A Scheme for universal blind quantum computation

Mear M. R. Koochakie

Department of Physics, Sharif University of Technology, Tehran 14588, Iran

 Blind quantum computation (BQC) is a recent proposition in which a computation is performed on a server by a client in a manner that the server is kept blind about the input, the algorithm and the output of the computation. This model possesses a natural security, which makes it suitable for applications where privacy matters. In this paper we propose a framework for BQC which can be implemented by circuit models as well as measurement-based models of quantum computation. It is based on gate teleportation, a process in which by storing one-qubit gates in entangled pairs one can apply it later via teleportation. We elaborate on examples of such universal BQC (UBQC) protocols. Specially, we elaborate on UBQC for correlation-space measurement-based quantum computation.

PACS numbers: 03.67.Ac, 03.67.Hk, 03.67.Dd

I. INTRODUCTION

From Shor’s factoring algorithm [1] to the BB84 quantum key distribution protocol [2], quantum physics has shown to surpass classical computational algorithms and communication protocols. A recent application of quantum physics in the privacy of delegated computations has emerged in the blind quantum computation protocols [3, 4]. A blind computation is a protocol which allows a client to compute on a remote server with full privacy about the computation and the results. Hence, the client delegates a computation to the server without letting it knows about the input, the computation, and the output. Apparently, it is not possible to perform universal classical computation on a server blindly using only a classical communication channel and classical computers [5], but with a quantum channel and a quantum computer in the server’s lab the story differs. Blind quantum computation first appeared in Ref. [3]. A milestone in this field is the work of Broadbent, Fitzsimons, and Kashefi [4], which has a proof-of-principle experimental implementation [6], that needs low requirement for the client’s quantum abilities. Next, several UBQC proposals were suggested [7–11] which mainly differ with each other in their preferred way of implementation, and/or the efficiency of the protocol (the amount of required quantum communication to hide a computation).

It seems that many of the proposed schemes have been tailored for some specific model. In other words, each of them can be implemented more easily on its own quantum computation model. For example, the model in Ref. [4] is based on one-way measurement-based quantum computation [12]; or the model in Ref. [10] is more suited for programmable quantum gate arrays [13].

In this paper we propose a fairly general framework for blind quantum computation, which can be applied more naturally and systematically to the computational models such as (general) measurement-based or circuit-based quantum computation, without need to simulate the computation in other paradigms. The requirements for the server is that it should be able to perform a Bell measurement between one of his (internal) qubits and a (an external) qubit that is sent to him via the quantum channel, and that the server is able to swap the internal and external qubits. On the other side, the client should be able to prepare a family of entangled two-qubit states which are locally equivalent to a Bell pair.

In this paper we also investigate the security of blind QC protocols in view of accessible information to the server from the quantum channel shared between the server and the client. We find that the existence of a procedure to catch a privacy-invasive server is very crucial for the blindness of a BQC protocol. Notably, we show a possible attack to the protocol introduced in Ref. [4] if no authentication procedure has been performed.

II. PRIVACY OF BQC PROTOCOLS

Our formal definition of blindness of a BQC protocol is similar to the one in Ref. [4]. Let \( \mathcal{Q} \) be a subset of \( \text{BQP} \). Let \( P \) be a quantum computation protocol in which two parties are involved, which should collaborate to solve a problem, one as “client” and the other as “server”, sharing a classical and a quantum channel with each other. We assume the client has less quantum power than the server. Specifically, we assume the client alone can not solve all instances of the problems in \( \mathcal{Q} \) efficiently but the server alone has quantum devices that can solve any problem in \( \mathcal{Q} \) efficiently. We say protocol \( P \) is a blind quantum computation protocol on \( \mathcal{Q} \) if the client can solve any problem in \( \mathcal{Q} \) without revealing the problem, the algorithm, and the answer to the server. In other words, the server at most can only randomly guess what

\* koochakie@gmail.com

1 In the original paper it has been called “Secure assisted quantum computation”.

2 Bounded error quantum polynomial time computational complexity class
is going on independent of the actual computation. A universal BQC is a BQC on BQP.

The description of any computation is transferred to the server using both classical and quantum channels. Let the client use set \( \{|\psi_i\} \) as the quantum alphabet for the communication with the server. The states in the alphabet should not be mutually orthogonal, otherwise the quantum channel basically acts like a classical channel. The security concerns are how much information the server can obtain from the quantum channel, and what are the methods to prevent the server from such intrusion.

**Lemma 1.** Let \( \{|\psi_i\} \) be a finite set of \( N > 1 \) distinct states on a finite-dimensional space. Let choose randomly from this set according to the distribution \( \eta = \{\eta_i | 0 < \eta_i, \sum_{i=1}^{N} \eta_i = 1\} \). There is always a POVM that can discriminate these states better than a random guess, namely, the probability of successfully detecting any state \(|\psi_i\rangle\) form others in the random sequence is strictly greater than \( \eta_i, \forall i \).

**Proof.** We introduce \( N \) positive operators \( \Pi_i \) from which the probability of triggering the \( j \)-th output while measuring the actual state \( \psi_i \) is given by \( p(j|i) := \langle \psi_i | \Pi_j | \psi_i \rangle \). Consider \( \Pi_j := \alpha_{\eta_j} |\psi_j\rangle \langle \psi_j| \) for \( j = 1 \ldots N \) and \( \Pi_j \) as the elements of the POVM, where \( \Pi_j \) represents the indecisive result. Thus \( p(i|i) = \alpha \eta_i \) is the probability of detecting state \( \psi_i \) successfully. Let us find the largest \( \alpha \), which maximizes these success probabilities. POVM operators should satisfy

\[
\sum_{j=1}^{N} \Pi_j + \Pi_\gamma = 1, \quad \text{and} \quad \Pi_\gamma \geq 0, \quad (1)
\]

hence

\[
1 - \alpha \sum_{j=1}^{N} \eta_j |\psi_j\rangle \langle \psi_j| \geq 0. \quad (2)
\]

Hence, \( 1 - \alpha \lambda_{\text{max}} \geq 0 \), where \( \lambda_{\text{max}} \) is the largest eigenvalue of the density matrix \( \sum_{j=1}^{N} \eta_j |\psi_j\rangle \langle \psi_j| \). Therefore, \( \alpha \) can be chosen as large as \( \alpha_{\text{m}} := \frac{1}{\lambda_{\text{max}}} > 1 \) which comes from \( \lambda_{\text{max}} < 1 \). This implies that

\[
p(i|i) = \alpha_{\text{m}} \eta_i > \eta_i. \quad (3)
\]

This lemma can be used to obtain information about the computation by a privacy-invasive server. As a case study, consider the scheme introduced in Ref. [4] with no trap/authentication part. In the protocol the client initially sends a sequence of quantum states chosen uniformly randomly from the set \( \{|\psi_j\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i j/N}|1\rangle) | 0 \leq j < N \} \) (here \( N = 8 \)) to the server. Due to the symmetry of these states, there exists an optimal minimum-error quantum state discrimination scheme with the following POVM defined as [14]

\[
\Pi_j := \langle \mu_j | \mu_j \rangle, \quad (4)
\]

\[
|\mu_j\rangle := B^{-1/2} |\psi_j\rangle, \quad (5)
\]

\[
B := \sum_{j=1}^{N} |\psi_j\rangle \langle \psi_j|. \quad (6)
\]

Since here \( B = \frac{N}{2} \mathbb{1} \), so the probability of correctly detecting the \( i \)-th state in the sequence is

\[
p(i|i) = \langle \psi_i | \mu_i \rangle^2 = \frac{2}{N}, \quad (7)
\]

which is twice the probability of a random guess (1/N).

The amount of information that the server can obtain using the above attack from the sequence can be quantified by the mutual information between these two random variables: \( X \) as the actual index of the state, and \( Y \) as the outcome of the POVM. We have

\[
p(Y = j | X = i) = \frac{2}{N} \cos^2 \frac{i - j}{N} \pi, \quad (8)
\]

\[
p(X = i) = \frac{1}{N}, \quad (9)
\]

from which the mutual information becomes

\[
I(X,Y) = 1 + \frac{2}{N^2} \sum_{i,j} \cos^2 \frac{i - j}{N} \pi \log_2 \cos^2 \frac{i - j}{N} \pi. \quad (10)
\]

For the case of Ref. [4], where \( N = 8 \), roughly 0.4496 bits out of each 3 bits can be gained by a privacy-invasive server. The mutual information (10) is non-vanishing, which means the server always can obtain information from the sequence by the measurement. Providing \( N > 2 \), it is also less that one, which agrees with the Holevo bound [15]. Also \( \lim_{N \to \infty} I(X,Y) = 1/\ln 2 - 1 \approx 0.4427 \).

A possible solution for such malicious behavior of the server can be, if it is feasible in the protocol, to frequently check whether the server is trying to identify the incoming states, and to discontinue the computation on the first such encounter.

**Observation 1.** Consider a client uses an (unverified) BQC protocol on a server. As a part of the protocol, the client should choose randomly from a set \( \text{(with given size)} \) of finite-dimensional quantum states and send them to the server. (By “given” we mean finite and independent of the problem size.) This protocol can not be blind if no way is determined in the protocol to check frequently whether the server is trying to discriminate the incoming states from each other.

Placing traps in the computation is probably an easy way to detect a privacy-invasive or non-cooperative server. The client instructs the server to perform random non-entangling operations on a few qubits using the
BQC protocol; these are the trap qubits. Then she randomly checks the states of the traps. Due to the fact that there is no way to discriminate non-orthogonal states perfectly, when the server tries to find any clue about the computation there is always a nonzero chance he make a mistake about the operation to be performed on the trap qubits. Therefore, a periodic direct or indirect checking of the traps ultimately would catch the cheating server.

III. OUR SCHEME

We use an idea based on Refs. [13, 16, 17], where it has been shown how to store a quantum gate in an entangled state and apply it later, a process which is called “gate teleportation”. Consider the standard quantum teleportation protocol. Now instead of using a Bell pair $|\Phi^+\rangle := \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ let us use $1 \otimes V|\Phi^+\rangle$. Thus the state of the teleported qubit now becomes

$$V\sigma|\psi\rangle,$$  \hspace{1cm} (11)

where $|\psi\rangle$ was the state of the target qubit before teleportation, and $\sigma \in \Sigma$, where $\Sigma := \{1, X, Z, ZX\}$ (Pauli matrices plus identity), is the by-product of the Bell measurement of the teleportation protocol. One can defer application of any one-qubit unitary $V$ (up to some Pauli matrix) by storing the gate into the entangled state

$$|\Phi_V\rangle := 1 \otimes V|\Phi^+\rangle,$$  \hspace{1cm} (12)

and performing the Bell-measurement part of the teleportation protocol later. Consider $V = \{V_i\}$ as the set of unitaries from which the client makes the states $|\Phi_V\rangle$. For the protocol to be blind, at least two of the states $\{|\Phi_V\rangle\}$ should be non-orthogonal. In other words, in the mutual inner products of these states,

$$\langle \Phi_V|\Phi_{V'}\rangle = \langle \Phi^+|1 \otimes V_i^\dagger V_j|\Phi^+\rangle$$

$$= \frac{1}{2} \text{tr} V_i^\dagger V_j,$$  \hspace{1cm} (13)

there should be non-vanishing results.

A. UBQC for circuit models

Assume the server has a circuit model quantum computer. Our idea for UBQC on such hardware is to use gate teleportation for implementing one-qubit gates. Hence, the client should be able to prepare $|\Phi_{V_i}\rangle$ where the set $V = \{V_i\}$ is sufficient to construct universal one-qubit gates. Therefore, if the client wants gate $V_i$ to be performed on one of the server’s quantum registers, she prepares $|\Phi_{V_i}\rangle$ and sends it to the server and ask him to teleport the register qubit through $|\Phi_{V_i}\rangle$. In the general case the client gets $V_i\sigma$, where $\sigma$ is a Pauli operator or the identity. Now, we present a stochastic process to circumvent this problem that $\sigma$ could be $\neq 1$. The client sends a series of $|\Phi_{V_i}\rangle$ adaptively depending on the previous measurements, through which she can obtain the desired result stochastically via teleportation. A desired result is a result of a Bell-state measurement when $\sigma = 1$. At the first step, the client applies $V^1 = V_i$ using teleportation. If this action was unsuccessful, i.e., rather $V^1\sigma^1$ with $\sigma^1 \neq 1$ has been implemented, she sends $|\Phi_{V_2}\rangle$ with $V^2 = V_i\sigma^1 V_i^\dagger$. Hence continuing on this fashion, if in the first $l$ steps the desired result has not been achieved for the step $l + 1$ the client should send $|\Phi_{V_{l+1}}\rangle$, where

$$V^{l+1} = V_i\sigma^l V_i^\dagger.$$  \hspace{1cm} (14)

For a universal quantum computation we should also hide placement of two-qubit gates from the server. The idea we use here is similar to that of Ref. [4]. Consider the following two-qubit operator:

$$R(U) := W^\dagger 1 \otimes UW,$$  \hspace{1cm} (15)

where $W$ is the two-qubit gate that the server can perform as one of the building blocks for universal quantum computation, and $U \in U(2)$. Due to the universality, $R(U)$ can also be implemented. There is always at least an $S$ such that $R(S)$ becomes an entangling gate. Hence our protocol to hide two-qubit gates is to use $R(S)$ as the two-qubit gate in the computation, and hide that the $U$ inside $R(U)$ is whether $1$ or $S$ using the method explained for one-qubit gates. To hide the placements of the $R(S)$ gates in the computation, one may need to use a structure similar to the Ref. [4] to spread the $R(U)$ gates (some redundant) all over the circuit (FIG. [1]).

It is possible to choose the set $V$ such that delegation for a computation in blind form, does not need a stochastic process. In such models, the set $V$ and the two-qubit gate $W$ have desirable properties that allow the client to neglect the Pauli by-product of any teleportation. Mathematically, these desired criteria can be expressed as fol-
lows:
\[ W\Sigma \otimes \Sigma \subseteq \Sigma \otimes \Sigma W, \quad (16a) \]
\[ v\Sigma^\dagger \subseteq \Sigma^\dagger v, \quad (16b) \]

The last relation says that for any \( V \in \mathcal{V} \) and any \( \sigma \in \Sigma \) there should exist \( V' \in \mathcal{V} \) and \( \sigma' \in \Sigma \) such that
\[ \sigma' v = v' \sigma. \quad (17) \]

Thus if the client wants to implement \( V \) in the server, while \( \sigma \) has been the overall by-product of the previous teleportations, she should get the server to perform \( V' \) by teleportation. To design a UBQC protocol using the above ideas, one should determine \( W \), and \( \mathcal{V} \) satisfying these criteria.

B. Desirable commutation relation

In this section we present an example of a UBQC model satisfying the relations in Eq. (16). First, we choose the two-qubit gate of the protocol to be the CZ gate. According to Ref. [18], all one-qubit unitaries passing the CZ gate can locally be formulated as
\[ Z(\theta)\sigma^i, \quad (18) \]
where \( \sigma^i \in \Sigma \), and \( Z(\theta) := \exp(i\theta Z/2) \) for any real \( \theta \). Hence automatically criterion of Eq. (16a) has been satisfied. Let us assume
\[ V_i = V_0 Z(\theta_i). \quad (19) \]
Here \( V_0 \) is a constant gate. It is easy to see that relation of Eq. (16b) is satisfied if \( V_0 \) is a member of one-qubit Clifford group:
\[ V_0 \in \mathcal{C}_1 := \langle H, \sqrt{Z} \rangle, \quad (20) \]
where \( H \) is the Hadamard gate. As an example satisfying criteria (16), one can choose \( \mathcal{V} = \{ H, HZ(\pm \pi/4) \} \) which is a universal set of one-qubit gates. It is straightforward to see that there are non-orthogonal states in \( \{ |\Phi_v \rangle \} \) of this example.

C. Continuous UBQC

Let the client be able to prepare \( |\Phi_{V_i} \rangle \) for any \( V_i \in \text{SU}(2) \). Also let \( \mathcal{U} = \text{SU}(2) \), which is the set of all one-qubit gates the server can perform at each step. Assume that the client wants to apply the one-qubit gate \( U_1 \) on a qubit in the server, for the first step. She chooses a random \( V_1 \) from \( \text{SU}(2) \) according to the Haar measure \( \mu \) on \( \text{SU}(2) \), and sends \( |\Phi_{V_i} \rangle \) to the server, which then be applied by teleportation. Assume \( V_1 \sigma_1 \) has been performed on the target qubit by teleportation. Then the client ask the server to perform \( R_1 = \tau_1 U_1 \sigma_1 V_1^\dagger \) on the target qubit, where \( \tau_1 \) has been chosen uniformly random from \( \Sigma \). Thus overall the operation \( R_1 V_1 \sigma_1 = \tau_1 U_1 \) is applied on the target qubit. By knowing only \( R_1 \) and \( \sigma_1 \), the server can not obtain any information about what \( U_1 \) or \( \tau_1 \) is since \( \mu(dR_1) = \mu(d\tau_1 U_1 \sigma_1 V_1^\dagger) = \mu(dV_1^\dagger) = \mu(dV_1) \), which means that \( R_1 \) is always as random as \( V_1 \) (independent of \( U_1 \) and \( \tau_1 \)). The random \( \tau_1 \) has been placed in the protocol to encrypt the state of the server’s register qubit using the quantum Vernam cipher, to prevent the server from gaining any information from the state of the register. Suppose that at the next step, the client wants to perform, e.g., the one-qubit \( U_2 \) on the same target qubit. She choses \( V_2 \) randomly and applies it using teleportation on the target qubit. Then she asks the server to apply \( R_2 = \tau_2 U_2 \sigma_2 V_2^\dagger \), where \( \sigma_2 \) is the by-product of the teleportation process, and \( \tau_2 \) has been chosen randomly from \( \Sigma \). The same idea can be used for next steps, when the client wants to apply a one-qubit gate.

To hide two-qubit gates in this example protocol, here we use the same idea introduced in Ref. [4]. In Eq. (15), we use a Control-Z (CZ) gate as \( W \), and \( S = X(-\pi/2) \), where \( X(\theta) := \exp(i\theta X/2) \). Hence \( R(S) \) will be locally equivalent to CNOT gate.

IV. UBQC FOR MQC MODELS

In measurement-based quantum computation (MQC), rather than performing unitary gates, quantum computation is implemented by applying a series of adaptive single-site measurements on a highly entangled state [12, 19]. Let \( \mathcal{M} = \{ M_i \} \) be the set of all single-site measurements \( M_i \) which is needed to be performed in an MQC model. Our idea here is to construct \( \mathcal{V} \) whose members permute elements of \( \mathcal{M} \). Let us restrict ourselves to projective measurements, where any measurement \( M_i \) can be described by a set of orthogonal basis \( M_i \leftrightarrow \{ |m_{ik} \rangle \} \) (measurement basis). Thus the unitary
\[ V_{ij} = \sum_k |m_{jk}|^2 |m_{ik} \rangle \langle m_{ik}|, \quad (21) \]
substitutes measurement \( M_i \) by \( M_j \) if applied to the target qubit before the measurement. Hence to perform \( M_j \) blindly, the client first applies \( V_{ij} \) via stochastic teleportation process we introduced in Sec. [11] where \( j \) is random: next the client asks the server to measure the target qubit in the \( M_j \) basis.

The above process is a stochastic process which works for all MQC models, but requires the client to be able to prepare a relatively large of entangled states. Now it is important to see how one can obviate the need for a blind stochastic process in a give MQC model. In other words, what are the constrains similar to the relations in Eq. (16) for MQC models. To this end, we use the “correlation-space MQC” framework [18, 20, 21] for its
generality in the sense that this framework already can describe all known MQC models.

A. Correlation-space MQC

Here we briefly review the idea of the correlation-space MQC [15, 20, 21]. A matrix product state (MPS) is a state, that can be written in the standard basis as follows:

\[ |\text{MPS}\rangle = \sum_{i_1, \ldots, i_N} (L|A_N(i_N) \cdots A_2(i_2)A_1(i_1)|R) |i_N, \ldots, i_2, i_1\rangle, \]

(22)

where \( N \) is the number of sites, \( 0 \leq i_j < D \), and \( A_j(i) \) is a \( d \times d \) matrix attributed to state \( |i\rangle \) of site \( j \). Here \( \langle L | \) and \( | R \rangle \) are the left and the right boundary vectors. The amplitude coefficients of the MPS are computed by a set of matrix multiplications in the so-called correlation space. If we measure the \( j \)th site and the output state is \( |\varphi\rangle \), then the MPS becomes

\[ |\varphi\rangle_j |\varphi| \langle \text{MPS} = \sum_{i_1, \ldots, i_{j-1}, i_{j+1}, \ldots, i_N} \langle L|A_N(i_N) \cdots A_j|\varphi\rangle \cdots A_1(i_1)|R \rangle |i_N, \ldots, i_{j-1}, i_{j+1}, \ldots, i_1\rangle, \]

(23)

in which \( A_j|\varphi\rangle := \sum_i \varphi_i^* A_j(i) \) assuming \( |\varphi\rangle = \sum_i \varphi_i|i\rangle \). Hence, any operator in \( \text{span}(A_j) := \{\sum_i \alpha_i A_j(i)\} \) can be realized at the correlation space by a single-site measurement. If we measure, e.g., \( l \) sequential sites after site \( m \) of the MPS, we obtain the following operator in the correlation space:

\[ U = A_{m+l}|\varphi_l\rangle \cdots A_{m+2}|\varphi_2\rangle A_{m+1}|\varphi_1\rangle. \]

(24)

Thus, by choosing appropriate \( A_j \) as the lists of matrices and suitable measurement bases, one can construct any unitary in \( U(d) \) at the correlation space. Let us select \( A_j \) and measurement bases such that \( A_j|\varphi\rangle \) for any measurement outcome at site \( j \) is a unitary operator. Randomness of measurement outcomes causes instead of a desired \( |\varphi\rangle \) as outcome one obtains \( |\varphi'\rangle \). Therefore, rather than \( U := A_j|\varphi\rangle, A_j|\varphi'\rangle \) is realized in the correlation space. In such cases we can think of \( A_j|\varphi'\rangle \) as

\[ A_j|\varphi'\rangle = EU, \]

(25)

where \( E = A_j|\varphi'\rangle U^\dagger \) is the so-called “by-product operator.” Such by-products are needed to be circumvented. We will present conditions to manage by-products later [in Eq. (26)].

Assume we are working on a translationally invariant correlation-space MQC model, which means \( A_j = A \). Also assume the model is deterministic, which means that the number of steps needed to perform any gate is predetermined. Let \( \mathcal{M} = \{M_i\} \) be the set of all single-site measurement \( M_i \) of the MQC model to implement one-qubit gates. Let \( \mathcal{U} \) be the set of all single-qubit unitaries \( U_i \) that can be performed at each step of the MQC model, which can be written as \( A|\varphi_i\rangle \), where \( |\varphi_i\rangle \) is the target measurement result of \( M_i \). Let \( \mathcal{E} \) denotes the set of all manageable by-products of the MQC model. By this we mean that for any \( U_j \in \mathcal{U} \) and any \( E \in \mathcal{E} \), there should exist an \( M_t \) such that for any \( |\varphi_j\rangle \) in \( M_t \),

\[ A|\varphi_i\rangle E U_j^\dagger \in \mathcal{E}, \]

and also \( W \mathcal{E} \otimes \mathcal{E} \subseteq \mathcal{E} \otimes \mathcal{E} \).

B. Example of a universal blind MQC model

One way to satisfy the criterion (27) is to have \( \sigma V^\dagger |\varphi_j\rangle \in M_i \) for some \( M_i \). In this section we introduce such \( V \) for the one-way MQC model [12]. In the one-way model the measurement basis is parametrized as

\[ M_\theta = \{|\theta\rangle := (|0\rangle \pm e^{i\theta}|1\rangle)/\sqrt{2}\}. \]

(28)

Now consider

\[ V_\gamma := Z(-\gamma) = \exp(-i\gamma Z/2). \]

(29)

By applying this operator for a random \( \gamma \) through teleportation before a measurement \( M_\beta \) (with \( \beta = \alpha - \gamma \)), the server will have \( M_\alpha = M_{\beta+\gamma} \) on his qubit. Hence, in this way the client can hide \( \alpha \) from the server. Application of the Pauli matrices is also well behaving as

\[ M_\alpha \xrightarrow{X} M_{-\alpha}, \]

(30)

\[ M_\alpha \xrightarrow{Z} M_{\alpha+\pi}. \]

(31)

Hence by choosing \( \beta \) correctly, depending on the output of the Bell measurement, the criterion (27) will be satisfied. For a discrete example, one can choose legitimate \( \theta s \) for \( M_\theta \) from \( \{\pi, \pi/2, \pm \pi/4\} \).

V. CONCLUDING REMARKS

In this paper we have shown that using the idea of storing one-qubit gates in entangled pairs a client can perform blind quantum computation on a remote server. We
present our idea for quantum circuit model and also measurement based quantum computation. We have elaborated on three examples for the idea of this paper, two for circuit model and one for correlation-space MQC. We have also discussed the security of BQC protocols via an example. It has been shown that existence of authentication steps throughout the computation is necessary for the blindness of the protocol.

For the future, it is interesting to know how one can use multipartite entanglement for BQC and what are the benefits of it. Also how one can reduce the amount of quantum communication to gain the same level of blindness. It is also interesting to see whether it is possible to design a BQC protocol without the use of traps to catch a cheating/privacy-invasive server.

ACKNOWLEDGMENT

We would like to thank Dr. A. T. Rezakhani and Prof. V. Karimipour for useful comments.

[1] P. W. Shor, in Proceedings of 35th Annual Symposium on Foundations of Computer Science (IEEE Computer Society Press, Los Alamitos, CA, USA, 1994) pp. 124–134.
[2] C. H. Bennett and G. Brassard, in Proceedings of the IEEE International Conference on Computer, Systems, and Signal Processing, Bangalore, India (IEEE Computer Society Press, New York, NY, USA, 1984) p. 175.
[3] A. M. Childs, Quantum Info. Comput. 5, 456 (2005).
[4] A. Broadbent, J. Fitzsimons, and E. Kashefi, in Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS ‘09) (IEEE Computer Society Press, 2009) pp. 517–526.
[5] M. Abadi, J. Feigenbaum, and J. Kilian, Journal of Computer and System Sciences 39, 21 (1989).
[6] S. Barz, E. Kashefi, A. Broadbent, J. F. Fitzsimons, A. Zeilinger, and P. Walther, Science 335, 303 (2012).
[7] V. Dunjko, E. Kashefi, and A. Leverrier, Phys. Rev. Lett. 108, 200502 (2012).
[8] T. Morimae, Phys. Rev. Lett. 109, 230502 (2012).
[9] T. Sueki, T. Koshiha, and T. Morimae, Phys. Rev. A 87, 060301 (2013).
[10] V. Giovannetti, L. Maccone, T. Morinmae, and T. G. Rudolph, Phys. Rev. Lett. 111, 230501 (2013).
[11] A. Mantri, C. A. Pérez-Delgado, and J. F. Fitzsimons, Phys. Rev. Lett. 111, 230502 (2013).
[12] R. Raussendorf and H. J. Briegel, Phys. Rev. Lett. 86, 5188 (2001).
[13] M. A. Nielsen and I. L. Chuang, Phys. Rev. Lett. 79, 321 (1997).
[14] M. Ban, K. Kurokawa, R. Momose, and O. Hirota, Int. J. Theor. Phys. 36, 1269 (1997).
[15] A. S. Holevo, Probl. Inf. Transm. 9, 177 (1973).
[16] D. Gottesman and I. L. Chuang, Nature 402, 390 (1999).
[17] G. Vidal, L. Masanes, and J. I. Cirac, Phys. Rev. Lett. 88, 047905 (2002).
[18] M. M. R. Koochakie, Phys. Rev. A 89, 012322 (2014).
[19] H. J. Briegel, D. E. Browne, W. Dur, R. Raussendorf, and M. Van den Nest, Nature Physics 5, 19 (2009).
[20] D. Gross and J. Eisert, Phys. Rev. Lett. 98, 220503 (2007).
[21] D. Gross, J. Eisert, N. Schuch, and D. Perez-Garcia, Phys. Rev. A 76, 052315 (2007).