Secure Quantum Computation

Project Report

Submitted in partial fulfillment of the requirements of
PHY F491 Special Project

By

Khyati Jain
ID No. 2016B5A70471G

Under the supervision of:

Dr. Radhika Vathsan

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI, GOA CAMPUS
April 2021

Acknowledgements

I would like to thank my supervisor Dr. Radhika Vathsan for her constant support and guidance in my study. This project would not have been possible without her mentoring.

I would also like to thank my peers, Rahul B. S. and K. Karthik Nair, regular discussions with whom, helped me understand the concepts better. Further, I would like to thank Rajat Chaurasia for his valuable inputs.

Contents

Introduction

When the technology to build quantum computers becomes available, it is likely that it will only be accessible to a handful of centers around the world. Much like today's rental system of supercomputers, users will probably be granted access to the computers in a limited way. Another scenario is when two or more parties have access to limited quantum computational power, but together if they trust each other, they could perform universal quantum computation.

In this project, we look at ways in which such multiparty computation, or assisted computation could be performed in a secure way. We begin with looking at a scenario where Alice has limited computational power, and takes help from Bob who has a universal quantum computer. She does this in a secure way such that Bob cannot know her inputs or the function she is trying to compute. But there are limited scenarios in which she could detect errors in Bob's computation. Next we look at a more powerful interactive protocol where Alice does not require any quantum computational power or memory. This protocol is secure and fault tolerant. As Quantum Machine learning and quantum Assisted Machine learning are fast gaining popularity, towards the end, we also look at a protocol which allows a secure and blind way to execute a quantum machine learning algorithm.

Chapter 1

Secure Assisted Quantum Computation

Suppose Alice wants to perform some computation that could be done quickly on a quantum computer, though she can perform some basic gates, she cannot do universal quantum computation. She can, however, do classical computations. Bob can do universal quantum computation and claims he is willing to help. We look at simple efficient protocols by which Bob can help Alice perform the computation in a secure and reliable way. That is, Bob should not be able to get any information about Alice's input, neither should an eavesdropper Eve. Furthermore, we look at the possibilities in which Bob cannot even know about the function Alice is trying to compute. Ways to verify reliable computation by Bob is also considered. Such a scenario is relevant to the case of a Universal Quantum Computer selling time on its QPU to users who will have access to basic gates, but not a universal quantum computer. This is also called *blind quantum computation* [3].

1.1 Problem Statement

Operations available to Alice are single qubit Pauli gates X and Z.

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \; Z = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

She can also swap her qubits. She can not perform any interaction between her qubits.
Bob has a universal quantum computer. We need a secure protocol such that Alice can perform universal computation with the help of Bob

1.2 Protocols for Secure Assisted Computation

Alice uses a private quantum channel to transfer qubits. She encodes the qubit and then Bob performs the gate on the qubit Alice would like him to, and sends it back to her. She can decode it suitably, since she knows the key used in encryption. We show that there exists protocol such that Alice can suitably decode the qubits after Bob applies the gate.

1.2.1 Blind Measurement

Here is the protocol such that Bob can securely help Alice measure her qubits.

1. Encoding: Alice generates two random numbers j, k. Alice encodes her qubit $|\Psi\rangle$ by acting $Z^k X^j$ on it.
2. Action: Bob measures the qubit and sends it to Alice.
3. Decoding: Alice flips the result if j = 1 , else does nothing.

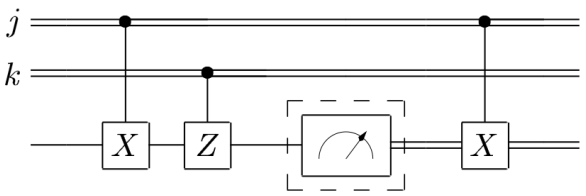

FIGURE 1.1: Secure assisted computational basis measurement. The meter inside a dashed box represents a computational basis measurement, the action performed by Bob

The correctness of the protocol can seen from the following points

- Bob receives a maximally mixed state, and hence cannot get any information from the qubit.
- The action of Z gate does not affect the probabilities of measurement. Application of X gate is equivalent to a bit flip.

1.2.2 Hadamard Gate

Here is the protocol such that Bob can securely help Alice apply the Hadamard gate.

1. Encoding: Alice generates two random numbers j, k. Alice encodes her qubit $|\Psi\rangle$ by acting $Z^k X^j$ on it.

2. Action: Bob acts Hadamard gate on it and sends it to Alice.

3. Decoding: Alice acts $Z^j X^k$ on the qubit.

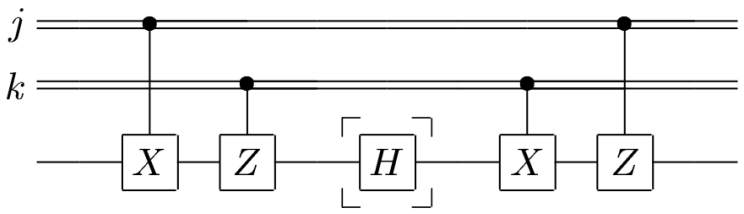

FIGURE 1.2: Secure assisted Hadamard gate

The correctness of the protocol can seen from the following points

- Bob receives a maximally mixed state, and hence cannot get any information from the qubit.
- Decoding successfully restores the qubit :

$$\begin{aligned} |\Psi_f\rangle &= Decode(HEncode(|\Psi_i\rangle)) \\ &= Z^j(X^k H Z^k)X^j |\Psi_i\rangle \\ \because XHZ &= ZHX = H \\ |\Psi_f\rangle &= H |\Psi_i\rangle \end{aligned} \tag{1.1}$$

1.2.3 Controlled Not Gate

Here is the protocol such that Bob can securely help Alice apply the Controlled NOT gate.

1. Encoding: Alice generates 4 random numbers j, k, l, m to encode the two qubits. Alice encodes her qubits $|\Psi_1\rangle$ and $|\Psi_2\rangle$ by acting $Z^k X^j$ and $Z^m X^l$ on them respectively.

2. Action: Bob performs CNOT gate and sends them to Alice.

3. Decoding: Alice acts $Z^m X^j Z^k$ on the control qubit and $X^l Z^m X^j$ on the target qubit.

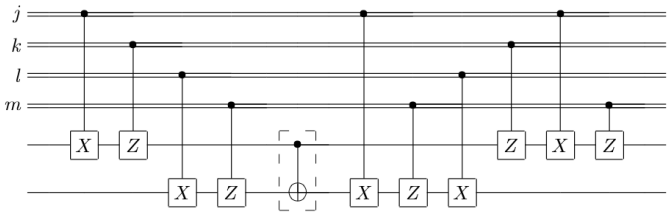

FIGURE 1.3: Secure assisted CNOT gate

The correctness of the decoding protocol can seen as follows:

$$\begin{aligned}\left|\Psi_1'\right\rangle \otimes \left|\Psi_2'\right\rangle &= Decode(CNOT_{12}Encode(|\Psi_1\rangle \otimes |\Psi_2\rangle)) \\ &= Z^m X^j Z^k \otimes X^l Z^m X^j \cdot CNOT \cdot (Z^k X^j \otimes Z^m X^l(|\Psi_1\rangle \otimes |\Psi_2\rangle))\end{aligned} \tag{1.2}$$

For the case l = j =1

$$\begin{aligned}\left|\Psi_1'\right\rangle \otimes \left|\Psi_2'\right\rangle &= Z^m X Z^k \otimes X Z^m X \cdot CNOT \cdot Z^k X \otimes Z^m X \cdot (a_1 |0\rangle + b_1 |1\rangle) \otimes (a_2 |0\rangle + b_2 |1\rangle) \\ &= Z^m X Z^k \otimes X Z^m X \cdot CNOT \cdot ((-1)^k a_1 |1\rangle + b_1 |0\rangle) \otimes ((-1)^m a_2 |1\rangle + b_2 |0\rangle) \\ &= a_1 |0\rangle \otimes (a_2 |0\rangle + b_2 |1\rangle) + (-1)^m b_1 |1\rangle \otimes ((-1)^m a_2 |1\rangle + (-1)^m b_2 |0\rangle) \\ &= a_1 |0\rangle \otimes (a_2 |0\rangle + b_2 |1\rangle) + b_1 |1\rangle \otimes (a_2 |1\rangle + b_2 |0\rangle) \\ \left|\Psi_1'\right\rangle \otimes \left|\Psi_2'\right\rangle &= CNOT \cdot |\Psi_1\rangle \otimes |\Psi_2\rangle\end{aligned} \tag{1.3}$$

Other cases can be proved similarly.

1.2.4 T gate

Here is the protocol such that Bob can securely help Alice apply the T gate.

Round 1:

1. Encoding: Alice generates 2 random numbers j, k to encode the qubit. Alice encodes her qubit $|\Psi\rangle$ by acting $Z^k X^j$.
2. Action: Bob performs T gate and sends it to Alice.
3. Decoding: Alice acts $X^j Z^k$ on qubit.

Round 2: Alice uses and ancillary qubit and swaps it with the qubit if j =1.

1. Encoding: Alice generates 2 random numbers j, k to encode the qubit. Alice encodes her qubit $|\Psi\rangle$ by acting $Z^k X^j$.
2. Action: Bob performs T gate and sends it to Alice.
3. Decoding: Alice acts $X^j Z^k$ on qubit.

The correctness of the decoding protocol can seen as follows:

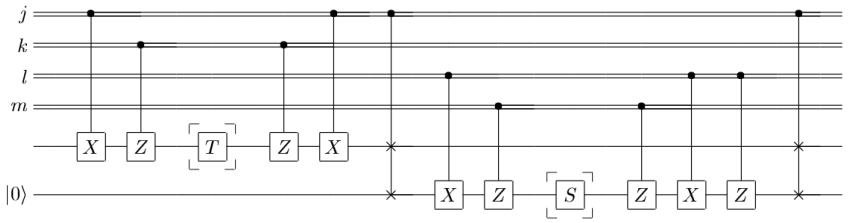

FIGURE 1.4: Secure assisted T gate

Case 1: j = 0 (The operations after the swap gate are done on junk qubit which is discarded)

$$\begin{aligned} |\Psi_f\rangle &= Decode(T \cdot Encode(|\Psi_i\rangle)) \\ &= Z^k \cdot T \cdot Z^k \, |\Psi_i\rangle \\ &\because [Z, T] = 0 \\ &= T \cdot |\Psi_i\rangle \end{aligned} \tag{1.4}$$

Case 1: j = 1 (The operations after the swap gate are done on $|\Psi\rangle$ too)

$$\begin{aligned} |\Psi_f\rangle &= (Z^l \cdot X^l \cdot Z^m \cdot S \cdot Z^m \cdot X^l) \cdot (X \cdot Z^k \cdot T \cdot Z^k \cdot X) \, |\Psi_i\rangle \\ &\because [Z, T] = 0 \text{ and } XTX = T^\dagger \\ &= (Z^l \cdot X^l \cdot Z^m \cdot S \cdot Z^m \cdot X^l) \cdot (T^\dagger) \, |\Psi_i\rangle \\ &\because S = T^2, [Z, S] = 0 \\ &= (Z^l \cdot X^l \cdot T^2 \cdot X^l) \cdot T^\dagger \, |\Psi_i\rangle \\ |\Psi_f\rangle &= T \cdot |\Psi_i\rangle \end{aligned} \tag{1.5}$$

Hadamard, controlled not and T gates are universal for quantum computation in the sense that any unitary transformation can be approximated arbitrarily closely by some sequence of these gates. Hence, using the above protocols, Alice can do universal quantum computation.

1.3 Security of the Protocol

Bob does not know the classical random numbers j, k. So, from his perspective, Alice has applied the depolarising channel. Bob receives the state $Z^k X^j |\psi\rangle$. The density operator is

$$\frac{1}{4} \sum_{j,k=0}^{1} Z^k X^j |\psi\rangle\langle\psi| X^j Z^k = \frac{I}{2}$$

Hence, Bob receives a maximally mixed state and it can obtain no information from it.

The secrecy of the function can be simply ensured by Alice fixing a sequence of gate operations, but passing junk qubits when a particular gate is not needed. She can do this using SWAP gate. The number of gates is increases by at most a factor of three. In such a scenario, Bob can only learn an upper bound on the number of gates in Alice's circuit.

Although we have ensured that Bob doesn't get to know Alice's qubits, Bob could potentially ruin the operation by performing wrong gates, or say not returning the qubits. Alice can check Bob by randomly performing test of a subset of her qubits. In a case where Alice is trying to solve NP, say factoring a large number, it could check the solution on a classical computer. But a general adversarial scenario, in which Bob could induce errors randomly is not handled.

We have showed that Alice can do universal quantum computation. It can be easily shown that there exists no reasonable restriction on classical computation such that "secure assisted classical computation" can be done. Furthermore, it has been proven that the secure assisted quantum computation protocol that we have set up computes the required function in a reasonable amount of time (rounds of the protocol - where each time Alice encodes her qubits, Bob acts a gate, and Alice decodes is one round) up to arbitrary accuracy.

Chapter 2

Measurement Based Quantum Computation

Quantum circuit model is a standard formalism for universal quantum computation. Measurement based quantum computation [1] is another model in which coherent quantum information processing is accomplished via a sequence of single-qubit measurements applied to a fixed quantum state known as a cluster state. It has been shown that such a model can be used to simulate a quantum circuit efficiently, and likewise, since we know quantum circuit model is universal, it can efficiently simulate cluster state quantum computation. Although, the two are equivalent in their power, this model has an ability to illustrate important ideas on quantum computing which will be useful in studying secure assisted quantum computing.

Measurement based quantum computation in remarkable in the sense that all the basic dynamical operations are non-unitary quantum measurements, yet they can still be used to simulate arbitrary quantum dynamics, including unitary dynamics.

2.1 Cluster State Model

In this formalism, the computation begins with a set of qubits with known initial states, and entangled in a certain way. This is followed by a sequence of measurements in different bases. A graph $\mathcal{G}$ with n vertices can be associated to a cluster state such that each node represents a qubit and each edge represents entanglement. The cluster state associated to the graph is defined as follows:

- Prepare each of the n qubits in the state $|+\rangle \equiv (|0\rangle + |1\rangle)/\sqrt{2}$

- Apply controlled-PHASE gates between qubits whose corresponding graph vertices are connected.

State preparation is followed by a sequence of measurements. Following conditions are satisfied:

- All measurements are single qubit measurements
- Choice of the basis of measurement may depend on a function of the output of the previous measurement. The function must be computable "efficiently" on a classical computer

For example in the figure 2.1, the numeric label indicates the time-ordering of the processing measurements, while the unlabelled are not measured, and are a part of the output. Two qubits which are entangled, can have the same label because measurement commutes with the controlled phase operator. Hence the order of measurement is not important in such case.

Unitary operator in the nodes represents an action of the operator followed by standard basis measurement. The $\pm$ notation in $HZ_{\pm\alpha_2}$ and $HZ_{\pm\beta_2}$ indicates that the choice of sign depends on the outcomes of earlier measurements, in a manner to be specified separately. This model,

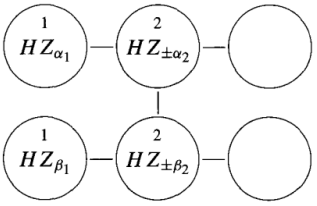

FIGURE 2.1: An example of a cluster state. Image sourced from [2]

although requires more number of qubits compared to the circuit model of computation, it does not require quantum "coherence" for a long time which is a considerable experimental challenge.

2.2 Simulating quantum circuits

We now see how the cluster state model can simulate the circuit based model. The key underlying idea is the 1-bit teleportation. Here m is the output of the measurement of first qubit in standard basis. One bit teleportation can be easily verified by taking $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$. One bit teleportation in figure 2.2 can be generalised to figure 2.3 because Z_θ commutes with $CTRL - Z$.

This is called "teleportation" because the two qubits could be spatially separated after entangling them. And although the first qubit is not transferred physically between the two points, by measuring the first qubit, the information contained in it is transferred to the second. It is remarkable because by varying θ, we can vary the unitary transformation effected on the second

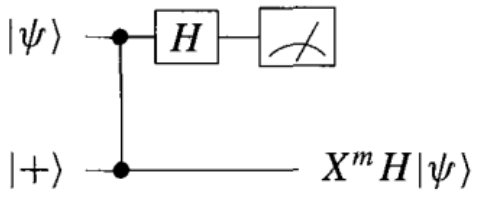

FIGURE 2.2: Quantum 1-bit teleportation

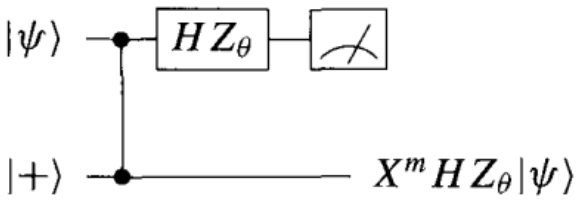

FIGURE 2.3: Modified 1-bit teleportation

qubit, without destroying any quantum information. Note that the information is not transferred *before* the measurement. Now we see how a quantum circuit can be simulated using the cluster state model. Consider the single qubit circuit given in figure 2.4. We show that the equivalent cluster state model is figure 2.5. This is seen as follows : By definition in section 2.1, the cluster state refers to three qubit being initialised with $|+\rangle$ followed by controlled phase gate between first and second qubits, and between second and third qubits. This is followed by acting the operator and measurement. Using the property that measurement and controlled phase commute, figure 2.6 represents the resultant circuit.

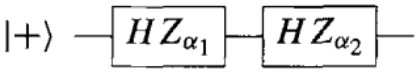

FIGURE 2.4: Single qubit Circuit

FIGURE 2.5: Corresponding cluster state model to figure 2.4

To determine the output of the circuit, we observe that both of the highlighted boxes are of the form of 1- bit teleportation. Hence the output of the circuit is $X^{m_2} HZ_{\pm\alpha_2} X^{m_1} HZ_{\alpha 1} |+\rangle$ where m_1 and m_2 are the outputs of the measurements on the first and second qubits. Observe that feedforward can be used to choose the sign of $\pm\alpha_2$ that $Z_{\pm\alpha_2} X^{m_1} = X^{m_1} Z_{\alpha_2}$. We also have $HX^{m_1} = Z^{m_1} H$, and thus the output may be rewritten as $X^{m_2} Z^{m_1} H\tilde{Z}_{\alpha_2} HZ_{\alpha 1}|+\rangle$, which, up to the known Pauli matrix $X^{m_2} Z^{m_1}$, is identical to the output of the conventional single-qubit quantum circuit. Note that in general each stage will lead to a know pauli correction as a

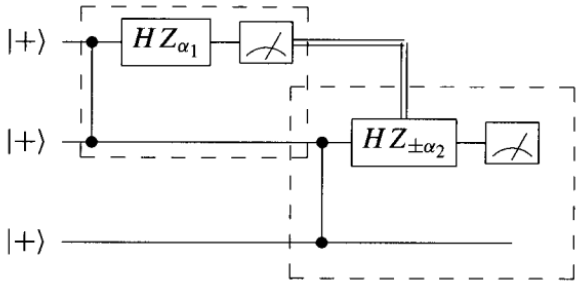

FIGURE 2.6: Reduction of figure 2.5 using definition

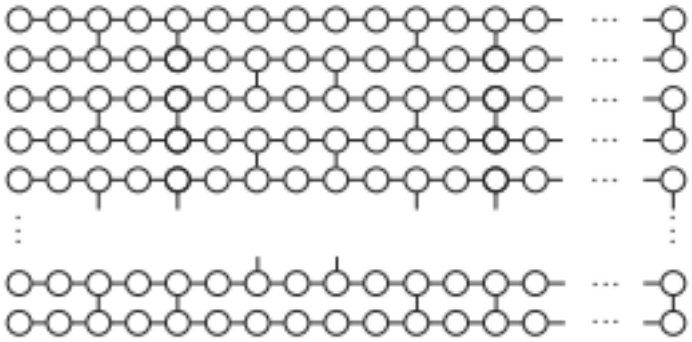

FIGURE 2.7: The brickwork state, $\mathcal{G}_{n\times m}$

function of the output of measurement. Since this is know, it is accounted for in further gates that will be acted upon.

2.3 Brickwork States

For the purpose of secure assisted quantum computation that we will look at next, we will use a special structure of cluster states. We construct Brick work states with a special uniform underlying graph structure. This structure has shown to be universal, that is any operator can be constructed using this underlying graph structure. Using this also ensures that when Alice sends bob the encrypted cluster state, Bob cannot even learn anything about the underlying graph structure too, apart from the initial state of each qubit (which will be encrypted).

2.3.1 Definition

A brickwork state $\mathcal{G}_{n\times m}$, where $m \equiv 5 \pmod 8$, is an entangled state of $n \times m$ qubits constructed as follows (see also Figure 2.7):

1. Prepare all qubits in state $|+\rangle$ and assign to each qubit an index (i, j), i being a column $(i \in [n])$ and j being a row $(j \in [m])$
2. For each row, apply the operator CTRL- Z on qubits (i, j) and $(i, j+1)$ where $1 \leq j \leq m-1$
3. For each column $j \equiv 3 \pmod 8$ and each odd row i, apply the operator CTRL- Z on qubits (i, j) and $(i + 1, j)$ and also on qubits $(i, j + 2)$ and $(i + 1, j + 2)$
4. For each column $j \equiv 7 \pmod 8$) and each even row i, apply the operator CTRL- Z on qubits (i, j) and $(i + 1, j)$ and also on qubits $(i, j + 2)$ and $(i + 1, j + 2)$

2.3.2 Universality of Brickwork States

Theorem: The brickwork state $\mathcal{G}_{n\times m}$ is universal for quantum computation. Furthermore, we only require single-qubit measurements under the angles $\{0, \pm\pi/4, \pm\pi/2\}$, and measurements can be done layer-by-layer.

Proof: We know that $U = \{\text{CTRL} - X, H, \frac{\pi}{8}, I\}$ forms the set of universal gates. We show how brickwork states can be used to simulate all the gates in U. Figures 2.8,2.9,2.10 and 2.11 present the required implementations. These images have been sourced from [3].

Also note that any pattern can be rewritten in a standard form, where all the preparation and entangling command are performed only at the beginning of the computation, because of the following commutation relations (where E_{ij} represents the CRTL-Z operator):

$$E_{ij}X_i^s = X_i^s Z_j^s E_{ij}$$
$$E_{ij}Z_i^s = Z_i^s E_{ij}$$
$$E_{ij}Z_i^s(\alpha) = Z_i^s(\alpha) E_{ij}$$

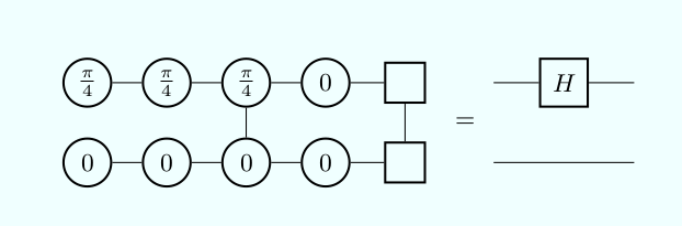

FIGURE 2.8: Implementation of a Hadamard gate.

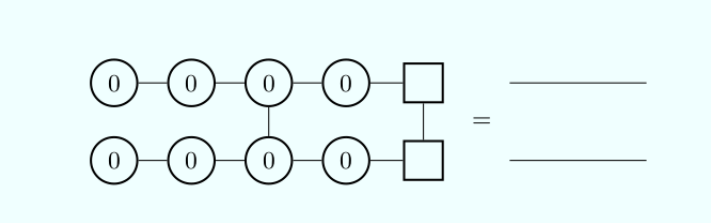

FIGURE 2.9: Implementation of a $\frac{\pi}{8}$ gate.

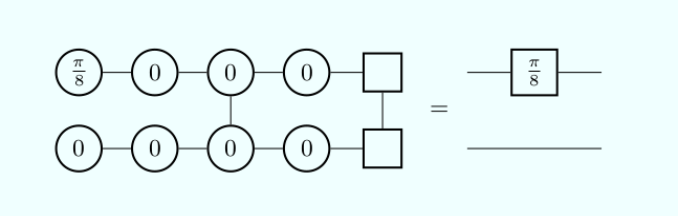

FIGURE 2.10: Implementation of the identity.

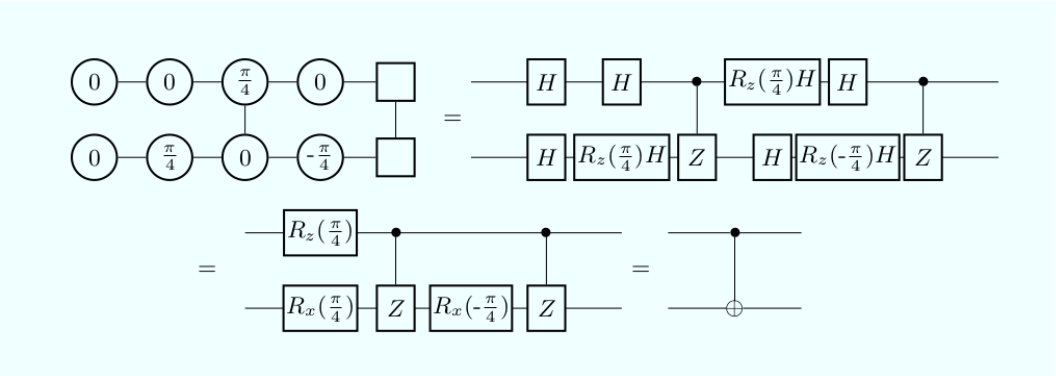

FIGURE 2.11: Implementation of a CTRL-X.

We can verify the above implementations by algorithmically converting to the circuit model using the definitions provided earlier. Using symmetry, the gates could be shifted from the first to the second qubit. These patterns can be tiled to construct any to implement any circuit using U as a universal set of gates.

Chapter 3

Protocol for Blind Computation without requiring quantum memory

Using the formalism presented in the previous chapter, we now present a formalism in which Alice's (client's) inputs, outputs, and computation remain secure as Bob computes the required function for Alice. This protocol is more powerful because it does not require Alice to be able to perform any quantum gates or have quantum memory. This protocol [2] requires a two way classical channel that will drive the interactive protocol forward.

3.1 Outline of the Protocol

Alice has a classical computer restricted to modulo 8 arithmetic, augmented with the power to prepare single qubits randomly chosen in

$$\left\{1/\sqrt{2}\left(|0\rangle + e^{i\theta}|1\rangle\right) | \theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4\right\}$$

Bob has a universal quantum computer. Shared quantum and classical channels are required. The protocol is interactive, it consists of rounds of communication between Alice and Bob as a "feedback" mechanism. This protocol is given in the formalism of measurement based quantum computation, described earlier.

We have proved that cluster states are universal, and any MBQC (and any quantum computation) can be performed using cluster states. We use them to make our protocol secure such that Bob does not even know the underlying graph structure used. The only information Bob gets is the upper bound on the size of computation.

Since MBQC is universal, we can express the computation Alice seeks to perform as as a measurement pattern on a brickwork state. The protocol can then be viewed as a distributed version of a measurement based quantum computation such that :

1. Alice prepares the individual qubits on the cluster state
2. Bob does the entanglement according to the brickwork state and measurements as specified by Alice
3. Alice computes the classical feedforward mechanism

If Alice is computing a classical function, the protocol finishes when all qubits are measured. If she is computing a quantum function, Bob returns to her the final qubits. A modification of the protocol also allows Alice's inputs to be quantum.

3.2 Main Protocol

U	Unitary operator to be implemented
$\mathcal{G}_{n\times m}$	brickwork state
$\|\psi_{x,y}\rangle$	qubit in $\mathcal{G}_{n\times m}$ indexed by a column $x \in \{1,\dots,n\}$ and a row $y \in \{1,\dots,m\}$
$\phi_{x,y}$	measurement angle for $\|\psi_{x,y}\rangle$ to compute U
$X_{x,y}$	set of X dependencies for the node (x,y)
$Z_{x,y}$	set of Y dependencies for the node (x,y)
$s^X_{x,y}$	$= \oplus_{i\in X_{x,y}} s_i$, parity of all measurement outcomes for qubits in $X_{x,y}$
$s^Z_{x,y}$	$= \oplus_{i\in Z'_{x,y}} s_i$, parity of all measurement outcomes for qubits in $Z_{x,y}$
$\phi'_{x,y}$	$= (-1)^{s^X_{x,y}}\phi_{x,y} + s^Z_{x,y}\pi$, actual measurement angle

3.2.1 Blindness

Definition Let P be a quantum delegated computation on input X and let $L(X)$ be any function of the input. We say that quantum delegated computation protocol is blind while leaking at most $L(X)$ if, on Alice's input X, for any fixed $Y = L(X)$, the following two hold when given Y:

1. The distribution of the classical information obtained by Bob in P is independent of X.

2. Given the distribution of classical information described in 1, the state of the quantum system obtained by Bob in P is fixed and independent of X

Protocol 1 Universal Blind Quantum Computation

1. **Alice's preparation**
   For each column $x = 1, \ldots, n$
   For each row $y = 1, \ldots, m$

   1.1 Alice prepares $|\psi_{x,y}\rangle \in_R \{|+_{\theta_{x,y}}\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta_{x,y}}|1\rangle) \mid \theta_{x,y} = 0, \pi/4, \ldots, 7\pi/4\}$ and sends the qubits to Bob.

2. **Bob's preparation**

   2.1 Bob creates an entangled state from all received qubits, according to their indices, by applying CTRL-Z gates between the qubits in order to create a brickwork state $\mathcal{G}_{n\times m}$.

3. **Interaction and measurement**
   For each column $x = 1, \ldots, n$
   For each row $y = 1, \ldots, m$

   3.1 Alice computes $\phi'_{x,y}$ where $s^X_{0,y} = s^Z_{0,y} = 0$.

   3.2 Alice chooses $r_{x,y} \in_R \{0, 1\}$ and computes $\delta_{x,y} = \phi'_{x,y} + \theta_{x,y} + \pi r_{x,y}$.

   3.3 Alice transmits $\delta_{x,y}$ to Bob. Bob measures in the basis $\{|+_{\delta_{x,y}}\rangle, |-_{\delta_{x,y}}\rangle\}$.

   3.4 Bob transmits the result $s_{x,y} \in \{0, 1\}$ to Alice.

   3.5 If $r_{x,y} = 1$ above, Alice flips $s_{x,y}$; otherwise she does nothing.

We now prove that **Protocol 1** [1] is blind leaking at most (n,m), the dimension of the brickwork state.

Alice's input consists of

$$\phi = (\phi_{x,y} | x \in [n], y \in [m])$$

with the actual measurement angles

$$\phi' = \left(\phi'_{x,y} | x \in [n], y \in [m]\right)$$

being a modification of ϕ that depends on previous measurement outcomes. Let the classical information that Bob gets during the protocol be

$$\delta = (\delta_{x,y} | x \in [n], y \in [m])$$

Independence of Bob's Classical information: for a uniformly random chosen $\theta_{x,y}$:

$$\theta'_{x,y} = \theta_{x,y} + \pi r_{x,y}$$

[1] The Protocol has been retrieved from [3] (https://arxiv.org/abs/0807.4154v3)

and

$$\theta' = \left(\theta'_{x,y} | x \in [n], y \in [m]\right).$$

$\delta = \phi' + \theta'$, with θ' is uniformly random (and independent of ϕ and/or ϕ'). This implies the independence of δ and ϕ.

Independence of Bob's quantum information:

For each qubit of A, one of the following two has occurred:

1. $r_{x,y} = 0$ so $\delta_{x,y} = \phi'_{x,y} + \theta'_{x,y}$ and $|\psi_{x,y}\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + e^{i\left(\delta_{x,y} - \phi'_{x,y}\right)}|1\rangle\right.$
2. $r_{x,y} = 1$ so $\delta_{x,y} = \phi'_{x,y} + \theta'_{x,y} + \pi$ and $|\psi_{x,y}\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - e^{i\left(\delta_{x,y} - \phi'_{x,y}\right)}|1\rangle\right.$

Because $r_{x,y}$ is uniformly random, and independent of ϕ, without the knowledge of $r_{x,y}$, the system is in maximally mixed state. Hence Bob can get no information out of it.

3.3 Modification: Quantum inputs and outputs

3.3.1 Quantum Inputs

The main protocol can be extended to allow for quantum inputs easily if we allow Alice to be able to apply X and Z gates. Alice then encrypts her qubit by acting $X^l Z^m$ where l, m are random coin tosses. Alice then sends this maximally mixed state to Bob for computation. The rest of the protocol is same as before. This protocol is blind because Bob only receives a maximally mixed state as inputs from Alice.

3.3.2 Quantum outputs

Instead of measuring the last layer of qubits, Bob returns it to Alice. Since, at each step in **Protocol 1**, the qubits are one time padded, **Protocol 2** [2] is correct and private.

3.4 Detecting a cheating server

We now present an authentication technique which enables Alice to detect an interfering Bob with overwhelming probability. That is, either an interfering Bob is corrected and not detected, or is detected with overwhelming probability.

[2] The Protocol has been retrieved from [3] (https://arxiv.org/abs/0807.4154v3)

Protocol 2 Universal Blind Quantum Computation with Quantum Outputs

1. **Alice's auxiliary preparation**
   For each column $x = 1, \ldots, n-1$
   For each row $y = 1, \ldots, m$

   1.1 Alice prepares $|\psi_{x,y}\rangle \in_R \{|+_{\theta_{x,y}}\rangle \mid \theta_{x,y} = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4\}$ and sends the qubits to Bob.

2. **Alice's output preparation**

   2.1 Alice prepares the last column of qubits $|\psi_{n,y}\rangle = |+\rangle$ $(y = 1, \ldots, m)$ and sends the qubits to Bob.

3. **Bob's preparation**

   3.1 Bob creates an entangled state from all received qubits, according to their indices, by applying CTRL-Z gates between the qubits in order to create a graphstate $\mathcal{G}_{n\times m}$.

4. **Interaction and measurement**
   For each column $x = 1, \ldots, n-1$
   For each row $y = 1, \ldots, m$

   4.1 Alice computes $\phi'_{x,y}$ where $s^X_{0,y} = s^Z_{0,y} = 0$ for the first column.

   4.2 Alice chooses $r_{x,y} \in_R \{0, 1\}$ and computes $\delta_{x,y} = \phi'_{x,y} + \theta_{x,y} + \pi r_{x,y}$.

   4.3 Alice transmits $\delta_{x,y}$ to Bob.

   4.4 Bob measures in the basis $\{|+_{\delta_{x,y}}\rangle, |-_{\delta_{x,y}}\rangle\}$.

   4.5 Bob transmits the result $s_{x,y} \in \{0, 1\}$ to Alice.

   4.6 If $r_{x,y} = 1$ above, Alice flips $s_{x,y}$; otherwise she does nothing.

5. **Output Correction**

   5.1 Bob sends to Alice all qubits in the last layer.

   5.2 Alice performs the final Pauli corrections $Z^{s^Z_{n,y}} X^{s^X_{n,y}}$.

Using Trap wires: For functions with classical input and output, Alice places N randomly placed trap wires with known random state $|0\rangle$ or $|1\rangle$. If Bob interferes,either his interference has no effect on the classical output or there is at least 50% chance that Alice detects incorrect value on one of the trap wires. This protocol is repeated s times, and if the function output is same all s times and Bob is not caught cheating, Alice accepts or rejects. There is an exponentially small (2^{-s}) probability that Alice accepts wrong result. The blind computation protocol is extended by allowing Alice to instruct Bob to measure specific qubits within the brickwork state in the computational basis at regular intervals. These qubits are chosen at regular spacial intervals so that no information about the structure of the computation is revealed.

Protocol 3 Blind Quantum Computing with Authentication (classical input and output)

1. Alice chooses $\mathbb{C}$, where $\mathbb{C}$ is some $n_{\mathbb{C}}$-qubit error-correcting code with distance $d_{\mathbb{C}}$. The security parameter is $d_{\mathbb{C}}$.

2. In the circuit model, starting from circuit for U, Alice converts target circuit to fault-tolerant circuit:

2.1 Use error-correcting code $\mathbb{C}$. The encoding appears in the initial layers of the circuit.

2.2 Perform all gates and measurements fault-tolerantly.

2.3 Some computational basis measurements are required for the fault-tolerant implementation (for verification of ancillae and non-transversal gates). Each measurement is accomplished by making and measuring a *pseudo-copy* of the target qubit: a CTRL-X is performed from the target to an ancilla qubit initially set to $|0\rangle$, which is then measured in the Z-basis.

2.4 Ancilla qubit wires are evenly spaced through the circuit.

2.5 The ancillae are re-used. All ancillae are measured at the same time, at regular intervals, after each fault-tolerant gate (some outputs may be meaningless).

3. Within each encoded qubit, permute all wires, keeping these permutations secret from Bob.

4. Within each encoded qubit, add $3n_T$ randomly interspersed *trap* wires, each trap being a random eigenstate of X, Y or Z (n_T of each). For security, we must have $n_T \propto n_{\mathbb{C}}$; for convenience, we choose $n_T = n_{\mathbb{C}}$. The trap qubit wire (at this point) does not interact with the rest of the circuit. The wire is initially $|0\rangle$, and then single-qubit gates are used to create the trap state. These single-qubit gates appear in the initial layers of the circuit.

5. Trap qubits are verified using the same ancillae as above: they are rotated into the computational basis, measured using the pseudo-copy technique above, and then returned to their initial basis.

6. Any fault-tolerant measurement is randomly interspersed with verification of $3n_T$ random trap wires. For this, identity gates are added as required.

7. For classical output, the trap wires are rotated as a last step, so that the following measurement in the computational basis is used for a final verification.

8. Convert the whole circuit above to a measurement-based computation on the brickwork state, with the addition of regular Z-basis measurements corresponding to the measurements on ancillae qubits above. Swap and identity gates are added as required, and trap qubits are left untouched.

9. Perform the blind quantum computation:

9.1 Execute **Protocol 1**, to which we add that Alice periodically instructs Bob to measure in Z-basis as indicated above.

9.2 Alice uses the results of the trap qubit measurements to estimate the error rate; if it is below the threshold (see discussion in the main text), she accepts, otherwise she rejects.

Chapter 4

Efficient Universal Blind Quantum Computation

We now look at an protocol for blind quantum computation which is efficient in terms of the communication between Alice and Bob [5]. In contrast to the protocol in Chapter 1, this does not require transfer of computational qubits between Alice and Bob. While the protocol in Chapter 2, which uses measurement based computation is optimal in communication complexity, this protocol has the advantage of being described in circuit model of computation, which is more intuitive. However, this protocol requires transfer of a"quantum register" as a trade off.

4.1 Main Protocol

In this protocol, Alice has limited capabilities, as before. She can generate a qubit in the standard basis ($|0\rangle, |1\rangle$) and the complementary basis ($|+\rangle, |-\rangle$). She also has a quantum register of size $O(\log_2(N))$, where n is the number of qubits. Bob has a universal quantum computer. Alice instructs Bob to perfrom an arbitrary J step computation by giving him $Jlog_2(G)$ bits, where $\mathcal{G} = N(N+2)$ is the set of universal gates. This is done using a pre-decided set of codes such that each number $n \in (0...(N(N+2)$ correspond to an instruction of the form "Apply $\mathcal{A}$ gate to $\mathcal{B}$ qubit(s)". The main protocol is described in 4. It is important to note that Bob's action is implemented as an oracle and he does not measure the register in every round. For each state $|\Psi\rangle_M$ of M and $|\phi\rangle_A = \sum_n \eta_n |n\rangle_A$ of A he performs the control-unitary gate $U_{\text{Bob}} = \sum_n |n\rangle\langle n| \otimes U_n$ which yields the mapping

$$|\phi\rangle_A \otimes |\Psi\rangle_{\mathcal{M}} \rightarrow \sum_n \eta_n |n\rangle_A \otimes U_n |\Psi\rangle_{\mathcal{M}}$$

Protocol 4 Efficient Universal Blind Quantum Computation

Setup: Bob has access to quantum computation and a quantum memory which can hold N qubits assigned an initial state(say the vector $|0\rangle^{\otimes N}$) and a set $\mathcal{G}$ of $O(\text{poly}(N))$ universal gates he can apply to them.

1. Bob initializes his qubits in $|0\rangle^{\otimes N}$.
2. jth computation step: Alice sends Bob a register A of $O(\log_2 N)$ qubits. It (randomly) *either* contains the qubit to which a gate is to be applied (e.g. $|3\rangle_A$ means "apply the Hadamard gate to qubit #3"), *or* it contains a decoy (e.g. $|n\rangle_A + |n'\rangle_A$).
3. Bob uses Alice's register to establish to which qubits to apply the gates of the universal set: e.g. Bob's action $U_{Bob}\,|3\rangle_A\,|\Psi\rangle_{\mathcal{M}}$ applies the Hadamard gate to Bob's qubit #3 (here $|\Psi\rangle_{\mathcal{M}}$ represents the global state of Bob's qubits). If the register contains a decoy, he will apply the gates to a *superposition* of registers.
4. Bob sends the register A back to Alice.
5. If Alice knows that the register A is unentangled from Bob's qubits, she measures it [case (a), see text]. Otherwise she sends it back to Bob as one of the successive instructions until it becomes unentangled, and then measures it [case (b)]. If the measurement result matches the state she had initially prepared, she proceeds to the next step of the computation through point 2, otherwise she halts the computation.
6. At the end of the computation, Bob measures the computation qubits and reveals the computation result (possibly encrypted, see text).

4.2 Correctness of the Protocol

4.2.1 Blindness

Blindness is achieved by randomly interspersing the instructions with lures. This only adds a small overhead in terms of size of memory and communication. A quantum lure is created by using a superposition of at least two instructions being sent to Bob. This can be done using Alice's limited ability of being able to generate qubits only in standard and complementary basis. For example, in the case of a two-qubit register a quantum lure as a superposition of the instructions $|n=0\rangle_A$ and $|n=2\rangle_A$ is prepared by the factorized state $|+,0\rangle_A = |0,0\rangle_A + |1,0\rangle_A$. After the operation, when Bob sends teh register back to Alice, she checks her lures to know if Bob has attempted to "read her qubits or not. The probability that Bob can cheat for j computational steps without being detected by Alice decreases exponentially as $p^{\gamma j}$, where γ is the average fraction of instructions that are lures and p is the probability of being detected on a single lure. Thus Bob can cheat only for a constant number of steps before being detected by Alice. Similarly, any eves-dropper who can access the register, will also be detected similarly.

4.2.2 Completeness

Assuming the setup as mentioned in the Protocol above, let's try to fix a notation. We can assume for instance that $\mathcal{G}$ contains $G = N(N+2)$ elements including a Hadamard and a $\pi/8$ gate for each qubit and a C-NOT for each (ordered) couple of qubits of $\mathcal{M}$. In this scenario, Alice can instruct him to perform an arbitrary computation by telling him which element of $\mathcal{G}$ he must apply at each step of the computation.

4.3 Efficiency of the Protocol

4.3.1 Communication Complexity

Before considering a general case. let's try to look at an example. Alice can send a number n between 0 and $G-1$, which Bob interprets in the following way: 0 to $N-1$ means "act with a Hadamard gate on qubit n", N to $2N-1$ means "act with a $\pi/8$ gate on qubit $n-N$", and any other number means "act with a C-NOT gate using n_1 as control and n_2 as target", where n_1, n_2 are such that $n = n_1 N + n_2 + 2N$. We can easily come up with such a setup for any computation we might need. For this, she needs a $\log_2 G \simeq O(\log_2 N)$ bit register. She can thus instruct Bob to perform an arbitrary J-step computation by giving him $J \log_2 G \simeq O(J \log_2 N)$ bits. This communication cost is optimal, because a programmable quantum computer requires a program register of dimension at least as large as the number of possible computations that it can perform(since at each step Bob can apply one out of G possible gates, in our case such number is indeed equal to G^J, which requires $J \log_2 G$ bits). Hence, this protocol is optimal in terms of the number of exchanged bits of information between Alice and Bob, as a universal quantum computer cannot have a software register of less than $O(J \log_2 N)$ bits. The blind protocol simply requires these to be qubits instead of bits. This requires a small overhead composed by the decoys and the final one-time-pad encoding. Hence, the total communication complexity is then still $O(J \log_2 N)$ qubits.

4.3.2 Computational and Memory Complexity

The running time for this protocol will be linear (a constant fraction γ of the operations will be decoy operations), so the algorithm running time will be $O((1+\gamma)J)$. Here we can choose the fraction γ to be arbitrarily small when $J \to \infty$ as it can scale as $J^{-1/2}$, ignoring logarithmic corrections.

The memory required in terms of qubits is linear. We need a constant fraction λ of qubits to be devoted to Alice's decoy operations, so that the qubit cost goes from N to $N(1+\lambda)$. In terms

of gates, there is no overhead with respect to what is necessary for a universal programmable quantum computer. The only difference is that Alice's register is encoded in quantum bits instead of a sequence of classical numbers. This means that Bob needs controlled-swaps that are controlled by a quantum register instead of a classical register.

Summarizing, modulo logarithmic or constant corrections, this protocol does not require any significant computational or communication overheads over what is necessary for a universal programmable quantum computer with the only substantial difference being that Alice's registers is encoded in qubits instead of bits.

Chapter 5

Secure Quantum Machine Learning

Quantum Machine Learning ([4] is a good review on QML) is touted to be the next major leap in the field of Machine Learning, although it doesn't seem likely that everyone will have access to Quantum Resources, blind computation provides a method for computations to be done at a central server securely and the users can provide their information and receive results of computation back in an encrypted fashion so that an intermediary eavesdropper cannot access any secure information.We describe below a protocol to perform a very basic Quantum Machine Learning protocol in a secure and blind manner[6]

5.1 Description of Protocol

Most Quantum Machine Learning Algorithms revolves around finding the distance between two states, one being the states we want to classify and the other being the label states.Let U_a and V_a be two label states and U be the state we wish to classify then if,

$$|U_a - U| < |V_a - U|$$

we classify the state to the label of U_a similarly if

$$|V_a - U| < |U_a - U|$$

we classify it as V_a,we know that,

$$|U - V| = \sqrt{|U|^2 + |V|^2 - 2\langle U|V\rangle}$$

Therefore we can see that in the calculation of distance between two states the crucial term to calculate is the term $\langle U|V\rangle$ since the other terms are easily calculable classically. Therefore, this

protocol deals with the calculation of $\langle U_a|U\rangle$ and $\langle V_a|U\rangle$ which can therefore help us classify the states. Given below is a detailed description of the protocol.

5.1.1 Initial Resources and setup.

Let Alice and Bob be the two parties involved in the computation Alice being the client and Bob being the Central server.Given below is a diagram describing Alice an Bob's setup

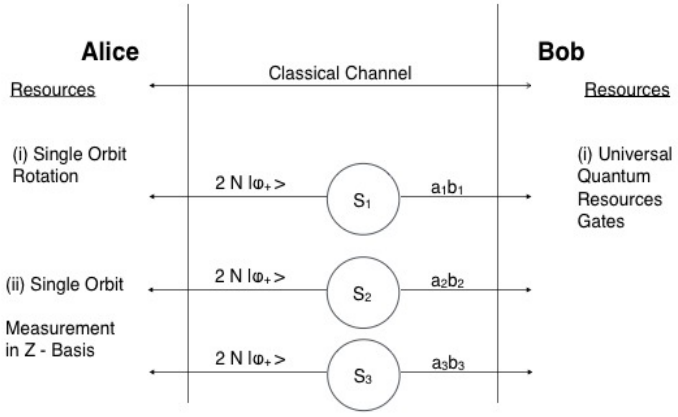

FIGURE 5.1: Resource distribution between Alice and Bob shown in a diagram

1. Alice and Bob share 3 Quantum channels a_1b_1,a_2b_2 and a_3b_3 as well as a classical channel.
2. There are three entanglement sources S1 S2 S3 which produce a Bell pair between Alice and Bob,Alice and Bob share 2N Qubit pairs in each channel.
3. Alice has single Qubit Rotations and Measurement Apparatus while Bob has access to universal set of resources.
4. Alice intends to calculate the overlap between two states i.e. $\langle U|V\rangle$
5. Let vector $U = \alpha\,|0\rangle + \beta\,|1\rangle$ and $V = \gamma\,|0\rangle + \delta\,|1\rangle$
6. The task is to calculate securely and blindly $\alpha^*\gamma + \beta^*\delta$

5.1.1.1 Checking Phase

The initial step of the Protocol is to check whether the channels are secure and there are no eavesdropper in the channel acquiring the information,as mentioned before there 2N shared

bell pairs $|\phi_+\rangle$ in each of the channels,we use N of these 2N qubits to check the security of the channel,the method used for this is that Alice and bob decide on a basis either the X-basis or the Z-basis for each of the N qubits and measure the qubits in that basis on their side. They then compare the measurement outcomes since they share a bell pair state $|\phi_+\rangle$ they must have the same measurement outcomes in the same basis,if an eavesdropper interferes the channel by either measuring or altering the state the measurement outcomes will not match.

5.1.2 Transmission of state from Alice to Bob

This part can further be divided into 3 sub parts which is assigned to each of the 3 Quantum Channels shared between Alice and Bob.

1. Alice randomly measures the N Qubits of Channel a1b1 either the X-basis or the Z-basis without revealing the information to bob,this leads to Bob's qubits getting collapsed to a string of $|0\rangle|+\rangle|+\rangle|1\rangle|0\rangle|-\rangle|+\rangle|-\rangle$..... N Qubits this will act as a code for our information transmission.

2. The Qubits of channel a2b2 are rotated by an operation given by R1,which is defined by

$$R1|0\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$R1|1\rangle = \beta|0\rangle - \alpha|1\rangle$$

Action of this rotation leads to the following combined state,

$$\{R1 \otimes I\}|\phi_+\rangle \implies \{R1 \otimes I\}\left[\frac{|00\rangle + |11\rangle}{2}\right] \implies \left[\frac{\alpha|00\rangle - \alpha|11\rangle + \beta|01\rangle + \beta|10\rangle}{2}\right]$$

Therefore,

$$\{R1 \otimes I\}|\phi_+\rangle = |0\rangle[\alpha|0\rangle + \beta|1\rangle] + |1\rangle[\beta|0\rangle - \alpha|1\rangle]$$

Now,Alice measure her qubits in the $[|0\rangle, |1\rangle]$ which leads to Bob's state collapsing to either $\alpha|0\rangle + \beta|1\rangle$ if measurement is 0 or $\beta|0\rangle - \alpha|1\rangle$ if the measurement is 1,now Alice shares her measurement result to Bob,after which Bob can alter his state to obtain the state $|U\rangle = \alpha|0\rangle + \beta|1\rangle$

3. Similarly Bob can obtain $|V\rangle$ using the qubits on the channel a3b3.

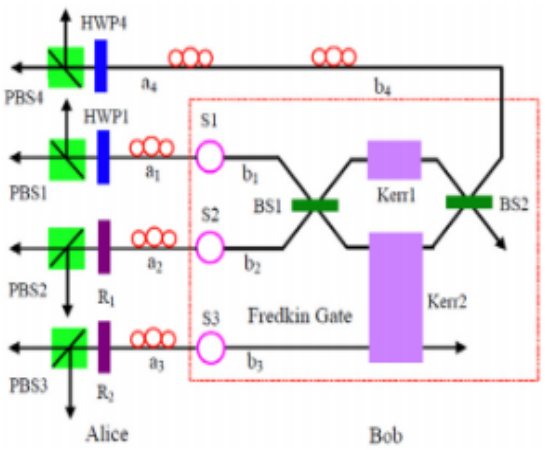

FIGURE 5.2: Schematic showing the series of operations taking place in the protocol

5.1.3 Calculation of $\langle U|V\rangle$

Bob acts as a Fredkin gate,Controlled swap gate between all three of his Qubits $[a1b1 \otimes a2b2 \otimes a3b3]$ this can lead to two possibilities, If control qubits is $[|0\rangle, |1\rangle]$

$$|0\rangle |U\rangle |V\rangle \xrightarrow{Fredkin} |0\rangle |U\rangle |V\rangle$$

$$|1\rangle |U\rangle |V\rangle \xrightarrow{Fredkin} |0\rangle |V\rangle |U\rangle$$

If control qubits is $[|+\rangle, |-\rangle]$

$$|\pm\rangle |U\rangle |V\rangle \xrightarrow{Fredkin} \frac{|+\rangle |U\rangle |V\rangle \pm |-\rangle |V\rangle |U\rangle}{\sqrt{2}}$$

Now Bob returns the control qubit back to alice and since Alice knows the Polarization of the Qubit,she can reject all the $[|0\rangle, |1\rangle]$ qubits and in case of the $[|+\rangle, |-\rangle]$ Alice performs measurement in the X-basis the probability of measurement of + is given by

$$P_+ = \frac{1 + |\langle U|V\rangle|^2}{2}$$

Therefore if the task is run for a statistically large no. N we can obtain the value of $\langle U|V\rangle$ using the probability of measurement in the state $|+\rangle$

5.2 Blindness of Protocol

1. The blindness of the protocol from an external eavesdropper is checked in the checking phase if there are any eavesdroppers in the channel they will be detected and the channel will be recaptured.

2. The reliability of the channel can also be checked when the control qubit is returned from Bob to Alice,since Alice is aware of the polarization she can check the qubit my measuring it in it's respective basis and comparing them to the initial measurement outcomes.

3. Blindness from Bob is ensured by the fact that Bob only has access to $|U\rangle, |V\rangle$ but not the values of $|U|, |V|$ so he cannot obtain the learning operations completely.

5.3 Conclusion

The above protocol gives a protocol to perform Client-Server based Quantum Machine Learning,The protocol can easily be extended to higher dimensions,A similar protocol can be proposed in which there exists a central Database who has access to the states,the Protocol is just a small modification of the original protocol with the operations being distributed between the Central Server,Alice and Bob. Given below is a diagram for the proposed network.

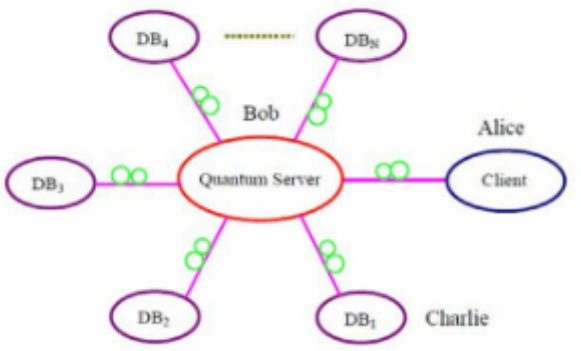

FIGURE 5.3: Schematic for a Modified protocol with a Database-Client-Server-Topology.

This protocol provides a stepping stone in the direction of future Quantum Big-Data operations.

Conclusion

We have reviewed protocols that allow non - trusting parties to do computation collaboratively in a secure and blind manner. Although we started with a simplistic algorithm, we were able to introduce the ideas of 'quantum encryption' and show that using X and Z gates (and classical keys) one can generate a maximally mixed state. We presented two other efficient protocols in terms of communication. All the three protocols satisfied the three requirements:

1. Universality : Can run any quantum algorithm
2. Blindness : Bob does not get to know the input or the function that is to be computed with an exponentially large probability
3. Verifiablity : Alice can detect an uncooperative Bob

The three protocols make different trade-offs. While the algorithm by Childs [3] requires transfer of n qubits, it required the transfer only once, irrespective of the depth of the of the computation. The algorithm by Giovannetti, et al [5] requires a quantum memory of $\log N$ size, but it requires transfer for each round. It also requires Bob to have the instruction to be implemented as an oracle, which is not simple. In comparison, Childs required Alice to have X and Z gates. The MBQC protocol [2], although requires a large number of qubits, does not require them to be coherent for a long time, as they are measured.

We have also presented a protocol such that a distributed quantum machine learning can be implemented in a secure manner. This is going to be relevant and quantum computing and quantum machine learning picks up pace. Measurement Based quantum computing presents us a fresh approach to think of quantum algorithms and quantum computation. Although it involves a larger number of qubits, the stability expectation is much lower and certain decoherence shall be acceptable. We present simulation of the MBQC secure protocol in Appendix.

Finally, the MBQC model has the scope of being extended to a blind neural network model. With the blindness and security in place, enabling multiple connections to very node will pave way to future work.

Bibliography

[1] Michael A.Nielsen. "Cluster-state quantum computation". In: *Reports on Mathematical Physics* 57 (1 2006), pp. 147–161.

[2] Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. "Universal Blind Quantum Computation". English. In: *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS '09)*. Annual Symposium on Foundations of Computer Science. United States: Institute of Electrical and Electronics Engineers (IEEE), 2009, pp. 517–526. ISBN: 978-1-4244-5116-6. DOI: 10.1109/FOCS.2009.36.

[3] Andrew M. Childs. "Secure Assisted Quantum Computation". In: *Quantum Info. Comput.* 5.6 (Sept. 2005), pp. 456–466. ISSN: 1533-7146.

[4] Vedran Dunjko and Hans J. Briegel. *Machine learning & artificial intelligence in the quantum domain*. 2017. arXiv: 1709.02779 [quant-ph].

[5] Vittorio Giovannetti et al. "Efficient Universal Blind Quantum Computation". In: *Phys. Rev. Lett.* 111 (23 Dec. 2013), p. 230501. DOI: 10.1103/PhysRevLett.111.230501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.111.230501.

[6] Yu-Bo Sheng and Lan Zhou. "Distributed secure quantum machine learning". In: *Science Bulletin* 62.14 (2017), pp. 1025–1029. ISSN: 2095-9273. DOI: https://doi.org/10.1016/j.scib.2017.06.007. URL: http://www.sciencedirect.com/science/article/pii/S2095927317303250.