SCIENTIFIC REPORTS
natureresearch

OPEN

New Fair Multiparty Quantum Key Agreement Secure against Collusive Attacks

Zhiwei Sun[1,2], Rong Cheng[1], Chunhui Wu[3] & Cai Zhang[4,5*]

Fairness is an important standard needed to be considered in a secure quantum key agreement (QKA) protocol. However, it found that most of the quantum key agreement protocols in the travelling model are not fair, i.e., some of the dishonest participants can collaborate to predetermine the final key without being detected. Thus, how to construct a fair and secure key agreement protocol has obtained much attention. In this paper, a new fair multiparty QKA protocol that can resist the collusive attack is proposed. More specifically, we show that in a client-server scenario, it is possible for the clients to share a key and reveal nothing about what key has been agreed upon to the server. The server prepares quantum states for clients to encode messages to avoid the participants' collusive attack. This construction improves on previous work, which requires either preparing multiple quantum resources by clients or two-way quantum communication. It is proven that the protocol does not reveal to any eavesdropper, including the server, what key has been agreed upon, and the dishonest participants can be prevented from collaborating to predetermine the final key.

It has been proven that Shor's algorithm can factor a large number and calculate the discrete logarithms in polynomial time by using a quantum computer. With the development of research on quantum computers, small-scale quantum computers have already been created by large companies and organisations around the world. If large-scale quantum computers become available in the not-too-distant future, current public-key cryptosystems like RSA or elliptic curves will become insecure. To solve this problem, it is necessary to select new techniques that are not vulnerable to quantum computers and to design, analyse, and implement new cryptographic schemes based on these techniques.

Quantum cryptography is a study of carrying out cryptographic tasks using the properties of quantum mechanics. Quantum key distribution (QKD), as a famous instance of quantum cryptographic tasks, enjoys information-theoretical security to exchange the key. QKD can detect outside attacks aiming at learning about the secret key by measuring the quantum system. However, the uncertainty principle shows that measuring a quantum system will unavoidable disturb it, which provides a method of detecting the presence of eavesdropping. A quantum cryptographic protocol is secure if no information about the secret key is leaked; otherwise, it will be aborted. So far various subfields of quantum cryptography have emerged to offer different functions, such as quantum secure direct communication[1–9], quantum private comparison[10–14], quantum signature[15,16], and quantum oblivious transfer[17].

In the past years, quantum key agreement (QKA) protocols have received much attention in the quantum cryptography world. Compared with quantum key distribution where one sends a generated key to the other one, quantum key agreement allows multiple parties to collaborate to equally produce a shared key. The security of QKA requires that no partial corrupted parties can determine the shared key and no information about the shared key can be obtained by any eavesdropper. There were only two parties involved in quantum key agreement protocols when they were studied at the beginning[18–24]. Later, they are generalized to the scenario where multiple parties are considered[25–36].

Unfortunately, Liu *et al*. showed that part of the parties in a multiparty QKA protocol can predetermine the final agreed key before the end of the protocol[37]. In other words, most of the existing QKA protocols cannot

[1]School of Artificial Intelligence, Shenzhen Polytechnic, Shenzhen, Guangdong, 518055, China. [2]Center for Quantum Computing, Peng Cheng Laboratory, Shenzhen, 518055, China. [3]Department of Computer Science, Guangdong University of Finance, Guangzhou, 510521, P.R. China. [4]College of Mathematics and Informatics, South China Agricultural University, Guangzhou, 510642, China. [5]School of Electronic and Electrical Engineering, University of Leeds, Leeds, LS2 9JT, UK. *email: zhangcai.sysu@gmail.com

resist collusive attack. One reason that the collusive attack can succeed in multiparty QKA protocols is that the malicious participants can share the the initial prepared quantum states with each other. When two parties in the particular position, they can calculate the bitwise exclusive OR result of all the other's secret key. With the result, they are able to predetermine the final agreed key. Thus, how to design a key agreement protocol which can be secure against collusive attack has obtained much attention[38–40]. On the other hand, a number of protocols have emerged where a user with limited quantum capabilities, delegates tasks to a server, who has the completely quantum power, which is known as delegated quantum computation[17]. Based on the idea of delegated quantum computation, we propose a multiparty QKA protocol in the client-server model.

Preliminaries

Let us review the existing travelling-type multi-party quantum key agreement (MQKA) protocol and the collusive attacks. Suppose that N participants $P_0, \ldots, P_{N-1}$ have secret bit-string keys $K_0, \ldots, K_{N-1}$, respectively.

Short review of the travelling-type MQKA protocol.

We will review the travelling-type MQKA protocol here, which has been discussed in ref. [37].

In the preparation stage, P_i ($i = 0, \ldots, N-1$) generates many entangled states, each of which is then divided into two parts. One of them called "the home qubit sequence" (denoted as R_i) will be kept, and the other one called "the travel qubit sequence" (denoted as S_i) will be sent out. P_i then generates decoy particles that are later inserted into S_i. The inserted S_i is denoted as S'_i. $P_0, \ldots, P_{N-1}$ stand in a circle such that P_i's neighborhoods are $P_{i-1modN}$ and $P_{i+1modN}$ ($P_0, \ldots, P_{N-1}$). All the S'_i are sent to $P_{i+1modN}$. When all the S'_i have been received by $P_{i+1modN}$, they detect attacks and encode $K_{i+1modN}$ into S_i (by removing decoy particles from S_i)'). Afterward, decoy particle will be inserted into the encoded sequence and the new sequence will be sent to next participant. This process is similar to what P_i does in the previous step. Each participant repeats this process. After all participants finish the above process, S_i forms a complete circle. S_i is then measured by P_i who obtains $K_0 \oplus K_1 \oplus \ldots \oplus K_{i-1} \oplus K_{i+1} \oplus \ldots \oplus K_{N-1}$. Finally, all participants can get the shared key.

Liu's collusive attacks against CT-MQKA protocol.

Liu's collusive attacks[37] consist of two stages. The first stage is the key-stealing stage and the the second stage is the key-flipping stage. In the first stage, the corrupted participants do their best to collaborate to computer the bitwise XOR outcome of the others' secret keys by exploiting various quantum resources. In the second stage, they then change the encoded keys in accordance with the above outcome to determine a fake shared key.

It has been shown in ref. [37] that any two parties P_i and P_j ($i > j$) can control the shared key if the following conditions hold:

$$i - j = \frac{N}{2} \qquad \text{for an even N;} \tag{1}$$

$$i - j = \frac{N-1}{2} \text{ or } \frac{N+1}{2} \qquad \text{for an odd N.} \tag{2}$$

Once Eq. (1) or Eq. (2) holds, the following attack can be launched by P_i and P_j. For easy described the attack, suppose N is an even number.

1. **The key-stealing stage**:

- When the protocol starts, P_i and P_j share the knowledge of R_i, S_i, K_i and R_j, S_j, K_j and the expected fake key K'.
- In the $(i-j)$-th period when P_j starts the protocol, upon receiving S_j, P_i is able to attain the bitwise XOR result of $K_{j+1}, K_{j+2}, \ldots, K_{i-1}$ according to the measurement outcomes of R_j and S_j. Analogously, P_j could obtain the XOR result of $K_{i+1}, K_{i+2}, \ldots, K_{j-1}$ in the $(N-i+j)$-th period when P_i starts the protocol.
- P_i and P_j exchange the above bitwise XOR results. Then, they can compute the legal shared key K in the $i-j$ period in advance.

2. **The key-flipping stage**:

- Suppose K' is the fake key that collusive participants want to share. In the $i-j$ period, P_i and P_j encode $K'_i = K_i \oplus K' \oplus K$ instead of K_i, and $K'_j = K_j \oplus K' \oplus K$ instead of K_j respectively. One can verify that any participant will obtain the fake final shared key K'.

Results

The proposed fair multiparty QKA protocol.

In the above attack, any two malicious parties P_i and P_j in particular positions can exchange the information about their initial prepared quantum states. Then they can collaborate with other to compute the the final shared key K before the last period. They can finally predetermine the fake key based on these information. Most MQKA protocols are therefore insecure against collusive attacks. To achieve the fairness property, two conditions should be removed. The first one is that the information about the initial prepared states cannot be shared among collusive parties. Without these information, any two malicious parties can obtain nothing about other parties' keys. Thus, they cannot compute the final shared key K in advance. There is no way for them to generate a fake final shared key. In order to launch Liu's attack[37], all quantum states generated by the honest parties should pass the malicious parties at least once. The situations in Sun's protocols[33,38], are a little different. The travelling model is divided into parts. Since the malicious parties are limited to only part of information about the other parties' keys before the last period, which makes them fail to computer the bitwise XOR outcomes of all the other's secret keys any more. In such way, Sun's protocols are secure against t-party collusive attacks. Here, $t < N$.

The first method will be employed to devise a fair MQKA protocol in this work. To make the collusive parties share nothing about the initial prepared states among them, these parties are restricted to generating initial states. The stage of initial states is delegated to a server. The server plays a role of generating the initial states, forwarding them to parties and announcing the generated initial states in the last period via authenticated classical channels. We assume that the server is semi-honest. In other words, the server will honestly follow the protocol and cannot collude with any other party but she may try to learn about extra information about the parties' secret keys, other than what the process of the protocol naturally implies. The parties are then only required to make measurements and do unitary operations. We also assume that the classical channels in our protocol are authenticated and the quantum channels are lossless and noiseless.

Suppose participants $P_1, \ldots, P_{N-1}$ have secret m-bit keys $K_0, \ldots, K_{N-1}$, respectively, they intend to generated a shared key K such that $K = K_0 \oplus \ldots \oplus K_i \oplus \ldots \oplus K_{N-1}$. The participants stand in a circle in the following way: P_i has P_{i-1} and P_{i+1} as his left and right neighbors, respectively, where $P_{i \pm N} = P_i$ for $0 <= i < N$.

Generally, our protocol will reveal nothing about the shared key to any eavesdropper, including the server. And it is also secure against the collusive attacks.

The detailed steps of our protocol can be described in the following:

1. **Preparation stage:** The server prepares N sequences $\{S_0, \cdots, S_{N-1}\}$, which are called the message sequences. Sequence S_i, $i = 0, \cdots, N - 1$ consists of m ordered single photons. Each single photon is randomly selected from $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$. To check for eavesdropping, the server prepares another N sequences $\{C_0, \cdots, C_{N-1}\}$ which is called the decoy sequence, and the decoy sequence C_i, $(i = 0, \cdots, N - 1)$ consists of m ordered single photons. The single photon is randomly in one of the states $\{|+\rangle, |-\rangle, |+y\rangle, |-y\rangle\}$. Here, $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$, $|\pm y\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm i|1\rangle)$, which are called decoy states. For all $i = 0, \cdots, N - 1$, the server randomly inserts C_i into S_i to get a new sequence S'_i which is called the travelling sequence, and sends S'_i to P_i.
2. **Detection stage:** After confirming that all the N parties, $P_0, \cdots, P_{N-1}$, have received the message sequences sent from the server, the server publishes the positions and corresponding bases of the decoy sequence in the travelling sequence. Based on these information, for $i = 0, \cdots, N - 1$, P_i can measure C_i in the correct bases. Then, he/she stores the measurement results and randomly publishes half of the measurement outcomes. Correspondingly, the server publishes the information of the initial states of the other half of C_i. By comparing the measurement results of the decoy sequence with their corresponding initial states, the server and P_i can calculate the error rate. If the error rate is lower than the predetermined threshold value, the protocol will be proceeded; otherwise, the protocol will be aborted and restarted from Step 1.
3. After the detection stage, P_i obtains the secure travelling sequence S_i. Here, $i = 0, \cdots, N - 1$. Each party P_i, $i = 0, \cdots, N - 1$, performs the following steps:

1) **Encoding stage:** P_i encodes K_i onto S_i by the following encoding rule: when the classical bit of the k_i is 1, the unitary operation $U = |0\rangle\langle 1| - |1\rangle\langle 0|$ is performed onto S_i. Otherwise, the identity unitary operation $I = |0\rangle\langle 0| + |1\rangle\langle 1|$ is made. The role of unitary operator U is to flip the quantum states, in other words, $U|0\rangle = -|1\rangle$, $U|1\rangle = |0\rangle$, $U| + \rangle = |-\rangle$, $U| - \rangle = -|+\rangle$. Then P_i rearranges the m decoy states generated by server in Step 1, and randomly inserts them into the encoded sequence to get a new one which is denoted as S_i^{i+1}. After the above encoding stage, P_i forwards S_i^{i+1} to P_{i+1}.
2) **Eavesdropping check stage:** The eavesdropping check stage is similar to the server and P_i did in Step 2. In other words, when P_{i+1} has received the sequence S_i^{i+1} from P_i, P_i tells P_{i+1} the decoy states' positions and the corresponding bases in the sequence S_i^{i+1}. According to these information, P_{i+1} measures the decoy sequence in the corresponding correct bases, stores them and randomly announces half of the measurement result. Then, P_i publishes the initial states of the other half decoy sequence. According to the announced information, i.e., the measurement results of the decoy sequence and the initial decoy sequence, they can calculate the error rate. If the error rate is lower than the predetermined threshold value, the protocol will be proceeded; otherwise, the protocol will be aborted and restarted from Step 1.
3) **Encoding stage:** After the detection phase, P_{i+1} obtains the message sequence S_i. He then encodes K_{i+1} onto S_i by the encoding rule in Step (1). Then P_{i+1} rearranges the m decoy states, and randomly inserts the decoy states into the encoded sequence to get a new one which is denoted as S_i^{i+2}. After the above encoding stage, P_{i+1} forwards S_i^{i+2} to P_{i+2}.
4) Then, the $N-1$ participants, $P_{i+2}, P_{i+3}, \cdots, P_{i-2}$ repeatedly execute the eavesdropping check stage and encoding state in the same way as in Steps (2) and (3).
5) When P_{i-1} receives S_i^{i-2} from P_{i-2}, P_{i-1} and P_{i-2} check for eavesdropping with the decoy states method. If the transmission is secure, P_{i-1} discards the decoy states and obtains the secure message sequence S_i, and he announces this fact to server.

4. Once all the P_{i-1} obtains the secure message sequence S_i, the server announces the positions and corresponding bases of the S_i to P_{i-1}. For each participant P_{i-1}, he then measures each of the message sequence S_i in the corresponding bases to obtain an m-bit string $K'_i = K_i \oplus K_{i+1} \oplus \cdots \oplus K_{i-2}$. Then P_{i-1} can deduce the final shared key $K_i^{'} \oplus K_{i-1} = K_0 \oplus \cdots \oplus K_i \oplus \cdots \oplus K_{N-1} = K$. Here, $i = 0, \cdots, N - 1$.

Note that the above protocol is considered in the semi-honest model, if there are malicious parties, the shared key K may be not identical. In order to prevent them from fooling the honest one, the N participants $P_0, \cdots, P_{N-1}$ can randomly select parts of the K to detect eavesdropping. If there is no malicious party, the rest of the K will be the final shared key. The following section will discuss the security analysis of the presented protocol.

Security analysis of the proposed protocol. First, we prove that the proposed protocol is secure against external eavesdropping. Then, we show that it is immune to attacks from internal eavesdropping.

Security against external eavesdropping. To detect outside eavesdropping, the decoy-state method is used in the presented protocol. The decoy-state method uses several non-orthogonal single states, $|+\rangle$, $|-\rangle$, $|+y\rangle$, $|-y\rangle$, which are randomly inserted in the message sequence. Because of quantum indistinguishability, Eve cannot distinguish between the message sequence and the decoy states. The Eve may apply the same operation on all the quantum states. Usually, the operation Eve makes is denoted as U_E which causes the message sequence to interact coherently with an auxiliary quantum system $|E\rangle$, which can be denoted as follows:

$$U_E|0\rangle|E\rangle = a|0\rangle|E_{00}\rangle + b|1\rangle|E_{01}\rangle, \tag{3}$$

$$U_E|1\rangle|E\rangle = c|0\rangle|E_{10}\rangle + d|1\rangle|E_{11}\rangle, \tag{4}$$

where $|a|^2 + |b|^2 = 1$ and $|c|^2 + |d|^2 = 1$. In the following part, we will prove that any malicious behavior by Eve will inevitably modify the photon statistic and expose her.

Since the decoy states involved in our protocol are $|+\rangle, |-\rangle, |+y\rangle$ and $|-y\rangle$, if Eve introduces no error in the eavesdropping check by participants, the general operation U_E must satisfy the following conditions:

$$\begin{aligned} U_E| + \rangle|E\rangle &= \frac{1}{\sqrt{2}}(a|0\rangle|E_{00}\rangle + b|1\rangle|E_{01}\rangle + c|0\rangle|E_{10}\rangle + d|1\rangle|E_{11}\rangle) \\ &= \frac{1}{2}(|+\rangle(a|E_{00}\rangle + b|E_{01}\rangle + c|E_{10}\rangle + d|E_{11}\rangle)) \\ &+ \frac{1}{2}(|-\rangle(a|E_{00}\rangle - b|E_{01}\rangle + c|E_{10}\rangle - d|E_{11}\rangle)) \\ &= \frac{1}{2}(|+\rangle(a|E_{00}\rangle + b|E_{01}\rangle + c|E_{10}\rangle + d|E_{11}\rangle)). \end{aligned} \tag{5}$$

$$\begin{aligned} U_E| - \rangle|E\rangle &= \frac{1}{\sqrt{2}}(a|0\rangle|E_{00}\rangle + b|1\rangle|E_{01}\rangle - c|0\rangle|E_{10}\rangle - d|1\rangle|E_{11}\rangle) \\ &= \frac{1}{2}(|+\rangle(a|E_{00}\rangle + b|E_{01}\rangle - c|E_{10}\rangle - d|E_{11}\rangle)) \\ &+ \frac{1}{2}(|-\rangle(a|E_{00}\rangle - b|E_{01}\rangle - c|E_{10}\rangle + d|E_{11}\rangle)) \\ &= \frac{1}{2}(|-\rangle(a|E_{00}\rangle - b|E_{01}\rangle - c|E_{10}\rangle + d|E_{11}\rangle)). \end{aligned} \tag{6}$$

$$\begin{aligned} U_E|+y\rangle|E\rangle &= \frac{1}{\sqrt{2}}(a|0\rangle|E_{00}\rangle + b|1\rangle|E_{01}\rangle + ic|0\rangle|E_{10}\rangle + id|1\rangle|E_{11}\rangle) \\ &= \frac{1}{2}(|+y\rangle(a|E_{00}\rangle - ib|E_{01}\rangle + ic|E_{10}\rangle + d|E_{11}\rangle)) \\ &+ \frac{1}{2}(|-y\rangle(a|E_{00}\rangle + ib|E_{01}\rangle + ic|E_{10}\rangle - d|E_{11}\rangle)) \\ &= \frac{1}{2}(|+y\rangle(a|E_{00}\rangle - ib|E_{01}\rangle + ic|E_{10}\rangle + d|E_{11}\rangle)). \end{aligned} \tag{7}$$

$$\begin{aligned} U_E|-y\rangle|E\rangle &= \frac{1}{\sqrt{2}}(a|0\rangle|E_{00}\rangle + b|1\rangle|E_{01}\rangle - ic|0\rangle|E_{10}\rangle - id|1\rangle|E_{11}\rangle) \\ &= \frac{1}{2}(|+y\rangle(a|E_{00}\rangle - ib|E_{01}\rangle - ic|E_{10}\rangle - d|E_{11}\rangle)) \\ &+ \frac{1}{2}(|-y\rangle(a|E_{00}\rangle + ib|E_{01}\rangle - ic|E_{10}\rangle + d|E_{11}\rangle)) \\ &= \frac{1}{2}(|-y\rangle(a|E_{00}\rangle + ib|E_{01}\rangle - ic|E_{10}\rangle + d|E_{11}\rangle)). \end{aligned} \tag{8}$$

From the above Eqs. (5–8), we can get

$$a|E_{00}\rangle - b|E_{01}\rangle + c|E_{10}\rangle - d|E_{11}\rangle = 0, \tag{9}$$

$$a|E_{00}\rangle + b|E_{01}\rangle - c|E_{10}\rangle - d|E_{11}\rangle = 0, \tag{10}$$

$$a|E_{00}\rangle + ib|E_{01}\rangle + ic|E_{10}\rangle - d|E_{11}\rangle = 0, \tag{11}$$

$$a|E_{00}\rangle - ib|E_{01}\rangle - ic|E_{10}\rangle - d|E_{11}\rangle = 0. \tag{12}$$

Here 0 denotes a column zero vector. Further, we can get $a = d = 1$, $b = c = 0$ and $|E_{00}\rangle = |E_{11}\rangle$. Therefore,

$$U_E|0\rangle|E\rangle = |0\rangle|E_{00}\rangle, \tag{13}$$

$$U_E|1\rangle|E\rangle = |1\rangle|E_{00}\rangle, \tag{14}$$

$$U_E|+\rangle|E\rangle = |+\rangle|E_{00}\rangle, \tag{15}$$

$$U_E|-\rangle|E\rangle = |-\rangle|E_{00}\rangle, \tag{16}$$

i.e., Eve introduces no error in the eavesdropping only when her ancillary state and the target photon $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ are product states. So outside eavesdroppers cannot obtain the shared key without being detected. In addition, each transmission of the qubit sequences is not a closed ring, i.e., the transmission is not a two-way quantum channel any more. Thus, the Trojan horse and invisible photon attacks can be naturally resisted.

Security against internal eavesdropping. As known to all, the dishonest parties in a protocol have more power than those from external eavesdroppers to attack the protocol. The dishonest parties can lie in the eavesdropping check stage or substitute the message sequence with their desired message sequence in order to predetermine the final shared key. Thus, all the proposed QKA protocols need to be secure against internal dishonest parites' attack.

Liu's collusive attack can be divided into two stages[37]: the key stealing stage and the key flipping stage. The key stealing stage or the key flipping stage must be destroyed in order to design a secure QKA protocol. In this paper, the proposed protocol which is secure in the stealing stage is analyzed as follows:

We first consider the worst case that there are $N-1$ dishonest parties and only one honest party, P_t, $t \in \{0, \cdots, N-1\}$. In order to predetermine the final shared key, the $N-1$ dishonest parties need to obtain P'_ts private key K_t before $P'_{t+1}s$ quantum sequence S_{t+1} is sent to P_t. If the dishonest parties have already obtained K_t, they can launch the following attack: When TP sends the message sequence S_{t+1} to P_{t+1}. Then, the $N-1$ dishonest participant $P_{t+1}, \cdots, P_{t-1}$ just forward the message sequence S_{t+1} to the next one using the decoy method. If P_{t-1} receives S_{t+1} from P_{t-2}, after the eavesdropping check stage, P_{t-1} encodes $K_t \oplus K'$ onto S_{t+1}, and sends the new sequence to P_t in the secure way. When the server announces the positions and corresponding bases of the S_{t+1} to P_t in step 4, it is easy to verify that the final key P_t obtained is the fake key $K_t \oplus K_t \oplus K' = K'$.

Fortunately, we will show that it is impossible to obtain P'_ts private key K_t before $P'_{t+1}s$ quantum sequence S_{t+1} is sent to P_t in our protocol. Since the initial quantum states are prepared by server and the server honestly executes the protocol and does not cooperate with any participant. He will not leak any information about the initial prepared quantum states to any participant before the step 4. In order to obtain K_t, the only way for the dishonest parties is to measure the message sequence just like the external Eve does. However, security against external eavesdropping has been proven in the above subsection. Thus, this kind of internal attack can be prevented.

Secondly, some dishonest parties may just intend to fool some parties, making the legitimate parties accept the fake key K' as the final shared key K. For example, when TP sends the message sequence S_{t+1} to P_{t+1}, the dishonest P_{t+1} can encode $K_{t+1} \oplus K_f$ in the encoding stage in order to fool the honest party P_t. Here, $t \in \{0, \cdots, N-1\}$, the key K_f is used to fool P_t. When the server announces the positions and corresponding bases of the S_{t+1} to P_t in step 4, it is easy to verify that the final shared key of P_t is the fake key $K_f \oplus K$. In order to detect the malicious behavior of the dishonest parties, the N parties can randomly choose part of the final shared key K to detect the error when they have already obtained K. If the error rate is higher than predetermined value, the protocol is abort. Otherwise, the rest bit of K is used for the final key.The details can be found in ref.[41].

Thirdly, the server may also try to learn extra information about participants' secret key from the protocol execution. Notice that the presented protocol is a one-way quantum channel, the server prepares the initial quantum states and sends them to the participant, but these quantum states will not be sent back to server. Thus, if the server tries to learn extra information about participants' secret key, he/she may need to measure the quantum channel, just like the external attackers do. Because of the decoy states method, this attack can be detected in the presented protocol. Thus, the server cannot get any information about the parties' secret key. If the server uses Trojan horse or invisible photon attacks, the method in ref.[42] can be used to resist these attacks.

Efficiency. In this section, we compare the qubit efficiency of different MQKA protocols. The qubit efficiency is defined as $\eta = \frac{c}{q+b}$[41]. Here, c represents the length of the final shared key, q denotes the number of the qubits required for encoding and eavesdropping process and b refers to the number of bits needed for decoding process.

QKA protocol	η	participants prepare message states	TP	Collusive Attacks
LGHW protocol	$\frac{1}{(N-1)N}$	$mN(N-1)$	No	Secure
HSX protocol	$\frac{1}{2N^2}$	mN	No	Secure
WSH protocol	$\frac{1}{2(N-1)N}$	$mN(N-1)$	No	Secure
Ours	$\frac{1}{3N}$	0	Yes	Secure

Table 1. Efficiency comparison. For easier comparison, let the key length is m, the number of participants is N, the detection rate $\kappa = 1$, the dishonest participants $t = N - 1$.

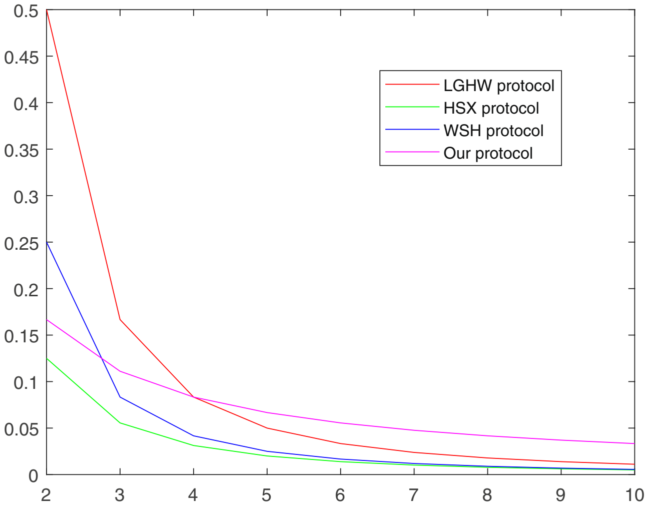

Figure 1. Efficiency comparison of the different protocols, where $\kappa = 1$ and $N = 2, 3, \ldots, 10$.

In our N-party QKA protocol, in order to share a m-bit secret key, m single photons are used, and m decoy qubits are required in every transmission and N rounds of transmission are involved. Totally, $N(m + m)$ qubits should be required. The server announces mN bits to the parties to decode the shared key. The qubit efficiency is therefore $\eta = \frac{m}{(m+m)N + mN} = \frac{1}{3N}$. However, in order to be secure against collusive attacks, the proposed protocol needs the server's help. Meanwhile, the initial qubits preparation is delegated to the server, while participants just make measurement and do unitary operations on them, which makes our protocol more practical. Table 1 shows the efficiency comparison of our protocol and several existing secure MQKA protocols. As we can see in Fig. 1, if there are more than four parties involved in MQKA protocols, our protocol efficiency becomes much better than that of other protocols.

Conclusion

In conclusion, we proposed a multiparty quantum key agreement protocol which can resist Liu's collusion attack which is presented in the ref.[37]. To prevent the Liu's attack, the participants are restricted to preparing the initial quantum states in the proposed protocol. The stage of initial quantum states preparation is delegated to a server. It is proven that the protocol does not reveal the final shared key to any eavesdropper, including the server. And the participants involved in the protocol no longer need to prepare quantum states for message encoding, which makes the protocol more practical. And the main contribution of the paper is that we proposed a new model for quantum key agreement in client-server model, which protects the honest participants' fairness.

Received: 28 June 2019; Accepted: 1 November 2019;
Published online: 20 November 2019

References

1. Long, G.-L. & Liu, X.-S. Theoretically efficient high-capacity quantum-key-distribution scheme. *Phys. Rev. A* **65**, 032302 (2002).
2. Deng, F.-G., Long, G. L. & Liu, X.-S. Two-step quantum direct communication protocol using the einstein-podolsky-rosen pair block. *Phys. Rev. A* **68**, 042317 (2003).
3. Deng, F.-G. & Long, G. L. Secure direct communication with a quantum one-time pad. *Phys. Rev. A* **69**, 052319 (2004).
4. Zhang, W. *et al.* Quantum secure direct communication with quantum memory. *Phys. review letters* **118**, 220501 (2017).
5. Zhu, F., Zhang, W., Sheng, Y. & Huang, Y. Experimental long-distance quantum secure direct communication. *Sci. Bull.* **62**, 1519–1524 (2017).

6. Chen, S.-S., Zhou, L., Zhong, W. & Sheng, Y.-B. Three-step three-party quantum secure direct communication. *Science China Physics, Mech. & Astron.* **61**, 90312 (2018).
7. Wu, F. *et al*. High-capacity quantum secure direct communication with two-photon six-qubit hyperentangled states. *Sci. China Physics, Mech. & Astron.* **60**, 120313 (2017).
8. Sun, Z. W., Du, R. G. & Long, D. Y. Quantum secure direct communication with quantum identification. *Int. J. Quantum Inf.* **10**, 1250008 (2012).
9. Sun, Z. W., Du, R. G. & Long, D. Y. Quantum secure direct communication with two photon four-qubit cluster states. *Int. J. Theor. Phys.* **51**, 1946–1952 (2012).
10. Liu, W. J., Liu, C., Liu, Z. H., Liu, J. F. & Geng, H. T. Same initial states attack in yang et al.'s quantum private comparison protocol and the improvement. *Int. J. Theor. Phys.* **53**, 271–276 (2014).
11. Liu, W. J. *et al*. Improvement on "an efficient protocol for the quantum private comparison of equality with w state". *Int. J. Quantum Inf.* **12**, 1450001 (2014).
12. Wen Jie, L. & Chao, L. Han Wu, C., Zhi Qiang, L. & Zhi Hao, L. Cryptanalysis and improvement of quantum private comparison protocol based on bell entangled states. *Commun. Theor. Phys.* **62**, 210 (2014).
13. Sun, Z. W. & Long, D. Y. Quantum private comparison protocol based on cluster states. *Int. J. Theor. Phys.* **52**, 212–218 (2013).
14. Sun, Z. W., Yu, J. P., Wang, P., Xu, L. L. & Wu, C. H. Quantum private comparison with a malicious third party. *Quantum Inf. Process.* **14**, 2125–2133 (2015).
15. Li, Q., Chan, W. H. & Long, D. Y. Arbitrated quantum signature scheme using bell states. *Phys. Rev. A* **79**, 054307 (2009).
16. Zou, X. & Qiu, D. Security analysis and improvements of arbitrated quantum signature schemes. *Phys. Rev. A* **82**, 042325 (2010).
17. Sun, Z., Yu, J., Wang, P. & Xu, L. Symmetrically private information retrieval based on blind quantum computing. *Phys. Rev. A* **91**, 052303 (2015).
18. Zhou, N., Zeng, G. & Xiong, J. Quantum key agreement protocol. *Electron. Lett.* **40**, 1149–1150 (2004).
19. Tsai, C. & Hwang, C. W. On quantum key agreement protocol. *Tech. Report, C-S-I-E, NCKU, Taiwan, ROC* (2009).
20. Chong, S. K., Tsai, C. W. & Hwang, T. Improvement on "quantum key agreement protocol with maximally entangled states". *Int. J. Theor. Phys.* **50**, 1793–1802 (2011).
21. Chong, S. K. & Hwang, T. Quantum key agreement protocol based on bb84. *Opt. Commun.* **283**, 1192–1195 (2010).
22. Huang, W., Wen, Q. Y., Liu, B., Gao, F. & Sun, Y. Quantum key agreement with epr pairs and single-particle measurements. *Quantum Inf. Process.* **13**, 649–663 (2014).
23. Shen, D. S., Ma, W. P. & Wang, L.-L. Two-party quantum key agreement with four-qubit cluster states. *Quantum Inf. Process.* **13**, 2313–2324 (2014).
24. He, Y. F. & Ma, W. P. Quantum key agreement protocols with four-qubit cluster states. *Quantum Inf. Process.* **14**, 3483–3498 (2015).
25. Shi, R. H. & Zhong, H. Multi-party quantum key agreement with bell states and bell measurements. *Quantum Inf. Process.* **12**, 921–932 (2013).
26. Liu, B., Gao, F., Huang, W. & Wen, Q. Y. Multiparty quantum key agreement with single particles. *Quantum Inf. Process.* **12**, 1797–1805 (2013).
27. Sun, Z., Zhang, C., Wang, B., Li, Q. & Long, D. Improvements on "multiparty quantum key agreement with single particles". *Quantum Inf. Process.* **12**, 3411–3420 (2013).
28. Yin, X. R., Ma, W.-P. & Liu, W. Y. Three-party quantum key agreement with two-photon entanglement. *Int. J. Theor. Phys.* **52**, 3915–3921 (2013).
29. Yin, X. R., Ma, W. P., Shen, D. S. & Wang, L. L. Three-party quantum key agreement with bell states. *Acta Phys. Sinica* **62**, 170304 (2013).
30. Shukla, C., Alam, N. & Pathak, A. Protocols of quantum key agreement solely using bell states and bell measurement. *Quantum Inf. Process.* **13**, 2391–2405 (2014).
31. Zhu, Z. C., Hu, A. Q. & Fu, A. M. Improving the security of protocols of quantum key agreement solely using bell states and bell measurement. *Quantum Inf. Process.* **14**, 4245–4254 (2015).
32. Sun, Z., Yu, J. & Wang, P. Efficient multi-party quantum key agreement by cluster states. *Quantum Inf. Process.* **15**, 373–384 (2016).
33. Sun, Z. *et al*. Multi-party quantum key agreement by an entangled six-qubit state. *Int. J. Theor. Phys.* **55**, 1920–1929 (2016).
34. Huang, W., Wen, Q. Y., Liu, B., Su, Q. & Gao, F. Cryptanalysis of a multi-party quantum key agreement protocol with single particles. *Quantum Inf. Process.* **13**, 1651–1657 (2014).
35. Cao, H. & Ma, W. Multiparty quantum key agreement based on quantum search algorithm. *Sci. reports* **7**, 45046 (2017).
36. Liu, W. J., Chen, Z. Y., Ji, S., Wang, H. B. & Zhang, J. Multi-party semi-quantum key agreement with delegating quantum computation. *Int. J. Theor. Phys.* **56**, 3164–3174 (2017).
37. Liu, B., Xiao, D., Jia, H. Y. & Liu, R. Z. Collusive attacks to "circle-type" multi-party quantum key agreement protocols. *Quantum Inf. Process.* **15**, 2113–2124 (2016).
38. Wang, P., Sun, Z. & Sun, X. Multi-party quantum key agreement protocol secure against collusion attacks. *Quantum Inf. Process.* **16**, 170 (2017).
39. Huang, W. *et al*. Improved multiparty quantum key agreement in travelling mode. *Science China Physics, Mech. & Astron.* **59**, 120311 (2016).
40. Huang, W. *et al*. Efficient multiparty quantum key agreement with collective detection. *Sci. reports* **7**, 15264 (2017).
41. Sun, Z., Wu, C., Zheng, S. & Zhang, C. Efficient multiparty quantum key agreement with a single d -level quantum system secure against collusive attack. *IEEE Access* **7**, 102377–102385, https://doi.org/10.1109/ACCESS.2019.2931612 (2019).
42. Li, X. H., Deng, F. & Zhou, H. Y. Improving the security of secure direct communication based on the secret transmitting order of particles. *Phys. Rev. A* **74**, 054302 (2006).

Acknowledgements

This work is funded by the National Natural Science Foundation of China (Nos 61602316, 61902132, 11647140, 61902081), the Science and Technology Innovation Projects of Shenzhen (No. JCYJ20170818140234295), the Natural Science Foundation of Guangdong Province, China (Nos 2018A030310147, 2018A030313974, 2015A030313630),and the Science and Technology Planning Project of Guangdong (No. 2018A050506087). Cai Zhang is sponsored by the State Scholarship Fund of the China Scholarship Council.

Author contributions

Study conception, design, and writing of the manuscript: Z.S. and C.Z. Analysis and discussion: R.C. and C.W. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to C.Z.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.