces this fact. Charlie, in turn announces the positions of D_A and D_B, and the basis [$\{|0\rangle|1\rangle\}$ or $\{|+\rangle, |-\rangle\}$] in which they're prepared.

4. Alice and Bob measure their decoy states in appropriate basis and compare it publicly. If the error rate is above an acceptable rate, they abort the protocol, otherwise they continue.

5. Alice and Bob measures their remaining qubits in $\{|0\rangle, |1\rangle\}$ basis, and get the sequences R_A and R_B respectively. In the absence of noise, these sequences are equal to T_A and T_B. And hence, $R_A \oplus R_B = C_T$.

6. Alice and Bob calculate $C_A = R_A \oplus M_A$ and $C_B = R_B \oplus M_B$ respectively.

7. They cooperate to calculate $C = C_A \oplus C_B$, and send the resultant string to Charlie.

8. Charlies calculates $R_C = C \oplus C_T$. The output will be non-zero iff $M_A \neq M_B$, otherwise it's equal to 0. Charlie announces the output publicly.

The function $f(M_A, M_B)$ has successfully been computed, given the quantum channel is noiseless.

5.2.2 QPC Protocol with W states

Here, a QPC Protocol which uses W-states[Zhang 14] is being presented. W-states are much more robust against noise. Also, the techniques involving preparation of W-states are much more mature, as compared to GHZ states. Following four W states are used in the protocol:

$$\begin{aligned} |w_1\rangle &= \frac{1}{\sqrt{3}}(|100\rangle + |010\rangle + |001\rangle) \\ |w_2\rangle &= \frac{1}{\sqrt{3}}(|110\rangle + |000\rangle + |110\rangle) \\ |w_3\rangle &= \frac{1}{\sqrt{3}}(|000\rangle - |110\rangle + |110\rangle) \\ |w_4\rangle &= \frac{1}{\sqrt{3}}(|100\rangle - |111\rangle - |010\rangle) \end{aligned} \tag{5.2}$$

An ensemble of these four states is used with equal probability, that is to be transmitted through a quantum channel. Alice and Bob has private information M_A and M_B. The protocol is as follows:

Protocol 5.2

1. Charlie prepares n W states, which are chosen randomly from states mentioned in Equation 5.2. Here, if i^{th} state is $|w_1\rangle$ or $|w_4\rangle$, then $r_i = 0$, $i \in [0, n]$, otherwise $r_i = 1$. Upon measurement, the first particle would form the sequence T_A, second one forms T_B, and the third one forms T_C.

2. Charlie introduces 2 sequences of decoy qubits D_A and D_B randomly chosen from states: $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$, and randomly intersperse between T_A and T_B to form the sequences S_A (sends to Alice), S_B (sends to Bob) and S_C (keeps with himself).

3. Alice and Bob receives the qubits and announces this fact. Charlie, in turn announces the positions of decoy qubits and the basis in they were prepared.

4. Alice and Bob measure the decoy qubits. If the error rate is above an acceptable rate, they abort the protocol, otherwise they continue.

5. Alice, Bob and Charlie measure their code qubits in computational basis, and get the sequence R_A, R_B and R_C respectively. In the absence of noise, these sequences are equal to T_A, T_B respectively, and R_C is anyway equal to T_C.

6. Alice and Bob calculate $C_A = R_A \oplus M_A$ and $C_B = R_B \oplus M_B$ respectively. They cooperate to calculate $C = C_A \oplus C_B$. They send the resulting string to Charlie.

7. Charlies computes $R = C_A \oplus C_B \oplus C_C \oplus\ r$. The output will be non-zero iff $M_A \neq M_B$, otherwise it's equal to 0. Charlie announces the output publicly.

Calculation

Following are possible outcomes, each of which results in $T_A \oplus T_B \oplus T_C \oplus\ r = 0$.

r_i	T_{A_i}	T_{B_i}	T_{C_i}
0	0	1	1
0	0	0	0
0	1	1	0
0	1	0	1
0	1	1	0
0	0	0	0
1	0	0	1
1	0	1	0
1	0	0	1
1	1	1	1
1	0	1	0
1	1	0	0

$$R = C_A \oplus C_B \oplus R_C \oplus\ r$$

$$R = (R_A \oplus M_A) \oplus (R_B \oplus M_B) \oplus R_C \oplus\ r$$

In the absence of noise,

$$R = (T_A \oplus T_B \oplus T_C \oplus\ r) \oplus M_A \oplus M_B$$

From above table, we have

$$R = M_A \oplus M_B$$

$$R = f(M_A, M_B) \tag{5.3}$$

5.3 QPC Protocols under noisy conditions

In the last section, we have discussed protocols which works perfectly fine when the quantum channel is noiseless. But in reality, quantum channels are prone to noise in many ways. So, we need to design our protocols such that effects of noise can be overcome. One way is to encode the quantum states with error-correcting codes to preserve the encoded information. Here, we use CSS Codes to encode our states and their subsequent error correction. First we look at the EPR-based QPC protocol[Siddhu 15] which uses CSS Codes to encode the information.(Recall CSS based Protocol 4.2). Then we propose a three-party entangled state QPC Protocol which also uses CSS Codes.

5.3.1 EPR-based QPC protocol using CSS Codes

1. Charlie creates two random n-bit strings R_A and R_B and uses CSS-based Protocol (4.2) separately for both and send it to Alice and Bob respectively.
2. Charlie computes $r = R_A \oplus R_B$.
3. Alice and Bob calculate $C_A = R_A \oplus M_A$ and $C_B = R_B \oplus M_B$ respectively.
4. They cooperate to calculate $C = C_A + C_B$, and send the resultant string to Charlie.
5. Charlies calculates $R_C = C \oplus r$. The output will be non-zero iff $M_A \neq M_B$, otherwise it's equal to 0. Charlie announces the output publicly.

The above protocol works as long as error rate is less than a particular threshold (t).

5.3.2 Three-party entangled state QPC Protocol using CSS Codes

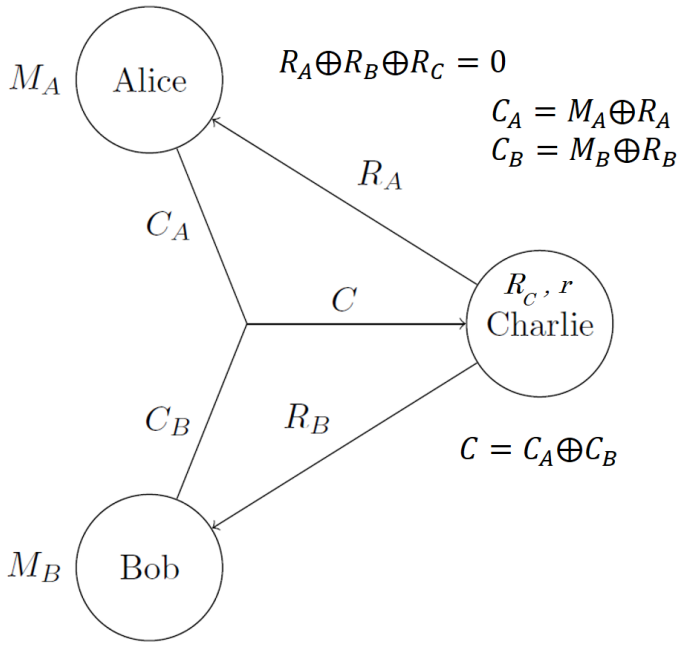


Figure 5.1: The schematic diagram of the protocol[Siddhu 15]

The protocol is as follows:

Protocol 5.4

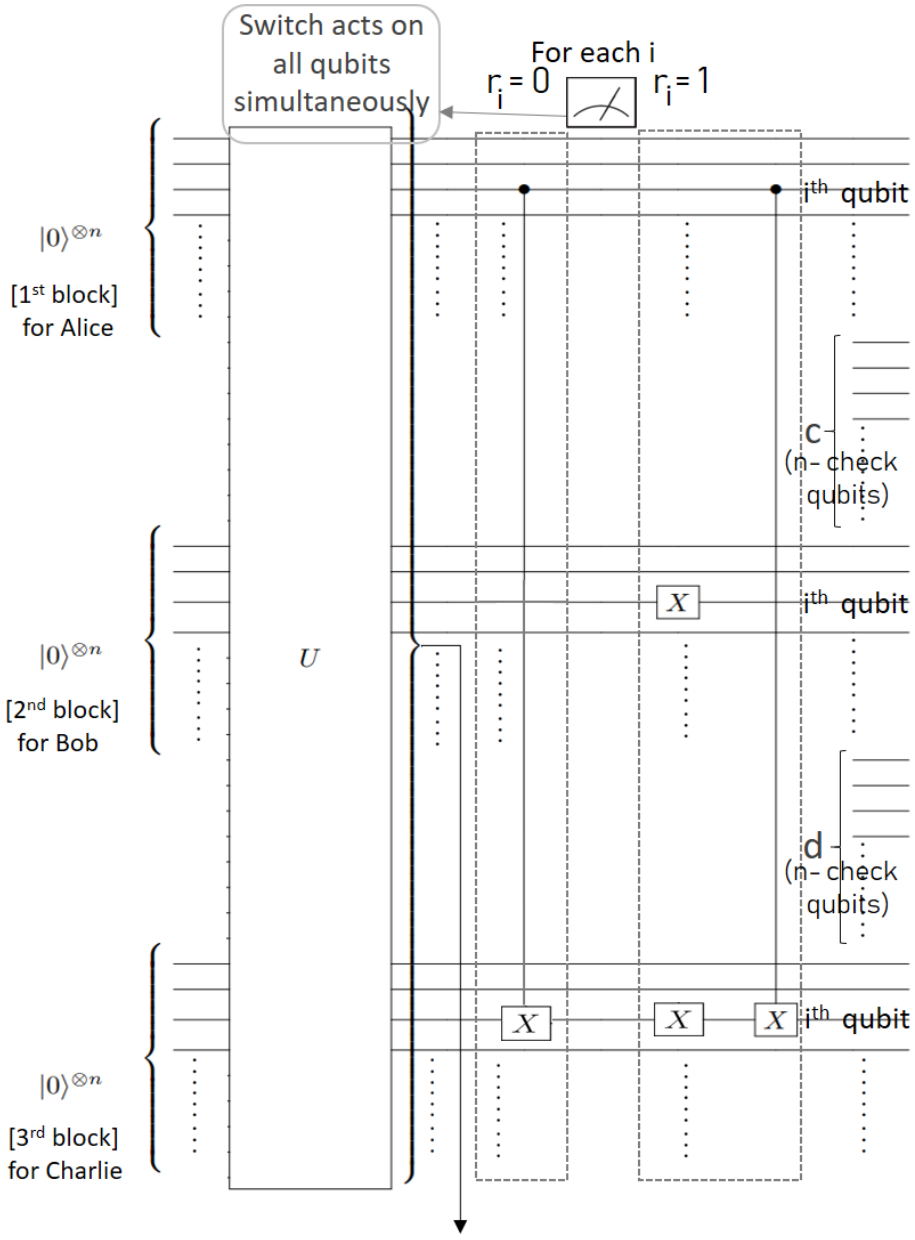
1. Charlie creates a random m-bit string r, two random $2n$-bit strings b_A and b_B and two another n-bit strings c and d.

2. Then he prepares the state Ψ as shown in the circuit diagram using

3. Corresponding to each bit r_i, there is a codeword v_{r_i} (of length n) which belongs to the coset of C_2 in C_1 [For example, if $r_i = 0$, $v_{r_i} = 0000....0(n$ times$)$, similarly for $r_i = 1$]. At positions where $r_i = 0(1)$, Charlie switches the button to left (right). (Switch acts on all qubits).

4. Charlie randomly chooses n positions (out of $2n$) and puts check bits $c(d)$ in the 1^{st} block (2^{nd} block) in these positions, and in the remaining positions, he puts code bits.

5. Charlie performs Hadamard transform on qubits where $b_A = 1$ ($b_B = 1$) on the 1^{st} block (2^{nd} block) and sends it to Alice (Bob).



$$\Psi = \sum_{j=0}^{2^n} |j\rangle|j\rangle|j\rangle = \sum_{v_k,x,z} |\chi_{v_k,x,z}\rangle|\chi_{v_k,x,z}\rangle|\chi_{v_k,x,z}\rangle \tag{5.4}$$

6. Charlie announces b_A and b_B and the positions of check bits. Alice (Bob) will perform Hadamard transform where $b_A = 1$ ($b_B = 1$).

7. Alice (Bob) and Charlie will perform measurement on check bits in the computational basis, and if more than t error occurs, they abort the protocol, otherwise they continue.

8. All 3 perform syndrome measurement on their code bits corresponding to H_1 and H_2. All 3 of them will get random values for x, z (which would be same for all 3 of them). Since Charlie did not send her qubit through the quantum channel and kept with himself, we can assume that no bit-flip and phase-flip errors would have occurred to his qubits. Her syndromes will be $H.x$ and $H.z$, whereas Alice (Bob) will get syndrome $H_1.(x + e_1^A)$ and $H_2.(z + e_2^A)$ ($H_1.(x + e_1^B)$ and $H_2.(z + e_2^B)$). Charlie will announce publicly the sequences x and z.

9. All 3 of them decode their qubits from $CSS_{x,z}(C_1, C_2)$.

10. Alice(Bob)(Charlie) measure their resultant states in computational basis and get the sequence $v_{k_A}(v_{k_B})(v_{k_C})$, and hence $k_A(k_B)(k_C)$ (which we refer as $R_A(R_B)(R_C)$, to follow similar notation). Their corresponding values will be as follows:

r	R_A	R_B	R_C
0	0	0	0
0	1	1	0
1	0	1	1
1	1	0	1

11. Alice (Bob) calculates $C_A = M_A \oplus R_A$ ($C_B = M_B \oplus R_B$).

12. Alice and Bob cooperate to calculate $C = C_A \oplus C_B$ and send it to Charlie.

13. Charlie computes $R = C \oplus R_C$. The output will be 0 if $M_A = M_B$, otherwise it will be non-zero. Charlie will announce whether their information is equal or not.

Calculation

$$\begin{aligned} R &= C \oplus R_C \\ &= C_A \oplus C_B \oplus R_C \\ &= M_A \oplus R_A \oplus M_B \oplus R_B \oplus R_C \\ &= M_A \oplus M_B \\ &= f(M_A, M_B) \end{aligned}$$

Security Analysis

If the protocol is secure against any insider's attacks, it will definitely be secure against any outsider's attacks. So let's analyze the security for only the former case.

Alice (Bob) can either gain information by attacking the quantum channel or directly trying to know M_A or M_B. Now since Alice(Bob) has no information about R_B (R_A), she(he) cannot know M_A (M_B) without attacking the channel. Similarly, Charlie has no information about R_A, R_B, C_A and C_B, so the situation is similar for him.

Now, performing error-correction using CSS Codes has enabled the party to achieve high fidelity of shared qubits, given the error rate is below a particular threshold (and anyway they abort the protocol whenever the error rate is greater than t). Now recall from 4.1, that high fidelity of shared qubits establishes an upper bound on the mutual information than any eavesdropper might have with the shared key. Thus, we can say that the qubits being shared between Alice (Bob) and Charlie are securely transmitted. Hence, we conclude that our QPC Protocol is unconditionally secure.

Conclusion

We proposed a CSS-based three-party entangled state QPC protocol which is robust under noise, as long as the noise is under a particular threshold rate.

While performing QPC under noise, we performed error-correction using CSS Codes to achieve high fidelity shared states, inhibiting the eavesdropper from gaining more than exponentially small amount of information, thus allowing us to render unconditional security of QPC.

In future, we may explore Quantum Multi-Party Comparison, which has many applications such as private auctions, secret ballot elections etc. We can use the same technique to encode the information using CSS Codes to ensure its security.

Bibliography

[Calderbank 96] A. R. Calderbank & Peter W. Shor. *Good quantum error-correcting codes exist*. Phys. Rev. A, vol. 54, pages 1098–1105, Aug 1996.

[Lo 97] Hoi-Kwong Lo. *Insecurity of quantum secure computations*. Phys. Rev. A, vol. 56, pages 1154–1162, Aug 1997.

[Nielsen 00] Michael A. Nielsen & Isaac L. Chuang. Quantum computation and quantum information. Cambridge University Press, 2000.

[Sain 00] Bheem Sain. *Error detection and Correction*. www.slideshare.net, vol. chapter-10, page 12, 2000.

[Shor 00] Peter W. Shor & John Preskill. *Simple Proof of Security of the BB84 Quantum Key Distribution Protocol*. Phys. Rev. Lett., vol. 85, pages 441–444, 2000.

[Siddhu 15] Vikesh Siddhu & Arvind. *Quantum private comparison over noisy channels*. Quantum Information Processing, vol. 14, no. 8, pages 3005–3017, Aug 2015.

[Tseng 12] Hsin-Yi Tseng, Jason Lin & Tzonelih Hwang. *New quantum private comparison protocol using EPR pairs*. Quantum Information Processing, vol. 11, no. 2, pages 373–384, Apr 2012.

[Yao 82] Andrew C. Yao. *Theory and Application of Trapdoor Functions*. In Proceedings of the 23rd Annual Symposium on Foundations of Computer Science, SFCS '82, pages 80–91, Washington, DC, USA, 1982. IEEE Computer Society.

[Zhang 14] Wei-Wei Zhang, Dan Li & Yan-Bing Li. *Quantum Private Comparison Protocol with W States*. International Journal of Theoretical Physics, vol. 53, 04 2014.