BLIND QUANTUM COMPUTATION

CHARLES HERDER

1. Introduction and Background

While Quantum Computation is easily motivated by the interesting computational problems that can be solved (both cryptographically and otherwise) that currently cannot be solved with a classical computer, other advantages can be obtained by leveraging quantum effects. For example, Quantum Key Distribution (QKD) offers a cryptographically secure method of distributing keys that does not rely on computational assumptions for security.[BB84] This recognition opened the door to many other quantum protocols that provide unconditional security based on the laws of physics rather than conjectures on computational hardness. One exciting example of such an application is the idea of Universal Blind Quantum Computation.[ABE08][BFK09]

2. Blind Quantum Computation as a Quantum Interactive Proof Model

One initial discussion of Blind Quantum Computation is due to Aharonov, Ben-Or, Eban.[ABE08] In analyzing quantum authentication schemes, they also recognized that if Alice has n qubits in state ρ, applies random Pauli operators P_n to each qubit, and then sends the qubits to Bob, Bob can only observe a maximally mixed state, since he has no knowledge of which Pauli operators have been applied.

If Bob subsequently applies some operation U (such as $X^{\otimes n}$ or $Z^{\otimes n}$), then there is a known commutation relation with the Pauli operators. Therefore, when Bob returns the modified state back to Alice, Alice can use these relations to determine a new set of Pauli operators P_n' that decrypt the state $U\rho U^\dagger$.

Unfortunately, in general, the commutation relations between arbitrary U and random Pauli matrix is not easy to calculate. The blind algorithm is described in general having Alice retrieve the encrypted data, decrypt it, run the computation herself, re-encrypt the data, and return it to Bob. Effectively, she only uses Bob as a quantum memory. As a result, for more complicated algorithms, the algorithm becomes less satisfying.

3. Measurement-Based Quantum Computation (MBQC)

Although the Quantum Interactive Proof model demonstrated the capabilities of universal blind quantum computation, there wasn't a clear transition between that model and an actual physical system. This perspective arose from an implementation using Measurement-Based Quantum Computation (MBQC).[BFK09] Before discussing how this implementation works, it is useful to briefly review MBQC.

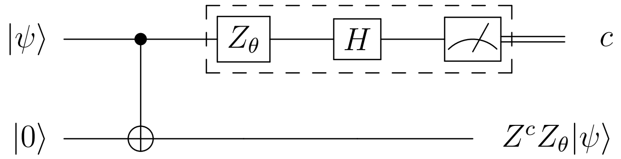
MBQC is an alternative model of quantum computation initializes a lattice of qubits (typically a "Cluster State"), and then uses only measurements to enact a universal quantum computation.[RB01] The initial state preparation does not depend on the computation, and therefore there all of the entanglement is performed at the beginning of the computation.

3.1. **MBQC and Quantum Teleportation.**

MBQC is based on the principles of quantum teleportation. Gottesman and Chuang recognized that, with the correct measurement basis, quantum teleportation could realize any primitive, and was therefore universal.[GC99] This work was extended by Raussendorf and Briegel in 2001 who recognized that all of the entangling operations could be performed prior to the beginning of a computation (creating cluster states), and then using only measurements to enact arbitrary circuits based on the principles of teleportation.[RB01]

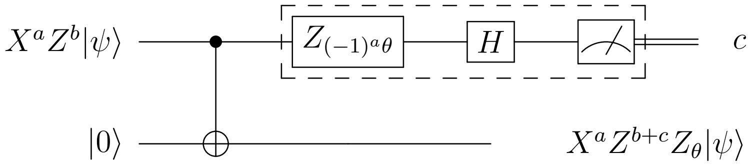
Prior to studying the actual Universal Blind Quantum Computation protocols with MBQC, it is important to understand how local Pauli operators interact with the MBQC computational model. This will give rise to intuition regarding why MBQC was the initial, natural choice for blind computation, and it will also help separate and identify which of the properties of MBQC are critical for blind quantum computation.

In the standard description of quantum teleportation, one always measures in the computational basis. By changing the basis of measurement, one effectively applies a single-qubit rotation about the Z-axis (Z_θ). This basis change is represented as several gates and a measurement grouped in the dashed box below. Note that an X rotation is easily obtained by a similar construction. It can be shown that, once X and Z rotations can be achieved, any single-qubit gate can be obtained.[CLN05]



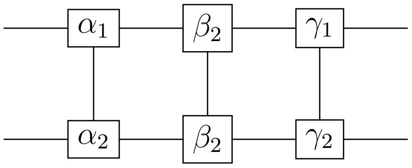
The teleported qubit also has a residual Pauli operator that depends on the result of the classical measurement. Instead of directly applying a classically controlled Pauli operator (the Z gate in the above example), MBQC adapts the subsequent measurements to incorporate and correct for these Pauli operators without explicitly applying the correcting operator. This is shown below.

In MBQC, one knows the result of the previous measurement which give rise to a and b below. By adapting the rotation to compensate for these operators, one can "commute" the local Pauli operator so that the resulting qubit is the rotated qubit multiplied by some new set of Pauli operators that depend both on the previous qubit and the classical measurement.



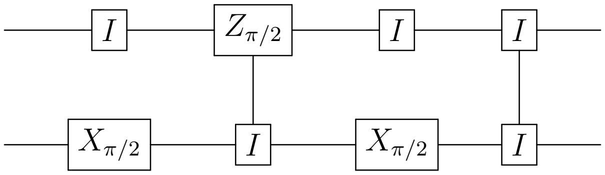
This is one of the primary reasons why MBQC is a natural choice for blind computation. There is a natural way of dealing with random Pauli operators applied to each qubit, because the random Pauli operators were already a part of the original computational protocol.

With some circuit identities, both the Z and X can be converted into circuits with the initial state being $|+\rangle$ and only Controlled-Z's between qubits. It can then be shown that a sequence of qubits initialized in this manner can perform arbitrary single-qubit rotations. Given that a qubit is consumed at each step, the circuit model becomes rather unwieldy quickly, so MBQC computations are typically represented as follows:



In the above representation, α_x and γ_x are rotations about the Z axis, and β_x is a rotation about the X axis. The lines between qubits represent the controlled-Z applied before the computation begins. A computation starts with the input data in the two far left qubits. It then measures α_1 and α_2, and the data for each qubit propagates to the right. In general, each horizontal line represents a single qubit propagation, and each vertical connection represents qubit interaction.

We already demonstrated that single qubit gates could be assembled. Now, to demonstrate universality, we only need to demonstrate how to construct a *CNOT* gate using the above method. This can be recognized as the following sequence:



Using CNOT and the arbitrary X and Z rotations, we can exactly compute any quantum function. However, it will be more useful to consider restricting X and Z rotations to $n\pi/4$, $(n \in \mathbb{Z})$ increments. In this case, we will show that MBQC is approximately universal.

4. Universal Blind Quantum Computation with MBQC

In the previous section, we demonstrated that MBQC is universal. We also showed several interesting properties about how MBQC deals with local Pauli operators that arise from the nature of quantum teleportation.

Now, let us consider how MBQC can be adapted for a blind quantum computation. The model of such a computation is again where the "client" has limited quantum resources and sends instructions to a "server" encoding the gates to be executed for the desired quantum algorithm. The server then executes the algorithm and returns the result to the client.

For a computation to be fully blind, the quantum server must not know:

- The input data
- The operations being performed
- The output data
- Any intermediate data

This means that any information that the quantum server receives must be decorrelated with the all of the actual computational data. Therefore, the above list corresponds to:

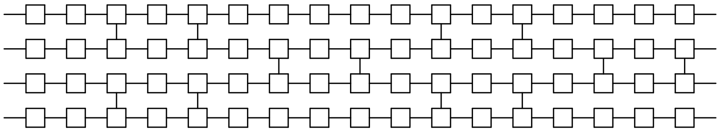
- The input quantum state sent by the client must be hidden
- The operation instructions sent by the client must be hidden
- The output quantum state sent back to the client must be hidden
- Any classical data collected by measurements during an intermediate step must have zero correlation to any data or operation being performed.

However, there is some form of data that is *always* leaked. These are the bounds of the computation. Any computation requires some physical resources from the server. In this case, those resources are physical quantum memory and number of operations. This information upper-bounds the complexity of the computation that the client wanted to execute. In the case of blind MBQC, this will has a numerical representation in terms of the size of the initial state on which the computation is performed. In fact, the number of operations is the x-dimension of the state, and the quantum memory is the y-dimension.

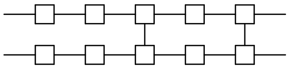
Now, consider the MBQC operation. The initial step of preparing states as $|+\rangle^{\otimes N}$ does not depend on the computation and therefore gives the server no information on the computational structure or any data. Next, applying controlled-Z operations between states also does not depend on the computation or data.

However, in an MBQC based on a cluster state, the computer must "delete" unwanted states. For example, the above *CNOT* operation could not be implemented on a cluster state without having ancillary qubits that were deleted before the computation ran. This is described in the original paper by Raussendorf and Briegel.[RB01] This initialization step reveals the structure of the computation, and is therefore not blind.

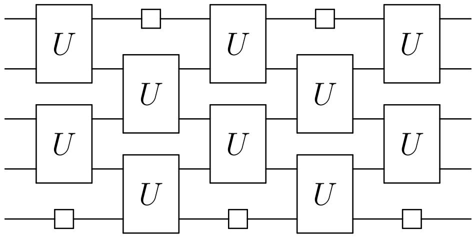
4.1. **The Brickwork States.** To resolve this issue, Broadbent, Fitzsimons, and Kashefi used a "brickwork" state.[BFK09] This state is similar to a cluster state, but having the following pattern:



The above pattern has a "universal unit cell." The problem with the cluster state was that different operations had different "shapes" on the computational surface. In the brickwork state, a universal set of qubit gates can be created that all have the same "shape" on the surface. This "universal unit cell" is shown below:

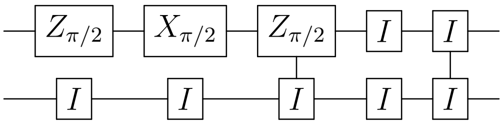


The brickwork state then becomes:

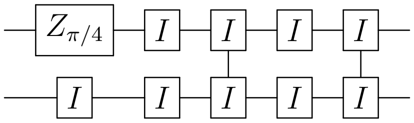


Where U is a the universal unit cell. The universality of U is shown by demonstrating that U can generate $CNOT, H, Z_{\pi/4}$.

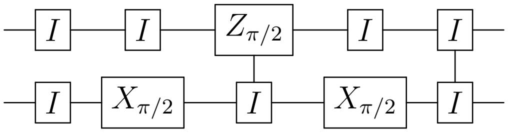
H is given by:



$Z_{\pi/4}$ is given by



$CNOT$ is given by



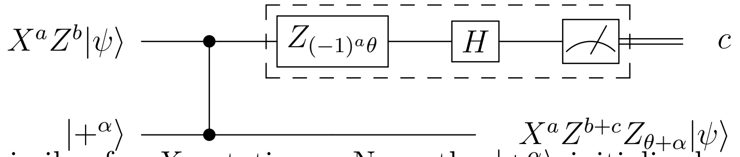
Thus, U is universal. Note that by symmetry, one can change the $CNOT$ orientation and change H or $Z_{\pi/4}$ to act on the second qubit trivially.

We have now demonstrated that by using the brickwork state, one can perform an arbitrary computation without running into the issue seen with the cluster state where one had to delete qubits based on the structure of the computation.

4.2. **Randomized Initial Phase for Qubit Operation Decorrelation.** Now, consider the second requirement for blindness: that that all operations being performed must be decorrelated with instructions given by the client. We have demonstrated how to initialize the state in a blind manner, but now we must consider how to decorrelate the adaptive measurements required for MBQC from the operations that they perform.

In the example of the Z rotation above, the angle of the measurement basis determined the amount of rotation of the operator. This measurement basis must be transmitted to the server by the client. (The basis is some $n\pi/4$ for $n \in \mathbb{Z}$). Therefore, the server can determine the operation that is being performed.

The second major innovation of Broadbent, Fitzsimons, and Kashefi was to recognize that an arbitrary Z rotation would commute through the teleportation process, adding to the rotation resulting from the server's measurement basis. This is demonstrated below. Note that $|+^\alpha\rangle = (|0\rangle + e^{i\alpha}|1\rangle)/\sqrt{2}$.

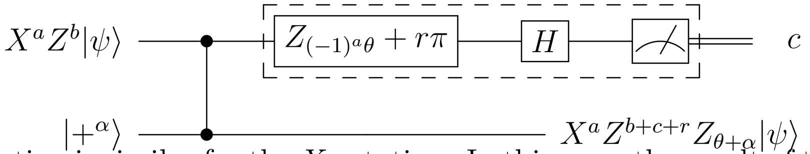


The result is similar for X rotations. Now, the $|+^\alpha\rangle$ initialized qubit is a part of the initial brickwork state. If this state is supplied to the server by the client with a random $\alpha \in n\pi/4$ for $n \in \mathbb{Z}$, the server cannot measure the value of α. The client then knows the offset that this phase will give to the rotation induced by the measurement basis. If the client sends a measurement basis rotation θ to the server that results in the overall qubit rotation being θ', then $\theta + \alpha = \theta'$. For universal quantum computation, θ' will be in $n\pi/4$ for $n \in \mathbb{Z}$. Therefore, θ is completely decorrelated from θ', since α is uniformly randomized over this set. If a different randomized α is chosen for each step of the MBQC, then each operation is completely decorrelated from the client's instructions to the server about measurement basis.

Since the server cannot know whether or not a given MBQC step applied a rotation to the qubit or not, it also cannot know what action a single universal cell has taken. Therefore, both single and multi-qubit gates are decorrellated.

4.3. **Local Paulis for Qubit State Decorrelation.** Finally, the server must be blind to the actual qubit registers. The input, intermediate, and output registers all must be considered. We discussed earlier that the MBQC method automatically compensates for arbitrary Pauli operators acting on each of the qubits, and these operators depend on the result of a random measurement.

Since the client is deciding how to measure the qubit based on the classical measurement, the client can choose a random $r \in 0, 1$, and use this to decide to rotate the measurement basis by π, which would result in a bit flip of the classical measurement. The client would incorporate this choice into the knowledge of the final qubit state, but there is no way for the server to determine what choice was made. This is demonstrated below:


$X^aZ^b|\psi\rangle$
$|+^\alpha\rangle$
$Z_{(-1)^a\theta} + r\pi$
H
c
$X^aZ^{b+c+r}Z_{\theta+\alpha}|\psi\rangle$

The construction is similar for the X rotation. In this case, the result of the classical measurement is now completely decorrelated from the effect on the local Pauli operator acting on the final qubit.

4.4. **Universal Blind MBQC Protocols.** Using the above principles, Broadbent, Fitzsimons, and Kashefi constructed a protocol for Universal Blind Quantum Computation using MBQC.[BFK09]

(1) Client's preparation

   (a) For each qubit that will be in brickwork state, prepare $|\psi\rangle = |+^\alpha\rangle = (|0\rangle + e^{i\alpha}|1\rangle)/\sqrt{2}$ for $\alpha = n\pi/4$. Send to the Server.
   (b) For each input qubit $|\psi\rangle$, apply $X^a Z_\theta|\psi\rangle$ for random $\theta = n\pi/4$ and $a = 0, 1$. Send to the Server.
   (c) For each output qubit, prepare $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$. Send to the Server
(2) Server's preparation.
   (a) Create entangled state according to the brickwork state.
(3) Interaction and measurement
   For each qubit in the brickwork state: (indexed by (x,y)).
   (a) Client computes a, b, (where the final state at (x,y) is $X^a Y^b |\psi\rangle_{(x,y)}$), with the special case of $X^a Z_\theta |\psi\rangle_{(0,y)}$ for input qubits.
   (b) Client computes $\theta = (-1)^a\theta' - \alpha + r\pi$ for desired rotation θ' and random $r \in 0, 1$. Sends to Server.
   (c) Server measures in basis $(\left|+^{\theta_{(x,y)}}\right\rangle, \left|-^{\theta_{(x,y)}}\right\rangle)$. Sends result to Client.
   (d) If r chosen above is 1, then Client flips resulting classical bit.
(4) Output Correction
   (a) Server sends all qubits in last layer to Client
   (b) Client performs Pauli corrections for each based on each qubit's associated a and b values.

BFK09 use the following definition for blindness.

Definition. *Let P be a quantum delegated computation on input $|\psi\rangle$ implementing function $U|\psi\rangle$. BFK09 defines P as blind while leaking at most $L(|\psi\rangle)$ if, on the client's input $|\psi\rangle$ and function U, the following two properties hold:*

(1) The distribution of the classical information obtained by the server in P is independent of $|\psi\rangle$.
(2) Given the distribution of classical information obtained above, the state of the quantum system obtained by the server in P is fixed and independent of $|\psi\rangle$.

They then proved the blindness of the protocol using the definition as follows:

Proof. First, recognize that due to the use of the brickwork state, no information is leaked by the shape of the computation or the initial state.

To show the independence of the server's classical information, consider the value of θ sent to the server by the client. In the protocol, $\theta = (-1)^a\theta' - \alpha + r\pi$. However, α is uniformly random, so θ must be decorrelated from θ'.

Next, consider the server's knowledge of the quantum state given the outcome of the measurement of each qubit. Because $r \in 0, 1$ is chosen randomly, for a given θ and measurement result, one of the following has occurred with 50% probability.

(1) $r = 0$: Resulting $|\psi\rangle = 1/\sqrt{2}(|0\rangle + e^{i(\theta+\alpha)}|1\rangle)$
(2) $r = 1$: Resulting $|\psi\rangle = 1/\sqrt{2}(|0\rangle - e^{i(\theta+\alpha)}|1\rangle)$

Since each of these have 50% probability, summing these terms results in the maximally mixed state. This holds regardless of the qubit being measured, so the system remains in a completely mixed state, which is fixed and independent of θ'.

□

4.5. **Protocol Variations.** Although the protocol above has been proven to be blind, there are no implications yet for either fault-tolerance or authentication. BFK09 describes a variation of the above protocol to incorporate both of the above concepts. At a high level, this is done by using traditional fault-tolerant codes and computation schemes on top of the blind protocol. Not surprisingly, because we have shown that the blind model above is universal, it can also simulate fault-tolerant circuits.

Authentication is achieved by adding trap quantum wires to the computation. For each logical qubit, a certain number of trap wires are added that are not involved with the primary computation. However, given that the server is blind to the operations being performed, it cannot tell which wires are trap wires and which aren't. This way, if the server interferes with the computation in any way, the client can detect the interference with some probability.

Do note that the client cannot tell the difference between a server's malicious interference and errors due to other non-malicious reasons. However, with enough trap wires, the client can measure the approximate error rate. The client then establishes some threshold above which he/she will reject the computation regardless of the nature of the errors.

5. Ancilla-Driven Blind Quantum Computation

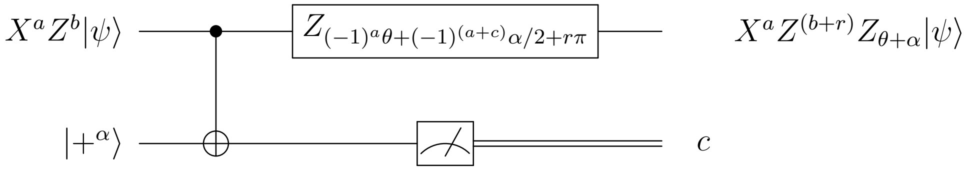
In the previous section, we described how Universal Blind Quantum Computation could be performed with MBQC. We analyzed what were the critical aspects of the Blind Quantum Computation protocol that allowed for the blindness property to hold.

We are now interested in investigating how Universal Blind Quantum Computation would work in a standard circuit model. Although these two models are equivalent, and indeed one can write out a circuit interpretation of any MBQC computation, we are interested in less trivial transformations of the blindness properties to the circuit model.

With the above analysis, we recognize that one crucial component of blindness is breaking all operations down into X and Z rotations. This way, local Pauli operators commute in a known fashion. In addition, using an ancillary qubit to add a random rotation allows the client to decorrelate the measurement basis instructions from the actual rotation.

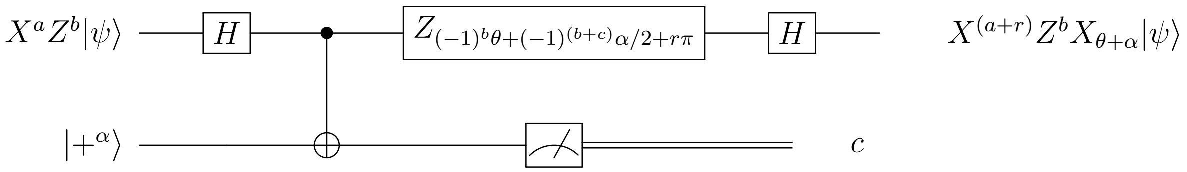
The key thing to recognize is that these features are not unique to quantum teleportation. Instead, consider a phase kickback circuit of the form below.

The Z rotation circuit is given:

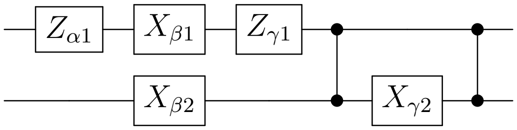

$X^aZ^b|\psi\rangle$
$Z_{(-1)^a\theta+(-1)^{(a+c)}\alpha/2+r\pi}$
$X^aZ^{(b+r)}Z_{\theta+\alpha}|\psi\rangle$
$|+^\alpha\rangle$
c

The secondary qubit has a randomized phase exactly as in the form describe for MBQC. In this case, however, instead of teleporting the qubit data onto the new qubit, we simply kick the phase back onto the original qubit. Since we are performing a Z rotation, the local X and Z Paulis affect the circuit in a similar fashion with one notable change. Now the classical measurement affects how the phase is added to the state, rather than changing the locally applied Pauli operator.

The circuit for X rotation is similar:


$X^aZ^b|\psi\rangle$
H
$Z_{(-1)^b\theta+(-1)^{(b+c)}\alpha/2+r\pi}$
H
$X^{(a+r)}Z^bX_{\theta+\alpha}|\psi\rangle$
$|+^\alpha\rangle$
c

When we switch from a teleportation-based rotation mechanism to a phase kickback, a number of results manifest. First, since there is no teleportation, the information remains on the original qubit. Therefore, a sequential computation is now better represented by the circuit model, rather than MBQC. Each operation consumes a qubit from some external reservoir, and the universal operator on 2 qubits would now be:



The proof is similar to the proof of universality for the universal operator for the brickwork MBQC case. A complete circuit has a similar format to the brickwork state described in MBQC, except now the horizontal lines are indeed qubit lines, and each operator is a gate performed on that qubit.

A more subtle result of switching to a phase-kickback mechanism arises from the meaning of the classical measurements. As mentioned above, the classical measurement no longer indicates if a Pauli operator has been modified. It indicates whether the phase of the second qubit was added or subtracted from the phase of the first.

We can now describe a complete protocol using these new mechanisms:

(1) Client's preparation
   (a) For each input qubit $|\psi\rangle$, apply $X^a Z_\theta |\psi\rangle$ for random $\theta = n\pi/4$ and $a = 0, 1$. Send to the Server.
(2) Interaction and measurement
   For each rotation in each universal gate sequence in the circuit.
   (a) For qubit y at step x, Client computes a, b, such that the qubit is in $X^a Z^b |\psi\rangle$, with the special case of $X^a Z_\theta |\psi\rangle_{(0,y)}$ for input qubits.
   (b) Client prepares $|\psi\rangle = |+^\alpha\rangle = (|0\rangle + e^{i\alpha}|1\rangle)/\sqrt{2}$ for $\alpha = n\pi/4$. Send to the Server as the ancilla qubit.
   (c) Server performs $CNOT$ as shown in the circuits below, measures ancilla qubit, and reports result c to the Client.
   (d) Client computes $\theta = (-1)^a \theta' + (-1)^c \alpha + r\pi$ for desired rotation θ' and random $r \in 0, 1$. Sends to Server.
   (e) Server completes rotation and remaining circuit operations as shown above.
   (f) If r chosen above is 1, then for Z rotation, client flips qubit's b value, for X rotation, client flips qubit's a value.
(3) Output Correction
   (a) Server sends all qubits in last layer to Client
   (b) Client performs Pauli corrections for each based on each qubit's associated a and b values.

We can prove blindness in a similar way to MBQC. Consider the same definition for blindness as incorporated by BFK09. Using this definition, we can prove that our protocol is blind.

Proof. First, remember that we are using a circuit similar to the "brickwork" state defined by BFK09. Therefore, we have shown that the shape of the circuit itself does not reveal any information about the underlying computation.

Next, consider the classical information received after measuring each ancilla qubit. This bit indicates whether or not the phase of the qubit (unknown to the server) was added or subtracted from the phase of the computational qubit. However, the phase of the ancillary qubit was randomized in both sign and magnitude, so this bit gives zero information.

Finally, given the classical data received by the server, let us consider the quantum state at any given step of the computation. Trivially, at the beginning of the computation, each qubit has a randomized Pauli operator applied ($X^i Z^j$ for random $i, j \in 0, 1$). In this case, it is easy to show that for any input state, the server would probabilistically expect the maximally mixed state.

Now, consider the effect of the X or Z rotations described above. Regardless of the ancilla measurement or the rotation amount designated by the client, the server has equal probability of all rotations. For Z, the server has equal probability of applying $Z_{n\pi/4}$ for $0 \leq n < 8$. Similarly for X, the server has equal probability of applying $X_{n\pi/4}$ for $0 \leq n < 8$.

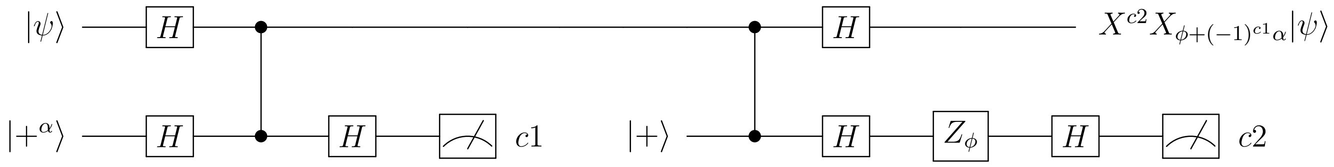
Given an arbitrary input state ρ, the resulting state is either a classical distribution of $1/2(Z\rho Z + \rho)$ and $1/2(X\rho X + \rho)$ respectively. This is the equivalent of an unknown Pauli operator applied to the initial state.

Since our initial state had unknown Pauli operators applied, and we proved that this corresponded to the maximally mixed state, additional random Pauli operators will not give the server any more knowledge. Indeed, it actually would make it more difficult for the server. Even if the server somehow exactly knew the input qubit's state ρ, after a single X and Z rotation, the server has lost that knowledge, and the state returns to the maximally mixed state to the best of the server's knowledge.

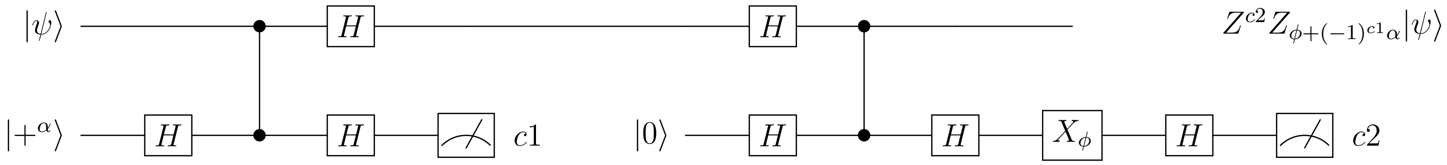
Note that although the application of the unknown rotation looks to the server like a randomized Pauli operation, the client controls the rotation. Therefore, the originally chosen Pauli operators for each qubit would not change if each rotation were executed exactly. This is why we have added the additional $r\pi$ rotation on each operation (random $r \in 0, 1$). For an input state $Z^a|\psi\rangle$ to a Z operation, this corresponds to changing the resulting state to $Z^{(a+r)}Z_\theta|\psi\rangle$. The result for X is similar. This randomizes the Pauli operations at each step, rather than relying on the same Pauli operators for each qubit during the entire computation. □

5.1. **Completely Ancilla-Driven Computation.** The above model has also been explored by Sueki, Koshiba, and Morimae.[SKM12] They use the model of Ancilla-Driven Quantum Computation to build a Blind Quantum Computation protocol.[AOKBA10] This uses the phase-kickback mechanism described above with the main difference being in the controlled rotation that occurs afterward.

The construction is similar to the above, except the X and Z rotation circuits are given below.



Similarly, the Z rotation is:



Note that each of these circuits contains two ancillary qubits that are measured. The first qubit kicks a random amount of phase onto the computational qubit in the exact same way as in the previous section. The second qubit executes a controlled X or Z rotation on the qubit, with the rotation angle ϕ known to the server. The main difference is in that in this protocol, the controlled rotation is performed with an ancillary qubit instead of directly.

Other than this small deviation, the protocol is identical and therefore has the same blindness and universality properties.

6. Conclusion

We have investigated several forms of blind quantum computation. First, ABE08 gave a high-level discussion of how random local Pauli operators could hide a computation from the server.

This observation was turned into a complete protocol by BFK09 using Measurement-Based Quantum Computation. However, we have now discussed what properties of MBQC allow for blind computing to be achieved. We discovered that these properties are separable from the MBQC construction.

Specifically, we identified the following properties:

(1) Hide computational structure with organized "universal cells" such as in the "Brickwork State".
(2) Break all operations down into X and Z rotations. These rotations have known commutation relations with the Pauli X and Z operators. Use these commutation relations to correctly apply the X and Z rotations to qubits that have randomized Pauli operators applied to them.
(3) Break each X and Z rotation into two steps: One randomized that only the client knows (but may not necessarily be able to choose), and one that adds to the random value to give the desired overal rotation amount. Only the latter angle is known to the server.

We observed that none of these properties are unique to MBQC. There is an equivalent "Brickwork State" for circuit computation. In addition, while X and Z rotations are the natural fundamental operations for MBQC, it is easy to implement rotations in other models as well. Finally, randomizing qubit phase addition can be done with a phase-kickback circuit instead of quantum teleportation.

This led us to a new model of blind quantum computation. This model turns out to be similar to blind "Ancilla-driven" quantum computation just recently explored in SKM12.

The three criteria above unify these two seemingly different mechanisms for blind quantum computation and provide direction for the development of new blind protocols using other quantum phenomena.

References

[BB84] Brassard, G., Bennett, C.H. Public key distribution and coin tossing. Proceedings of IEEE International Conference on Computers, Systems and Signal Processing (1984)

[ABE08] Aharonov, D., Ben-Or, M., Eban, E. Interactive Proofs for Quantum Computations arXiv preprint arXiv:0810.5375 (2008).

[GC99] Gottesman, D., Chuang, I.L. Quantum teleportation is a universal computational primitive. Nature 402 (1999)

[RB01] Raussendorf, R., Briegel, H. J. A one-way quantum computer Phys. Rev. Lett. 86, 5188 (2001), eprint arXiv:quant-ph/0010033.

[BFK09] Broadbent, A., Fitzsimons, J.,Kashefi, E. Universal blind quantum computation. Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS), (2009).

[CLN05] Childs, A.M., Leung, D.W., Nielsen, M.A. Unified derivations of measurement-based schemes for quantum computation. Physical Review A 71 (2005), quant-ph/0404132

[SKM12] Sueki, T., Koshiba, T., Morimae, T. Ancilla-Driven Universal Blind Quantum Computation eprint arXiv:1210.7450

[AOKBA10] Anders, J., Oi, D. K. L., Kashefi, E., Browne, D. E., Andersson, E. Ancilla-driven quantum computation with twisted graph states Phys. Rev. A 82, 020301(R) (2010).