SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

June 2021, Vol. 64 162302:1–162302:14
https://doi.org/10.1007/s11432-020-2977-x

Multi-party blind quantum computation protocol with mutual authentication in network

Rui-Ting SHAN, Xiubo CHEN* & Kai-Guo YUAN

Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876*, China*

Received 14 May 2020/Revised 15 June 2020/Accepted 1 July 2020/Published online 2 April 2021

Abstract Blind quantum computation (BQC) can ensure a client with limited quantum capability safely delegates computing tasks to a remote quantum server. In order to resist attacks from ignoring identity authentication in BQC protocols, it is necessary to guarantee the legality of both clients and servers in a multi-party BQC network. So we propose a multi-party BQC protocol involving three phases to distribute shared keys and authenticate identities. Firstly, by using the advantages of measurement device independent quantum key distribution (MDI-QKD), the registered client and the assigned server could share the initial key safely in registration phase. Secondly, with the help of semi-honest certificate authority (CA), mutual identity authentication phase realizes the two-way authentication from both sides through the shared key simultaneously. Thirdly, in the blind quantum computing phase, a registered client can complete his computing task by just measuring the qubits from the assigned server rather than preparing the qubits. Moreover, combined with first-in-first-out (FIFO) principle, clients' authentication and blind quantum computing can be processed in parallel. The protocol can also be applied in other multi-party BQC protocols with the universality of resource states. Compared with other BQC protocols, the reliability of the protocol with identity authentication is guaranteed, and the efficiency will be significantly reflected in real experiments.

Keywords blind quantum computation, mutual identity authentication, load balancer, semi-trust CA, quantum network

Citation Shan R-T, Chen X B, Yuan K-G. Multi-party blind quantum computation protocol with mutual authentication in network. Sci China Inf Sci, 2021, 64(6): 162302, https://doi.org/10.1007/s11432-020-2977-x

1 Introduction

With the help of quantum computing theory, the parallel computing speed and storage capacity of quantum computer are far faster than the classical Turing machine model. It can be predicted that quantum computer is likely to be used to solve some problems that the classical computer cannot effectively solve. However, owing to the expensive cost and maintenance difficulties of quantum computer, the first generation of quantum computer is likely to be used in a 'cloud' operation mode. This means that users with limited quantum capabilities or no quantum capabilities will delegate computing tasks to servers on the cloud. If a client wants to delegate the computing tasks to a remote server confidentially, it must guarantee the privacy of the client's input, output and algorithm. In the face of such a demand, the concept of blind quantum computation (BQC) came into being [1]. In 2005, using the circuit-based quantum computing model, Childs proposed the first blind quantum computation protocol [2], which requires the client to have a large quantum memory and the ability to perform Pauli operations. In addition, the client also needs the ability to access quantum channels. It can be seen that although the first blind quantum computation protocol guarantees the blindness and privacy of the algorithm, the protocol still requires the client to own quantum capabilities, which in fact does not meet the requirements of BQC. Owing to the good prospect of blind quantum computation, it has been widely developed and become the research goal of many researchers to design a universal blind quantum computation protocol with classical clients [3–8]. For the purpose of enabling classical clients to successfully complete the delegated

*Corresponding author (email: flyover100@163.com) quantum computing, the ultimate goal of designing a BQC protocol is to reduce the required quantum capability of the client. The optimal blind quantum computation protocol can ensure that the client is completely classical [9], that is, the client receives or sends only classical information. In order to achieve this goal, methods such as introducing a trusted third party [10] or more servers [4] have emerged. However, in order to ensure identity authentication, data integrity, fault tolerance and other goals, in most single-server BQC protocols [11–15], it has been proved that the client cannot be completely classical without trust third party [10], which means that the client needs at least the quantum memory or the ability to access the quantum channel.

As an important kind of quantum secure multi-party computation, BQC addresses the question of guaranteeing the security of quantum delegated computation between the classical client and the quantum server. Apart from BQC, there are also other hot topics of interest in quantum secure multi-party computation area, such as quantum private comparison [16–18] and quantum private query [19–21]. In the former, two parties that distrust each other can compare their secrets to see if they are equal. The third party should be introduced to assist the comparison process. In the latter, the aim is to ensure the security of the multi-user's queries in the database, which can be seen 1-out-of-N oblivious transfer protocol. In a word, these researches form a substrate for exploring a variety of applications in quantum secure multi-party computation. In 2009, Broadbent, Fitzsimons and Kashefi (BFK) [3] first proposed a universal blind quantum computation protocol via measurement-based quantum computation (MBQC) model. The client only needs the ability to prepare the single-qubit states and sends them to the server. The server should generate brickwork states, and measure the qubits in certain angles which are determined by the instructions of the client, so that the server can help the client successfully complete any computing task. Then, in 2012, Barz et al. [22] conducted experiments on the single-server BQC protocol. In order to ease Alice's burden in BFK protocol, Dunjko et al. [23] proposed a blind quantum computation protocol based on the coherent states in 2012. In this protocol, Alice only needs to have a more classical device to prepare coherent states instead of single-qubit states to complete blind quantum computation. In 2013, Morimae et al. [24] proposed a protocol that Alice can successfully realize blind quantum computation only by single-qubit measurements. Based on no-signaling principle, the device-independent security of this protocol is more fundamental than the security of quantum mechanics. In order to solve the problem that two servers cannot communicate in BFK protocol, in 2014, Li et al. [4] proposed the almost classical client's triple-server blind quantum computation protocol on the basis of entanglement swapping. In this protocol, the client only needs to access quantum channel, and the server can communicate with each other. Similarly, in 2016, Kong et al. [25] proposed multiple server BQC protocol by using entanglement swapping. In this protocol, even if the client loses connection with one or more servers, the client can flexibly delegate its computing tasks to the servers available in the network. Unfortunately, even in the triple-server BQC protocol [4], the clients are not completely classical. Hence, we should use as few quantum servers as possible in BQC protocol [10], meanwhile the quantum capability required by the client should be reduced.

Quantum identity authentication is also an important aspect to guarantee quantum authentication and a desirable property for BQC protocols. Especially in the network environment of multiple clients and servers, the identity authentication of each participant is particularly important. Li et al. [26] first introduced identity authentication to the field of blind quantum computation, and proposed single-server BQC protocols and double-server BQC protocols that can resist man-in-the-middle attack and denial-of-service attack. However, in [26], the third party must be trusted, and the client must have the quantum ability to prepare the rotated single-qubit states and BB84 states. In fact, it cannot meet the requirements of clients' quantum capability in BQC. Hence, in the proposed protocol, the introduction of semi-trust CA and load balancers is more applicable and practical. As for the secure distribution of the identity shared key, in 2012, Lo et al. [27] proposed measurement device independent quantum key distribution (MDI-QKD), which solved the security vulnerability caused by the imperfection of measurement device in QKD protocol. In 2016, MDI-QKD was realized experimentally [9]. By combining with the decoy technology, MDI-QKD shows good robustness, which not only effectively extends the distance of key distribution, but also significantly improves the key generation rate. In theory, MDI-QKD can also be directly extended to multi-party network, so MDI-QKD has an attractive application prospect in multi-party network. In the proposed protocol, MDI-QKD is utilized in registration phase so that each legal client can share an initial key with the corresponding server which is allocated by load balancers.

In blind quantum computation, there is a trade-off between multi-server approaches and any quantum capability required by the client. The introduction of multi-server can easily reduce the quantum capabilities required by the client. Therefore, the most ideal setting is the BQC protocol composed of a classical client and a single server. In this paper, a mutual identity authentication single-server BQC protocol is proposed. Considering the development of quantum computer in the future, there will be more and more user nodes and server nodes in the network. So the proposed single server BQC protocol with identity mutual authentication is also considered in a certain scale of network. In addition, for a limited number of servers in the network, how to assign more clients than servers to the server is also a problem that needs to be considered in the identity authentication protocol in the network environment.

The remainder of the paper is as follows. In Section 2, we will briefly review two common single-server blind quantum computation protocols. In Section 3, using the shared key generated in MDI-QKD, a multi-party single-server BQC protocol with mutual identity authentication is proposed. Three phases of the proposed BQC protocol are explained in network. Section 4 analyzes correctness, blindness and security of the proposed BQC protocol. At last, Section 5 will discuss and summarize the proposed protocol.

2 Review of common single-server BQC protocols

In this section, we will briefly review two common single-server BQC protocols and compare them. One is that Alice is required to prepare rotated single-qubit states [3], and the other is that only Alice is required to do single-qubit measurements [24] to realize the universal blind quantum computation protocol. Unlike Protocol 1 where Alice needs to prepare states, the role of Alice in Protocol 2 has changed from preparing quantum states to measuring single-qubit states, which greatly reduces the quantum capability required by Alice. In the optical experiments, the measurement devices of photons are much easier to be maintained than single-qubit state generation devices. In addition, the resource state in Protocol 2 can be any universal quantum resource state in MBQC model, so the protocol also provides a general solution to directly convert the MBQC model to blind BQC model.

2.1 Protocol 1: single-sever BQC protocol where Alice prepares states

(1-1) Alice prepares randomly rotated single-qubit states' sequence $\{|+\theta_i\rangle\} = \{\frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{\mathrm{i}\theta_i}|1\rangle)\}$, where $\theta_i \in \{\frac{k\pi}{4}|k = 0, 1, 2, \ldots, 7\}$, and transmits it to Bob via quantum channel.

(1-2) According to the sequence obtained from Alice, Bob generates an universal resource state via controlled-Z (CZ) operation $\mathrm{CZ}_{a,b} \equiv |0\rangle\langle 0|_a \otimes I_b + |1\rangle\langle 1|_a \otimes Z_b$, where (a, b) is a pair of qubits, I_b is identity operator performed on qubit b and Z_b is Pauli Z operator performed on qubit b.

(1-3) Alice computes the measurement angle δ_i, and transmits it to Bob through classical channel.

(1-4) Bob measures a qubit of resource state in $\{|\pm\delta_i\rangle\}$ basis, and returns the measurement results m_i to Alice.

(1-5) Repeat above (1-3) and (1-4) until the computation is completed. The measurement angle of each step calculated by Alice needs to be corrected according to the Bob's previous measurement result. If Bob is honest, Alice will eventually get the correct computation results she wants.

2.2 Protocol 2: single-sever BQC protocol where Alice only measures states

(1-1) Bob generates a universal quantum resource state firstly.

(1-2) Bob sends each qubit of quantum resource state to Alice once a time.

(1-3) After Alice receives the qubit successfully, she would compute measurement angle that is decided by the algorithm and previous measurement results and measure each qubit in the measurement angle.

(1-4) Repeat above (1-2) and (1-3) until Alice's computation is completed. In other words, the coding and measurement between Alice and Bob are repeated layer by layer.

In detail, Protocol 2 can be summarized as follows: through the server Bob's resource state, the client Alice can get the correct output through multiple single-qubit measurements. In [28, 29], Protocol 2 is proved to have composable security. In addition, the direction of information flow in Protocol 2 is always from the server to the client. In other words, the server transmits a fixed qubit in the graph state to the client every time, so such a protocol is trivially blind and has device independent security based on no-signaling principle. That is to say, Bob cannot get any information about Alice, even if Alice's device does not perform the operation correctly. Furthermore, based on variable quantum resource state in

MBQC model, Protocol 2 provides a more universal BQC protocol, which can be extended to more BQC protocols.

By comparing the above two single-server BQC protocols, it is easy to know that Protocol 1 includes Alice's preparation and Bob's computation, while Protocol 2 includes Alice's adaptive measurement and Bob's computation. Compared with Protocol 1, the role of Alice in Protocol 2 changes from state preparation to state measurement, and Alice does not need quantum memory, so that clients are more classical. In addition, it has been proved impossible to design a blind quantum computation protocol with only a completely classical client and a single server [10], so a trusted third party must be introduced into the design of a blind quantum computation protocol with a single server [30].

3 The proposed multi-party BQC protocol where Alice only measures states

In this section, we will introduce three phases of the proposed protocol in turn, that is, the registration phase (quantum key distribution phase), identity authentication phase and blind quantum computing phase, as shown in Figure 1. It should be noted that the registration phase is the necessary step before each legitimate client joins the network, including the generation of initial key and the distribution of specific server through load balancers. Only the legitimate client sharing the key with the specific server can carry out mutual identity authentication phase and blind quantum computing phase subsequently.

In our multi-party network, there are four different roles: client, server, load balancer and semi-honest certificate authority (CA). Load balancer is divided into the allocator of m clients Load_Balancer_A, and the allocator of n servers, Load_Balancer_B. Phases 1 and 2 refer to client, server, Load_Balancer_A, Load_Balancer_B and CA. The role of CA in the proposed protocol is different of CA in public-key infrastructure, which needs only performing Bell measurement in Phase 1 and preparing non-orthogonal states in Phase 2. Only the emergence of a third party can complete the process of authentication key distribution and authentication of each other's identity through authentication key. In Phase 3, only the interaction between the client and the server is needed. In essence, Phase 3 is the single-server BQC protocol mentioned above where Alice only measures states. Because of the assumption of quantum key distribution, an unjammable public channel is required between each load balancer and CA in Phases 1 and 2. For security reasons, it should be noted that the clients, servers and load balancers are assumed as honest ones.

3.1 Phase 1: registration phase (quantum key distribution phase)

The purpose of Phase 1 is to establish a point-to-point initial key between the client and the server based on the MDI-QKD and assign a specific server to the registered client. In order to reduce the quantum capability required by the client, the Load_Balancer_A and Load_Balancer_B are used to randomly prepare single photon sequences. Only after all legitimate clients are registered, can subsequent identity authentication and blind quantum computation be carried out. That is to say, each legitimate client in the network is assigned a specific server and obtained shared key with the corresponding server. In order to avoid the eavesdropping of Eve, the registration of each legitimate client in the network is carried out in sequence. As long as the existence of Eve is found in decoy check, it is considered that the shared key is at risk and the client needs to be registered again.

Before going further, the principle of MDI-QKD [31] is reviewed briefly. MBQC model plays an important role in the universal BQC protocol. In MBQC model, the universal quantum computing can be realized by a series of single-qubit measurements acting on the universal resource state [5, 32–35]. The common resource states are graph state [36], weighted graph state [37], hypergraph state [34] and brickwork state [3,22]. As will be discussed latter, brickwork state would be used as the resource state to achieve the proposed protocol. MDI-QKD is actually a three-party quantum key distribution process, which is composed of Alice, Bob and a third party (TP) who is responsible for Bell-basis measurement. Firstly, Alice and Bob will randomly prepare a series of single photon sequences $|\gamma\rangle_A, |\gamma\rangle_B \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ under X or Z basis. Then TP will make joint Bell measurement on the photons $|\gamma\rangle_A|\gamma\rangle_B$ received from Alice and Bob, and return the measurement result $R_{AB} \in \{00, 01, 10, 11\}$ to Alice and Bob across the classical channel. If $R_{AB} = 00$, then the Bell measurement result is BMR $= |\phi^+\rangle_{AB}$. If $R_{AB} = 01$, then the Bell measurement result is BMR $= |\phi^-\rangle_{AB}$. If $R_{AB} = 10$, then the Bell measurement result is BMR $= |\psi^+\rangle_{AB}$. If $R_{AB} = 11$, then the measurement result is BMR $= |\psi^-\rangle_{AB}$. Next, Alice and Bob will publish the basis used to prepare each photon respectively, and keep the bits under the measurement

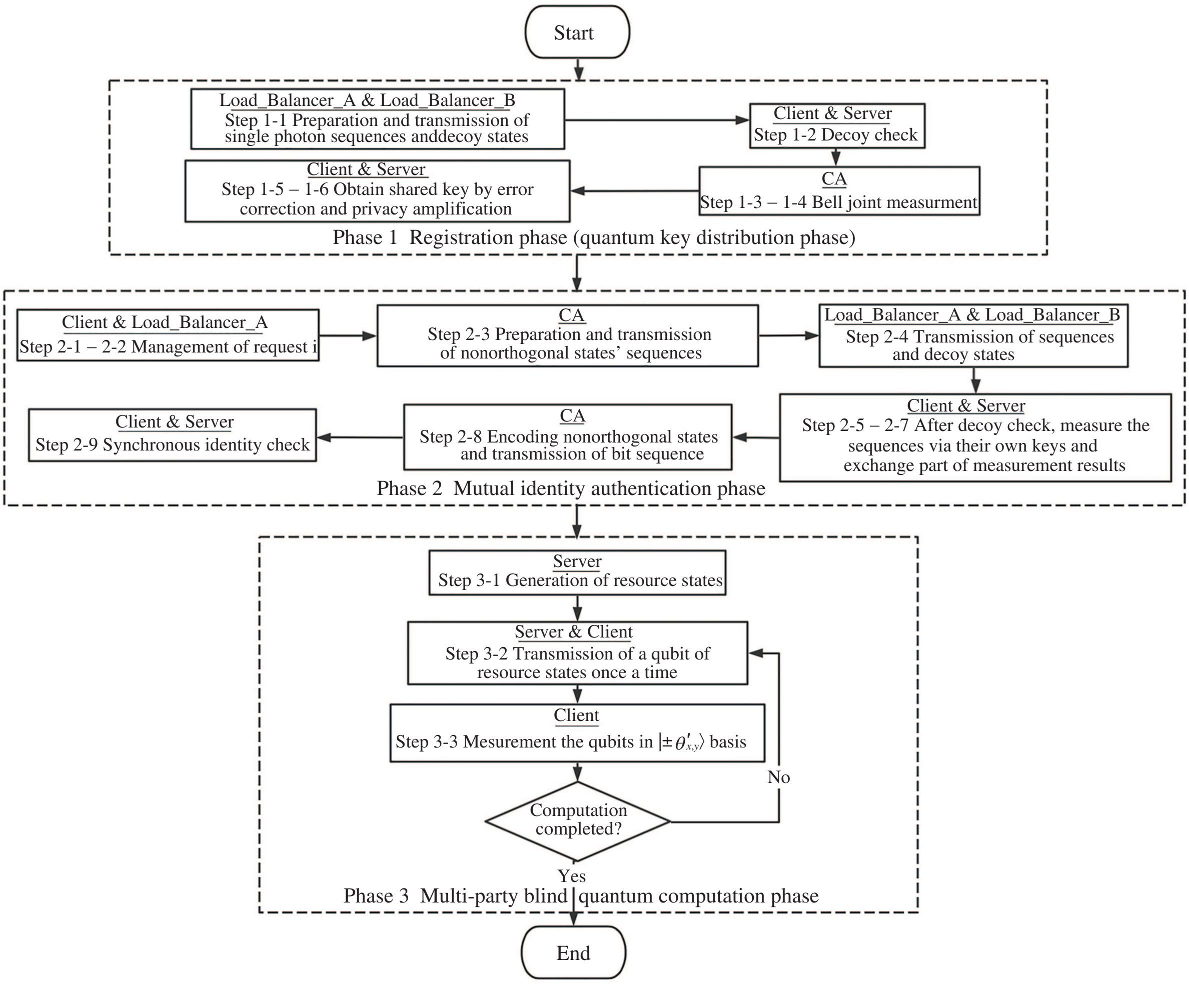
Start
Load_Balancer_A & Load_Balancer_B
Step 1-1 Preparation and transmission of single photon sequences anddecoy states
Client & Server
Step 1-2 Decoy check
CA
Step 1-3 – 1-4 Bell joint measurment
Client & Server
Step 1-5 – 1-6 Obtain shared key by error correction and privacy amplification
Phase 1 Registration phase (quantum key distribution phase)
Client & Load_Balancer_A
Step 2-1 – 2-2 Management of request i
CA
Step 2-3 Preparation and transmission of nonorthogonal states' sequences
Load_Balancer_A & Load_Balancer_B
Step 2-4 Transmission of sequences and decoy states
Client & Server
Step 2-5 – 2-7 After decoy check, measure the sequences via their own keys and exchange part of measurement results
CA
Step 2-8 Encoding nonorthogonal states and transmission of bit sequence
Client & Server
Step 2-9 Synchronous identity check
Phase 2 Mutual identity authentication phase
Server
Step 3-1 Generation of resource states
Server & Client
Step 3-2 Transmission of a qubit of resource states once a time
Client
Step 3-3 Mesurement the qubits in $|\pm\theta'_{x,y}\rangle$ basis
Computation completed?
No
Yes
Phase 3 Multi-party blind quantum computation phase
End

Figure 1 Flow chart of three phases in the proposed protocol.

Table 1 Distribution probabilities of CA's Bell-basis measurement results (BMR) and single photons' states $|\gamma\rangle_A|\gamma\rangle_B$

	$\vert\phi^+\rangle_{AB}$	$\vert\phi^-\rangle_{AB}$	$\vert\psi^+\rangle_{AB}$	$\vert\psi^-\rangle_{AB}$
$\vert 0\rangle_A\vert 0\rangle_B$	1/2	1/2	0	0
$\vert 0\rangle_A\vert 1\rangle_B$	0	0	1/2	1/2
$\vert 1\rangle_A\vert 0\rangle_B$	0	0	1/2	1/2
$\vert 1\rangle_A\vert 1\rangle_B$	1/2	1/2	0	0
$\vert +\rangle_A\vert +\rangle_B$	1/2	0	1/2	0
$\vert +\rangle_A\vert -\rangle_B$	0	1/2	0	1/2
$\vert -\rangle_A\vert +\rangle_B$	0	1/2	0	1/2
$\vert -\rangle_A\vert -\rangle_B$	1/2	0	1/2	0

result BMR $= |\psi^-\rangle_{AB}$ as the initial raw key, which means Alice and Bob share the same initial preparation basis in this instance. Finally, some bits of the initial raw key are selected for public comparison and detection monitoring, and the real shared key is obtained after post-processing. Any two-qubit can be expressed via four Bell bases, for example, $|0\rangle_A|1\rangle_B = \frac{1}{\sqrt{2}}(|\psi^+\rangle_{AB} + |\psi^-\rangle_{AB})$. Table 1 below lists corresponding distribution probabilities of single photons' states $|\gamma\rangle_A|\gamma\rangle_B$ and CA's Bell-basis measurement results (BMR).

As shown above, when BMR $= |\psi^-\rangle_{AB}$ Alice and Bob prepare the photons in the same X basis or Z basis. In this situation, Alice's and Bob's quantum states $|\gamma\rangle_A|\gamma\rangle_B$ can be $|0\rangle_A|1\rangle_B$, $|1\rangle_A|0\rangle_B$, $|+\rangle_A|-\rangle_B$, $|-\rangle_A|+\rangle_B$, which means they are mutually orthogonal. Based on above three-party MDI-QKD scheme, an extended multi-party MDI-QKD scheme whose network structure is tree-type is proposed as shown in Figure 2.

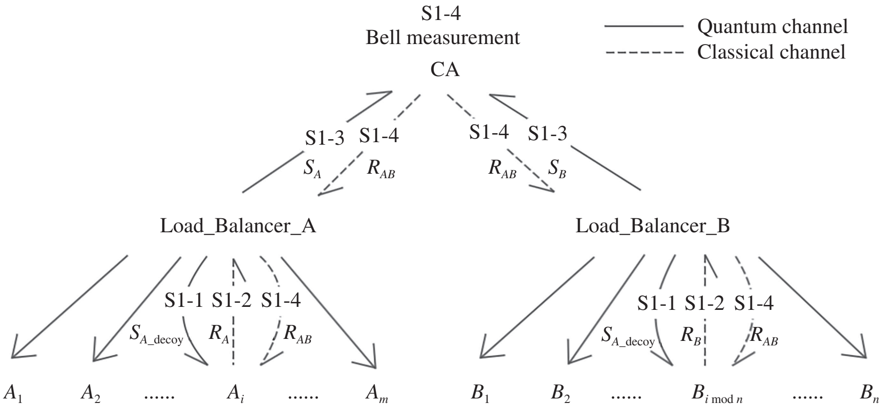

Figure 2 Information flow of each step in registration phase.

In this paper, Load_Balancer_A and Load_Balancer_B are responsible for the preparation of random single photon sequences. The bit comparison and post-processing work are completed by the client and the corresponding specific server, and the point-to-point shared key is finally shared between the client and the server. Assume there are m clients and n servers in the network, where $m \geqslant n$, transmission of information flow in each step of registration phase is shown in Figure 2.

General steps of registration phase.

S1-1. Load_Balancer_A and Load_Balancer_B prepare a series of single photon sequences randomly in X basis or Z basis, denoted by S_A and S_B. Then Load_Balancer_A and Load_Balancer_B also prepare decoy sequences with fixed length. The decoy states in $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ are inserted into S_A and S_B, and the new sequences are denoted as S_{A_decoy} and S_{B_decoy}. These two sequences are transmitted to the client A_i and the corresponding server $B_{i \bmod n}$.

S1-2. After A_i and $B_{i \bmod n}$ receive S_{A_decoy} and S_{B_decoy}, they both need to detect whether there is an eavesdropper Eve in the quantum channel by decoy check. Next Load_Balancer_A and Load_Balancer_B tell the positions and basis of decoy states to A_i and $B_{i \bmod n}$. After that, A_i and $B_{i \bmod n}$ extract out the decoy states of S_{A_decoy} and S_{B_decoy}, and measure them in the informed measurement basis. Then A_i and $B_{i \bmod n}$ broadcast their measurement results to Load_Balancer_A and Load_Balancer_B. Later, Load_Balancer_A and Load_Balancer_B compute error rates which would be compared with the predefined threshold.

S1-3. When any one of Load_Balancer_A and Load_Balancer_B finds the error rate is higher than predefined threshold, a potential eavesdropper Eve is considered to exist in the channel, and turn to S1-1. Otherwise, they can send their S_A and S_B to CA, and move to S1-4.

S1-4. CA receives both sequences from Load_Balancer_A and Load_Balancer_B and performs a Bell measurement on each pair (a, b) of the sequences. CA encodes and records the measurement results as R_{AB}. The encoding rule is $|\phi^+\rangle_{AB} \rightarrow 00$, $|\phi^-\rangle_{AB} \rightarrow 01$, $|\psi^+\rangle_{AB} \rightarrow 10$, $|\psi^-\rangle_{AB} \rightarrow 11$. CA sends R_{AB} to Load_Balancer_A and Load_Balancer_B across the classical channel. Load_Balancer_A and Load_Balancer_B then resend to A_i and $B_{i \bmod n}$.

S1-5. Once A_i and $B_{i \bmod n}$ receive R_{AB} successfully, Load_Balancer_A and Load_Balancer_B would announce the basis of each states in S_A and S_B. It is important to note that by removing the decoy states, A_i and $B_{i \bmod n}$ can both obtain S_A and S_B. And then A_i and $B_{i \bmod n}$ keep the bits under the same basis as the raw key when $R_{AB} = 11$.

S1-6. Considering the effect of eliminating the noise in the actual channel, A_i and $B_{i \bmod n}$ choose part of their raw keys to estimate error rate and detect eavesdropping. If error rate is below the threshold, then the channel is secure. Then both A_i and $B_{i \bmod n}$ can obtain the initial key K_{A_i} and K_{B_i} by using error correction and privacy amplification, where K_{A_i} and K_{B_i} are in $\{0, 1\}^2$. Bit 0 represents Z basis, and bit 1 represents X basis.

S1-7. Repeat above steps until all clients finish the registration and obtain the shared key with a specific server in the network. When $m > n$ and $i > n$, $B_{i \bmod n}$ needs store more than one keys in his own memory securely so that the corresponding clients can authenticate each other in the next phase.

It should be noted that if no eavesdropper is found in the communication process, the shared key as the authentication key can be used only once, and it also needs to be updated dynamically on a regular

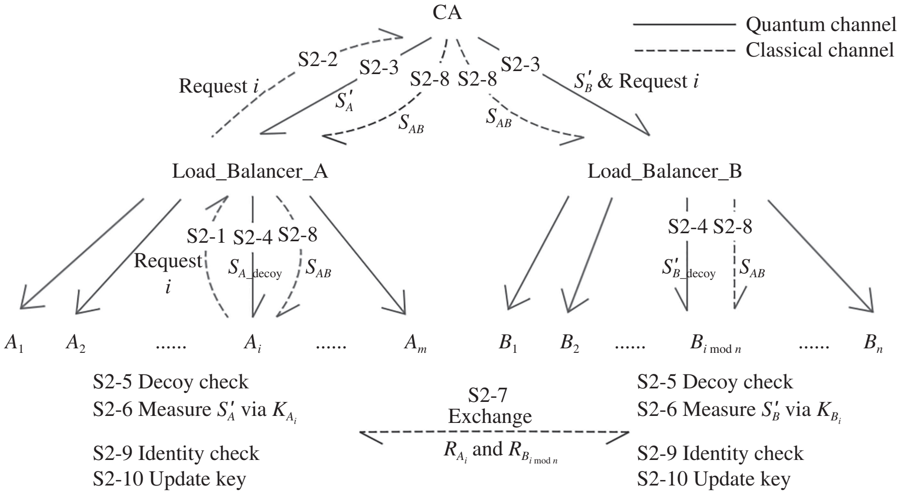

Figure 3 Information flow of each step in mutual identity authentication phase.

basis, so that the client and the server have new random shared authentication key to ensure the security of the shared key; otherwise, if the eavesdropper is found, the shared authentication key in the network needs to be updated immediately.

3.2 Phase 2: mutual identity authentication phase

The purpose of Phase 2 is to determine the legitimacy of the client and the server through the shared authentication key, and ensure the identity authentication between the client and the server. In order to prevent man-in-the-middle attack, through non-distinguishability of non-orthogonal states, semi-honest CA would prepare non-orthogonal states and send them to A_i and $B_{i \bmod n}$. For the purpose of enhancing efficiency and saving resources, Load_Balancer_A will selectively forward the authentication requests of clients under the first-in-first-out (FIFO) principle, so that the work efficiency of servers can be maximized. In the protocol, we assume that the channels between CA and Load_Balancer_A, Load_Balancer_B are unjammable. Figure 3 shows the specific information flow in Phase 2.

General steps of mutual identity authentication phase.

S2-1. When a registered client A_i wants to delegate computing tasks to a remote quantum server, A_i needs to send request i to Load_Balancer_A firstly across the classical channel.

S2-2. After receiving request i, Load_Balancer_A puts it into request queue. Then Load_Balancer_A checks whether the corresponding server is available, and according to FIFO principle to retransmit the request i to CA.

S2-3. CA prepares randomly non-orthogonal states $|\varphi\rangle_{AB} \in \{|\phi^-\rangle_{AB}, |\psi^+\rangle_{AB}, |\Psi^+\rangle_{AB}, |\Phi^-\rangle_{AB}\}$ with length of $4k$, where

$$\left|\phi^-\right\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),$$

$$\left|\psi^+\right\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle),$$

$$\left|\Phi^-\right\rangle = \frac{1}{\sqrt{2}}\left(\left|\phi^-\right\rangle - \left|\psi^+\right\rangle\right) = \frac{1}{\sqrt{2}}(|0-\rangle - |1+\rangle) = \frac{1}{\sqrt{2}}(|+1\rangle - |-0\rangle),$$

$$\left|\Psi^+\right\rangle = \frac{1}{\sqrt{2}}\left(\left|\phi^-\right\rangle + \left|\psi^+\right\rangle\right) = \frac{1}{\sqrt{2}}(|0+\rangle + |1-\rangle) = \frac{1}{\sqrt{2}}(|+0\rangle + |-1\rangle),$$

and transmits particle A's sequence S'_A and particle B's sequence S'_B with length of $4k$ to Load_Balancer_A and Load_Balancer_B, respectively.

S2-4. After Load_Balancer_A and Load_Balancer_B receive S'_A and S'_B, they would generate two sequence of decoy states with fixed length and insert decoy states into S'_A and S'_B. New sequences are denoted by $S'_{A-\text{decoy}}$ and $S'_{B-\text{decoy}}$, which send to A_i and $B_{i \bmod n}$ separately with request i.

Table 2 Correlation of CA's non-orthogonal states and A_i's & $B_{i \bmod n}$'s measurement results

	$\|0\rangle_B$	$\|1\rangle_B$	$\|+\rangle_B$	$\|-\rangle_B$
$\|0\rangle_A$	$\|\phi^-\rangle_{AB}$	$\|\psi^+\rangle_{AB}$	$\|\Psi^+\rangle_{AB}$	$\|\Phi^-\rangle_{AB}$
$\|1\rangle_A$	$\|\psi^+\rangle_{AB}$	$\|\phi^-\rangle_{AB}$	$\|\Phi^-\rangle_{AB}$	$\|\Psi^+\rangle_{AB}$
$\|+\rangle_A$	$\|\Psi^+\rangle_{AB}$	$\|\Phi^-\rangle_{AB}$	$\|\psi^+\rangle_{AB}$	$\|\phi^-\rangle_{AB}$
$\|-\rangle_A$	$\|\Phi^-\rangle_{AB}$	$\|\Psi^+\rangle_{AB}$	$\|\phi^-\rangle_{AB}$	$\|\psi^+\rangle_{AB}$

S2-5. Similar to S1-2, A_i and $B_{i \bmod n}$ would detect whether there is an eavasdropper in the channel and compute the error rate. If error rate is below the threshold, then A_i and $B_{i \bmod n}$ would obtain the original sequences S'_A and S'_B and continue S2-6. Otherwise, return to S2-3.

S2-6. A_i and $B_{i \bmod n}$ measure S'_A and S'_B with the initial keys K_{A_i} and K_{B_i} generated in Phase 1, and obtain the measurement results R_{A_i} and $R_{B_{i \bmod n}}$ with length of $4k$.

S2-7. R_{A_i} and $R_{B_{i \bmod n}}$ with length of $3k$ are selected randomly and transmitted to $B_{i \bmod n}$ and A_i across the classical channel, together with the corresponding particle positions.

S2-8. CA encodes each of non-orthogonal states into classical bit sequence S_{AB} and broadcasts it unjammably to A_i and $B_{i \bmod n}$. The encoding rule is $|\phi^-\rangle_{AB} \rightarrow 00$, $|\psi^+\rangle_{AB} \rightarrow 01$, $|\Phi^-\rangle_{AB} \rightarrow 10$, $|\Psi^+\rangle_{AB} \rightarrow 11$.

S2-9. According to Table 2, A_i and $B_{i \bmod n}$ check independently whether the other side's measurement results are correct. In other words, assume that $|\varphi\rangle_{AB} = |\psi^+\rangle_{AB}$, when the measurement result from A_i is bit 1, the measurment result from $B_{i \bmod n}$ should be bit 0. Otherwise, prover $B_{i \bmod n}$ fails to authenticate for verifier A_i. A_i needs to return Phase 1 to reallocate another server. Conversely, if prover A_i fails to authenticate for verifier $B_{i \bmod n}$, $B_{i \bmod n}$ needs to abandon A_i and waits new authentication request.

S2-10. After success mutual identity authentication, A_i and $B_{i \bmod n}$ need update shared keys by keeping the rest of measurement results with length of k as new shared keys. Load_Balancer_A's next job is to forward new request i to CA via FIFO principle after checking which server is available. Ultimately, repeat S2-3 to S2-10 until all requests in the queue are forwarded.

In addition to the key update step S2-10 in the identity authentication phase, the unused identity authentication key also needs to be updated regularly to ensure the security of the protocol. In particular, the sequence of non-orthogonal states sent by CA contains the authentication string for identity authentication and the next shared identity key information. CA does not know the initial key between the client and the server, nor which bit string is selected between the client and the server for identity authentication, so CA cannot steal the information between the client and the server for mutual identity authentication.

3.3 Phase 3: multi-party blind quantum computation phase

Similar to Phase 2, this phase still deals with the computing requirements of different clients according to the FIFO principle. Once the client and the server authenticate each other's identities, they can directly carry out the subsequent blind quantum computation without waiting for other clients to complete the authentication. In the network, a significant advantage is that different clients can carry out blind quantum computation in parallel, which greatly improves the efficiency of the server in the network, and efficiently solves the requirements of multi-client quantum computing on demand. In this phase, with the help of the advantage that Alice only needs to carry out the measurement operation to complete the blind quantum computation in [24]. Any universal quantum resource states can be utilized in this phase such as cluster state [32], brickwork state [3] or weighted graph state [37]. In order to describe the specific process of BQC, brickwork state is used in the following phase, which can implement any unitary gates via a universal gate set $\left\{\mathrm{CNOT}, H, \frac{\pi}{8}\right\}$. Brickwork state can realize universal quantum computation only through the single-qubit measurements on the X-Y plane. The structure of brickwork state is shown in Figure 1 in supplemental material of [3]. The measurement pattern of brickwork state is from left to right. So the qubits of first column is the input of the algorithm, and the qubits of the last column is the output.

General steps of blind quantum computation phase.

S3-1. After A_i and $B_{i \bmod n}$ authenticate each other successfully, $B_{i \bmod n}$ should generate a quantum resource state, for example brickwork state $G_{a\times b}$, where $b \equiv 5 \pmod{8}$.

S3-2. $B_{i \bmod n}$ sends a qubit (x, y) of brickwork state to A_i via quantum channel, and keeps the rest of qubits in the quantum memory.

S3-3. A_i computes the real measurement angle $\theta'_{x,y} = (-1)^{s^X_{x,y}}\theta_{x,y} + s^Z_{x,y}\pi$, where $\theta_{x,y} \in \{-\frac{\pi}{4}, 0, \frac{\pi}{4}, \frac{\pi}{2}\}$ is the desired measurement angle, $s^X_{x,y}$ is summation (modulo 2) of all previous measurement results in $X_{x,y}$, $s^Z_{x,y}$ is summation (modulo 2) of all previous measurement results in $Z_{x,y}$, $s^X_{0,y} = s^Z_{0,y} = 0$. Then A_i needs to measure the received qubit (x, y) in basis $\{|\pm\theta'_{x,y}\rangle\} = \{\frac{1}{\sqrt{2}}(|0\rangle \pm \mathrm{e}^{-\mathrm{i}\theta'_{x,y}}|1\rangle)\}$. The measurement result is recorded as $s_{x,y} \in \{0, 1\}$.

S3-4. Repeat above S3-2 and S3-3 until the computation is completed.

By using the brickwork state as a general quantum resource state, we realized the single-client single-server blind quantum computation in the network. The measurement angle of the client according to the algorithm will be affected by the previous measurement result, which is also the characteristic of the MBQC model. Based on MBQC model, the desired algorithms can be achieved by lay-by-layer measurement, which exists a dependency between measurement angles of different qubits in different layers of the brickwork state. In an actual multi-client and multi-server network, it may be necessary to deal with the blind quantum computing requests of many clients in network. As long as it is not a client that corresponds to the same server, parallel computing can be performed, which greatly improves the efficiency in network. Work efficiency also ensures that the needs of many clients in the network can be efficiently solved. In addition, because this phase does not involve the participation of Load_Balancer_A, Load_Balancer_B and CA, and the identity authentication phase does not affect the normal execution of blind quantum computing. So when there are available servers in the network, identity authentication and multi-party blind quantum computation can be performed simultaneously according to the FIFO principle, which ensures the fairness of the protocol and the efficient work of the servers in the network.

4 Correctness, blindness and security analysis

In this section, we show the correctness, blindness and security of the proposed multi-party BQC protocol, which are three basic elements of analyzing BQC protocols. Each analysis would be considered in each phase of the proposed protocol. Because the correctness of Phases 1 and 2 is shown in Section 3. So correctness analysis here is focused on Phase 3. Blindness refers to the privacy of inputs, algorithms and outputs of the clients while delegating quantum computing. Hence the blindness analysis here concerns in Phase 3 as well. As for the security analysis, the security of all three phases should be discussed under insider and outsider attacks.

4.1 Correctness analysis

Correctness refers to the correct outputs of the algorithms if the clients run the correct pattern in Phase 3. It should be noted that Phases 1 and 2 are completely uncorrelated with the correctness of client's outputs. Assume both clients and servers follow the steps in Phase 3. Then the outputs are correct.

Proof. Brickwork state we used in Phase 3 is same as the resource state in [3]. In MBQC, brickwork state is a universal quantum state that can implement any unitary operations (gates) through a universal gate set $\{\mathrm{CNOT}, H, \frac{\pi}{8}\}$. Figures 3–6 in supplemental material of [3] show the implementation of each gate by measuring the qubits of a brickwork state's unit in $\{|\pm\theta'_{x,y}\rangle\}$ basis, where $\theta'_{x,y} = (-1)^{s^X_{x,y}}\theta_{x,y} + s^Z_{x,y}\pi$ and $\theta_{x,y} \in \{-\frac{\pi}{4}, 0, \frac{\pi}{4}, \frac{\pi}{2}\}$. Figure 7 in supplemental material of [3] illustrates the stacking style of three unitary gates of brickwork state if the input is four qubits. We can extend the three unitary gates with four input states to a larger brickwork state, so the clients in the proposed protocols can finish computing tasks by layer-by-layer measurements. If the output is quantum information, then the servers need to send the last layer to the clients after clients' measurements. If the output is classical information, then the client can obtain the outcomes directly through the client's previous measurement results. Therefore, the correctness of the protocol is proved via MBQC model in [3].

4.2 Blindness analysis

The blindness analysis of the proposed protocol is mainly considered in stage of blind quantum computing, that is in Phase 3. The blindness of BQC should guarantee the privacy of inputs, algorithms and outputs. The preparation of the inputs is included in the client's computation part so that the blindness of the inputs is kept secret on the client side. Without loss of generality, client's measurement angles of the proposed BQC are the algorithms that the client wants to keep privacy. So the blindness proofs the protocol can be verified via the blindness of client's measurement angles and outputs. Similar with the mathematical proofs in supplemental material of [24, 38], Bayes' theorem can be utilized to prove the blindness of measurement angles and outputs. That is, the conditional probability distributions of measurement angles and outputs known by the servers should be equal to their priori probability distributions. When the servers obtain some classical information such as the measurement angles of the clients at any time, the servers would attempt to learn something about the clients' computation angles or outputs from the measurement outcomes of any positive-operator valued measurements (POVMs) on their systems.

Proof. Suppose A is the random variable related with the measurement results of the clients' algorithms, and O is the random variable related with the outputs of the algorithms. Let T be the random variable related with the time that the servers choose to get the clients' privacy, and K be the random variable related with the servers' knowledge about the clients' algorithms via POVMs. According to no-signaling principle, the choice of measurement angles and the servers' knowledge about the clients' algorithms are independent events. Then we can get

$$\begin{aligned}
P(A=\theta'_{x,y}|T=\tau_{x,y},K=k_{x,y}) &= \frac{P(A=\theta'_{x,y},T=\tau_{x,y},K=k_{x,y})}{P(T=\tau_{x,y},K=k_{x,y})}\\
&= \frac{P(T=\tau_{x,y}|A=\theta'_{x,y},K=k_{x,y})P(A=\theta'_{x,y},K=k_{x,y})}{P(T=\tau_{x,y},K=k_{x,y})}\\
&= \frac{P(T=\tau_{x,y}|A=\theta'_{x,y},K=k_{x,y})P(A=\theta'_{x,y})P(K=k_{x,y})}{P(T=\tau_{x,y}|K=k_{x,y})P(K=k_{x,y})}\\
&= P(A=\theta'_{x,y})\frac{P(T=\tau_{x,y}|A=\theta'_{x,y},K=k_{x,y})}{P(T=\tau_{x,y}|K=k_{x,y})}\\
&= P(A=\theta'_{x,y}).
\end{aligned}$$

As shown in above formulas, it implies that the conditional probability distribution of measurement angles known by the severs is equal to the prior probability distribution of the clients' measurement angles. It means that the servers cannot determine the algorithms of the clients even if they perform any POVMs on their systems.

In a similar way, the blindness of the outputs of the algorithms can be proved as follows:

$$\begin{aligned}
P\left(O=o|T=\tau_{x,y},K=k_{x,y}\right) &= \frac{P\left(O=o,T=\tau_{x,y},K=k_{x,y}\right)}{P\left(T=\tau_{x,y},K=k_{x,y}\right)}\\
&= \frac{P\left(T=\tau_{x,y}|O=o,K=k_{x,y}\right)P\left(O=o,K=k_{x,y}\right)}{P\left(T=\tau_{x,y},K=k_{x,y}\right)}\\
&= \frac{P\left(T=\tau_{x,y}|O=o,K=k_{x,y}\right)P(O=o)P\left(K=k_{x,y}\right)}{P\left(T=\tau_{x,y}|K=k_{x,y}\right)P\left(K=k_{x,y}\right)}\\
&= P(O=o)\frac{P\left(T=\tau_{x,y}|O=o,K=k_{x,y}\right)}{P\left(T=\tau_{x,y}|K=k_{x,y}\right)}\\
&= P(O=o).
\end{aligned}$$

It shows that the conditional probability distribution of outputs known by the severs is equal to the prior probability distribution of the outputs. So the outputs of the algorithms are independent of the server's knowledge via any POVMs at any time. Therefore, the blindness of the measurement angles and outputs is satisfied by using the Bayes' theorem. Just as the advantages of no-signaling principle, the measurement angles are only computed and known by the clients. No matter which type of POVMs is chosen, the servers cannot get any information or knowledge about the clients' algorithms and outputs.

4.3 Security analysis

In this network, in addition to the multiple classic clients and multiple quantum servers that must be included in the blind quantum computation protocol, it also involves the roles of providing request queue management, key distribution, and mutual identity authentication, that is, CA, Load_Balancer_A and Load_Balancer_B. In security analysis, it is necessary to prevent insider attacks (CA, client, server) and outsider attacks (impersonation attack, intercept-resend attack, entangle-measure attack). The protocol consists of three phases—the registration phase (quantum key distribution stage), the mutual identity authentication phase, and the blind quantum computing phase. Next, the security under insider attacks and outsider attacks in each phase will be discussed separately.

4.3.1 *Security analysis of Phase 1*

The security of Phase 1 mainly relies on the MDI-QKD protocol, and the MDI-QKD protocol has been strictly proven to be safe [39]. MDI-QKD solves the security vulnerability that measurement equipment may bring. Under actual experimental conditions, it is safe under the attack of the defects of quantum devices, providing an information-theoretically secure quantum key distribution. Neither CA nor Eve can determine whether the quantum states in the sequence sent by Load_Balancer_A and Load_Balancer_B were prepared under the X basis or Z basis, so any dishonest measurement results may be discovered. On the one hand, assuming that the quantum states of the corresponding positions in the Load_Balancer_A and Load_Balancer_B are $|0\rangle_A$ and $|1\rangle_B$ respectively, the measurement result of the dishonest CA would only be $|\psi^+\rangle_{AB}$ and $|\psi^-\rangle_{AB}$. The occurrence probability of the two measurement results is 1/2. If the dishonest CA returns the wrong measurement result $|\phi^+\rangle_{AB}$ and $|\phi^-\rangle_{AB}$, the detection probability of dishonest behavior is 1/2. In the error detection, the probability that the dishonest CA is discovered is proportional to the length of the sequence, so the length of the sequence needs to be determined appropriately so that the dishonest CA can be discovered with a certain probability. On the other hand, when Eve maliciously intercepted the Bell measurement results of CA, according to the nature of MDI-QKD, the measurement results only show whether the quantum states of Load_Balancer_A and Load_Balancer_B are the same or opposite, and it cannot be inferred the shared identity key. For example, if Eve obtains the result $|\phi^+\rangle_{AB}$, because $|\phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = \frac{1}{\sqrt{2}}(|++\rangle + |-\rangle)$, then Eve can only get that the quantum states of Load_Balancer_A and Load_Balancer_B are the same. Eve cannot judge whether the quantum state is prepared in X basis or Z basis. So the key distribution process is absolutely secure under the attacks from dishonest insider CA or outsider Eve. In addition, because there may be noise in the actual channel, Eve may obtain part of the key information under the cover of the noise, which ultimately causes the keys of Alice and Bob to be completely different, and directly affects the security and accuracy of the key. As long as the bit error rate is within an acceptable range, then the security enhancement technology can reduce the mutual information obtained by Eve to an arbitrary small. Therefore, the accuracy and security of the shared key are improved by correcting the error of the raw key and applying secret enhancement technology.

In the protocol, CA is semi-honest and does not affect the security of the protocol. That is to say, whether the CA in the protocol is honest or not, the shared key between the legitimate client and the specific server can always be generated safely. However, according to the premise of MDI-QKD, in this stage, we need to assume that the single photon preparation devices of Load_Balancer_A and Load_Balancer_B are safe. For the classical clients in the registration stage, because the key distribution process of different clients is an independent event, whether the client is camouflaged by Eve in this phase will not affect the next legitimate client to obtain a secure initial key. In addition, because the MDI-QKD process of each legitimate client is independent, combined with the decoy state check technology, eavesdropper Eve in the network will have a certain probability to be found, preventing the intercept-resend attack and entangle-measure attack. A key assumption in MDI-QKD is that classical channels cannot be eavesdropped. Therefore, even if Eve tries to carry his computing tasks by pretending to be the client that just sends the legitimate request, and assuming that Eve has passed the decoy check, however, Eve cannot get the CA's measurement result R_{AB} by eavesdropping the classical channel, and finally Eve cannot share the key with the legal server, and cannot reach his goal through the impersonation attack.

4.3.2 *Security analysis of Phase 2*

In Phase 2, on the basis of MDI-QKD, a scheme of mutual identity authentication is provided for both clients and servers. In the classical mutual identity authentication protocol, using the two pairs of keys shared between CA and client, CA and server, honest CA needs two independent processes to authenticate the identity of client and server respectively. However, in the authentication phase, the information flow is transmitted from CA to the client or the server. The role of CA is only to prepare and transmit non-orthogonal states sequence. The client and server can use only one pair of initial shared key to perform mutual identity authentication at the same time. They will not return the key information about identity authentication to the CA, and the bit string used by the client and server for identity authentication is unknown to the CA. Compared with the CA in the classical authentication, the CA in Phase 2 cannot get the identity key and authentication information of the legitimate client and the server even after repeated attempts. Therefore, no matter there is dishonest CA or Eve in Phase 2, the security of the client and the server can be guaranteed by using the initial security key and randomly selected authentication string shared between the client and the server.

In addition, because the CA in Phase 2 sends the non-orthogonal states, based on the Heisenberg's uncertainty principle and non-cloning theorem, the non-orthogonal states cannot be distinguished accurately. Any operation on the quantum state may cause changes in the measurement results of the receiver, so it provides unconditional security in theory. Eve cannot distinguish the intercepted non-orthogonal states with ambiguity, so the identity authentication under Eve's intercept-measure-resend attack is secure. Of course, through the use of decoy check technology, we can also avoid Eve's intercept-resend attack and entangle-measure attack, because the client and the server will discover the existence of these attacks with non-zero probability. It should be noted that in order to ensure the security of the shared identity key, the shared key used as the authentication key in this protocol is only applicable once, and the identity key will be updated dynamically on a regular basis. The security of the authentication process depends on whether the previous shared identity key is updated, so it is very important to use MDI-QKD to ensure the security of the initial authentication key. The update of the shared identity key avoids Eve using the information of the shared key leaked during the authentication to obtain the quantum computing resources. Furthermore, before the client and the server independently check each other's identity, they need to publish the partial measurement results and corresponding positions. However, whether Eve pretends to be a client or a server, he cannot publish the measurement results and the corresponding location correctly. Hence, after CA publishes the non-orthogonal state, that is, when verifier Alice or Bob checks the identity of the other party, he will realize the existence of illegal Eve in tampered quantum channel and/or classical channel.

In particular, mutual identity authentication is a necessary step before blind quantum computation, so Eve may interfere with the authentication process between the legal client and the server through the denial-of-service attack, and then influence the blind quantum computing of the legal client in the queue. Eve constantly sends requests to Load_Balancer_A through pretending to be a client. However, because Eve does not have the shared key in the registration phase, Eve cannot accurately measure the non-orthogonal states from the CA, so Bob can detect Eve's illegal identity in the mutual identity authentication phase which would fail Eve's identity authentication. Then Load_Balancer_A processes the request of the next client in the queue according to the FIFO principle, so even if Eve exists in the network, it will not affect the normal processing of the request of other clients.

4.3.3 *Security analysis of Phase 3*

In Phase 3, based on no-signaling principle, a single-server BQC scheme was proposed, which shows a stronger security which is composable security [29]. No-signaling principle is more fundamental than quantum mechanics, so the security of this BQC scheme is inherent and it satisfies device-independent security. No matter the server Bob only sends the qubits to Alice, but it will not get the information about what measurement has been made by Alice to the received qubits, so Bob will not get Alice's information. In most BQC protocols, there needs to be two-way communication between the classical client and the quantum server. However, in our BQC protocol where Alice measures, only one-way communication is needed, that is, there is only one-way information flow from Bob to Alice. Therefore, this protocol offers a simpler and stronger security. As shown in [24], this BQC protocol offers the device-independent security, which means the less requirements for the client's device. That is to say, even the client's device does not work correctly, Bob, CA or Eve cannot obtain any information about Alice's computation, which keeps the security of this phase both in insider or outsider attacks.

5 Comparison and conclusion

In this section, the details of the protocol are discussed in three phases. Next, we make multiple aspects of comparison between the proposed multi-party BQC protocol and common BQC protocols. Through the comparison shown in Table 3, the contributions of the proposed protocol are concluded as well. At last, the future work is also illustrated.

Table 3 Comparison between our proposed protocol and other BQC protocols

	Our multi-party protocol	Single-server protocol [24]	Single-server protocol [26]	Double-server protocol [31]	Triple-server protocol [4]
Client's capability	Only measuring states	Only measuring states	Preparing states	Classical	Preparing states
Server's capability	Preparing states	Preparing states	Measuring states	Measuring states	Measuring states
Server's number	n	One	One	Two	Three
Third party	Semi-honest	None	Trusted	Trusted	None
Resources states	Any universal resource state	Brickwork state	Brickwork state	Hyperentangled state	Bell states
Extra features	Identity authentication, multi-party	None	Identity authentication	Noisy	None

In conclusion, a multi-party blind quantum computation protocol with mutual identity authentication was proposed to meet the identity authentication requirements in field of delegated quantum computing. There are three main contributions in the protocol. Firstly, the almost classical clients can securely delegate their computing tasks to a remote quantum server after mutual identity authentication. Compared with other BQC protocols with almost classical clients, quantum memory is not required in the protocol. Along with the device-independent security of the protocol, it embodies to the significance in theory and experiment. Secondly, quantum authentication is provided novelly in the field of BQC to meet the basic requirements of identity authentication in quantum cryptography. Finally, the construction of quantum multi-party network in this protocol will provide a better reference for the popularization and practicality of entrusted quantum computing in the future. Besides, the quantum resource states in the protocol is general which makes the protocol adapted to other multi-party BQC protocols.

Specifically speaking, the proposed protocol consists of three phases: registration phase, mutual identity authentication phase and blind quantum computing phase, which relate to the distribution of initial identity key, the identity authentication of requested client and the corresponding server and the secure single-server blind quantum computation, respectively. Learned from classical computing network, load balancers are introduced to check the availability of each requested server and manage different clients' requests. In Phase 1, the combination of decoy technology and MDI-QKD improves the anti-attack ability of the system from the aspect of experimental implementation, so as to improve the availability of the system. Then in Phase 2, on the one hand, the sequences of non-orthogonal states consist of authentication string and next shared key's information, which protect the security under Eve's attacks. On the other hand, the simultaneous mutual identity authentication process via semi-honest CA improves the efficiency and accuracy of the protocol in some detail. In Phase 3, based on no-signaling principle, a practical single-server blind quantum computation scheme with only Alice's measuring is proposed and provides a stronger security than other BQC protocols. Therefore, the proposed multi-party BQC protocol with identity authentication would contribute to future work of multi-party BQC in real network.

In order to elaborate on the contributions of the proposed BQC protocol, we make a comparison between the proposed multi-party BQC protocol and other common BQC protocols with single-server, double-server and triple-server in [4, 24, 26, 31]. As shown in Table 3, there are four advantages of the proposed multi-party BQC protocol. First of all, multiple classical clients with limited quantum capability who can measure states could finish 'cloud' quantum computing securely with a certain assigned server in the multi-party network, which has enriched the research in multi-party BQC networks. Secondly, in our proposed protocol, any universal quantum resource state can be utilized which makes it more flexible and applicable to other universal BQC protocols. Thirdly, combined with MDI-QKD, this BQC protocol guarantees the security of pre-shared keys. Mutual identity authentication of BQC in network makes the protocol more reliable and practical in the future. Moreover, inspired by distributed cloud computing, load balancers are introduced to manage the requests from different clients fairly and dynamically. Hence, the work efficiency of servers is maximized and the latency time of each requested client is minimized in the proposed multi-party network. Obviously, the reliability and efficiency of the proposed protocol can be reflected in real experiments.

Besides from the identity authentication, it would be interesting to research a verifiable multi-party BQC protocol with mutual identity authentication in multi-party network, so that the client can prevent the server's dishonest behaviors and verify the outputs' correctness in follow-up work. For a verifiable protocol, it also brings some additional overhead and cost, such as the difficulty of preparing and maintaining entangled states between multi quantum servers. Therefore, whether a classical client with no quantum capability can achieve a secure verifiable BQC with mutual identity authentication needs to be further discussed.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61671087, 61962009, 61003287), Huawei Technologies Co. Ltd. (Grant No. YBN2020085019), and the Fund of the Fundamental Research Funds for the Central Universities (Grant No. 2019XD-A02).

References

1 Arrighi P, Salvail L. Blind quantum computation. Int J Quantum Inform, 2006, 4: 883–898

2 Childs A M. Secure assisted quantum computation. Quantum Inform Comput, 2005, 5: 456–466

3 Broadbent A, Fitzsimons J, Kashefi E. Universal blind quantum computation. In: Proceedings of 2009 50th Annual IEEE Symposium on Foundations of Computer Science, 2009. 517–526

4 Li Q, Chan W H, Wu C, et al. Triple-server blind quantum computation using entanglement swapping. Phys Rev A, 2014, 89: 040302

5 Raussendorf R, Briegel H J. A one-way quantum computer. Phys Rev Lett, 2001, 86: 5188–5191

6 Greganti C, Roehsner M C, Barz S, et al. Demonstration of measurement-only blind quantum computing. New J Phys, 2016, 18: 013020

7 Huang H L, Zhao Q, Ma X, et al. Experimental blind quantum computing for a classical client. Phys Rev Lett, 2017, 119: 050503

8 Huang H L, Bao W S, Li T, et al. Universal blind quantum computation for hybrid system. Quantum Inf Process, 2017, 16: 199

9 Yin H L, Chen T Y, Yu Z W, et al. Measurement-device-independent quantum key distribution over a 404 km optical fiber. Phys Rev Lett, 2016, 117: 190501

10 Morimae T, Koshiba T. Impossibility of perfectly-secure delegated quantum computing for classical client. 2014. ArXiv: 1407.1636

11 Hayashi M, Morimae T. Verifiable measurement-only blind quantum computing with stabilizer testing. Phys Rev Lett, 2015, 115: 220502

12 Morimae T, Fitzsimons J F. Post hoc verification with a single prover. 2016. ArXiv: 1603.06046

13 Morimae T. Verification for measurement-only blind quantum computing. Phys Rev A, 2014, 89: 060302

14 Gheorghiu A, Kashefi E, Wallden P. Robustness and device independence of verifiable blind quantum computing. New J Phys, 2015, 17: 083040

15 Gheorghiu A, Wallden P, Kashefi E. Rigidity of quantum steering and one-sided device-independent verifiable quantum computation. New J Phys, 2017, 19: 023043

16 Yang Y G, Wen Q Y. An efficient two-party quantum private comparison protocol with decoy photons and two-photon entanglement. J Phys A-Math Theor, 2009, 42: 055305

17 Tseng H Y, Lin J, Hwang T. New quantum private comparison protocol using EPR pairs. Quantum Inf Process, 2012, 11: 373–384

18 Zhang W W, Zhang K J. Cryptanalysis and improvement of the quantum private comparison protocol with semi-honest third party. Quantum Inform Process, 2013, 12: 1981–1990

19 Wei C Y, Cai X Q, Wang T Y, et al. Error tolerance bound in QKD-based quantum private query. IEEE J Sel Areas Commun, 2020, 38: 517–527

20 Gao F, Qin S J, Huang W, et al. Quantum private query: a new kind of practical quantum cryptographic protocol. Sci China-Phys Mech Astron, 2019, 62: 70301

21 Wei C Y, Cai X Q, Liu B, et al. A generic construction of quantum-oblivious-key-transfer-based private query with ideal database security and zero failure. IEEE Trans Comput, 2018, 67: 2–8

22 Barz S, Kashefi E, Broadbent A, et al. Demonstration of blind quantum computing. Science, 2012, 335: 303–308

23 Dunjko V, Kashefi E, Leverrier A. Blind quantum computing with weak coherent pulses. Phys Rev Lett, 2012, 108: 200502

24 Morimae T, Fujii K. Blind quantum computation protocol in which Alice only makes measurements. Phys Rev A, 2013, 87: 050301

25 Kong X Q, Li Q, Wu C, et al. Multiple-server flexible blind quantum computation in networks. Int J Theor Phys, 2016, 55: 3001–3007

26 Li Q, Li Z, Chan W H, et al. Blind quantum computation with identity authentication. Phys Lett A, 2018, 382: 938–941

27 Lo H K, Curty M, Qi B. Measurement-device-independent quantum key distribution. Phys Rev Lett, 2012, 108: 130503

28 Dunjko V, Fitzsimons J, Portmann C, et al. Composable security of delegated quantum computation. In: Proceedings of the 20th Annual International Conference on the Theory and Application of Cryptology and Information Security, 2014. 406–425

29 Morimae T, Koshiba T. Composable security of measuring-Alice blind quantum computation. 2013. ArXiv: 1306.2113

30 Liang M. Blind quantum computation with completely classical client and a trusted center. 2015. ArXiv: 1508.07778

31 Sheng Y B, Zhou L. Deterministic entanglement distillation for secure double-server blind quantum computation. Sci Rep, 2015, 5: 7815

32 Briegel H J, Browne D E, Dür W, et al. Measurement-based quantum computation. Nat Phys, 2009, 5: 19–26

33 Raussendorf R, Browne D E, Briegel H J. Measurement-based quantum computation on cluster states. Phys Rev A, 2003, 68: 022312

34 Rossi M, Huber M, Bruß D, et al. Quantum hypergraph states. New J Phys, 2013, 15: 113022

35 Hayashi M, Hajdušek M. Self-guaranteed measurement-based quantum computation. Phys Rev A, 2018, 97: 052308

36 Childs A M, Leung D W, Nielsen M A. Unified derivations of measurement-based schemes for quantum computation. Phys Rev A, 2005, 71: 032318

37 Hein M, Dur W, Eisert J, et al. Entanglement in graph states and its applications. 2006. ArXiv: quant-ph/0602096

38 Zhang X, Luo W, Zeng G, et al. A hybrid universal blind quantum computation. Inf Sci, 2019, 498: 135–143

39 Gottesman D, Lo H K, Lutkenhaus N, et al. Security of quantum key distribution with imperfect devices. In: Proceedings of International Symposium onInformation Theory, 2004. 136