Journal of Computers Vol. 28, No. 6, 2017, pp. 13-25
doi:10.3966/199115992017122806002

Quantum Public-key Cryptosystem without Exchanging Qubits between Any Two Users

Xiaoyu Li[1] and Yun Shang[2, 3]

[1] School of Information Engineering, Zhengzhou University,
Zhengzhou City 450001, P. R. China
iexyli@zzu.edu.cn

[2] Institute of Mathematics, Academy of Mathematics and System Science, CAS
Beijing City 100190, P. R. China

[3] NCMIS, Academy of Mathematics and System Science, CAS
Beijing City, P. R. China
shangyun602@163.com

Received 23 September 2015; Revised 4 January 2016; Accepted 9 May 2016

Abstract. A quantum public-key cryptosystem using entangled states is present in this paper. Every user shares a set of entangled quantum systems with the key management center (KMC) as the (private key, public key) pair. When a user wants to send encrypted message to another user, he or she asks KMC for the latter's public key and produces a string to encrypt his or her pain text. Then the latter can get the same string using the correlations between the subsystems of the entangled quantum systems and recover the plain text while no third one can make it. So users can achieve secret communication and digital signature by the help of KMC. The principles of quantum physics guarantee the unconditional security of the cryptosystem. Any two users needn't performing exchanging quantum systems with each other. So it is easier to carry out and more robust in practice.

Keywords: digital signature, entangled states, quantum cryptography, quantum public-key cryptosystem

1 Introduction

To sends a secret message through an insecure channel people need cryptography. The message needs to be transmitted is called the plain text. It is integrated with a string (called the key) to produce the encrypted message (called the cipher text). Only the cipher text is transmitted through the insure channel so that everyone can get it. But no one can recover the plain text without the key. So users who share the key can achieve secret communications. As a result, secure key distribution becomes the precondition for secure communication. But it's a very difficult problem. In classical cryptography nearly no unconditionally secure key distribution protocols exist. Most practical classical key distribution keys are based on computing complexity which depend the hardness of NP problems.

Quantum key distribution (QKD) protocol is a good solution to this problem. QKD protocols can show unconditionally secure key distribution with an open quantum channel and an insecure but authenticated classical channel. Bennett and Brassard proposed the first quantum key distribution protocol in 1984 (so called BB84 protocol) [1]. Since then people have developed many QKD protocols, such as the EPR protocol [2], B92 [3], the protocol of Lo-Chau [4], and so on [5-13]. At the same time experimental work for OKD has also been implemented. Bennett et al first completed BB84 protocol in laboratory in 1992 [14]. Now QKD in optical fiber has been achieved beyond 150 km [15]. QKD protocol in free space has been implemented over a distance of 1 km [16].

All the QKD protocols mentioned above are symmetrical key protocols, that is to say, encryption and

decryption use the same key. All symmetrical key cryptosystems are faced with a difficult problem: how to distribute and manage keys if there are many users in a cryptosystem? If N users want to communicate with each other, one user must share a key with any other user. So every user must keep N-1 keys secret to exchange information with the other N-1 users respectively. Moreover, N(N-1)/2 key distribution processes should be implemented before the cryptosystem begins to work. It's very hard to guarantee it when N is a large number. On the other hand in practice maybe the users don't trust each other, which make key distribution impossible from the beginning. As known in classical cryptography a solution to overcome such difficulties is public-key cryptosystem, such as RSA algorithm [17], etc. Every user has a (public key, private key) pair. The public key is used to decrypt the information encrypted by the private key while the private key is used to decrypt the information encrypted by the public key. Moreover the public key and the private key can't be deduced from one to the other. All users' public keys are kept open by a key management center (KMC) while every user keeps his or her private key secret so that no other people can get. When a user, for example, Bob, wants to send a secret message to another user Alice, he first asks KMC for Alice's public key. Then Bob encrypts the message using the public key and sends the encrypted message (the cipher text) to Alice. When Alice receives the cipher text, she decrypts the cipher text by her private key and gets the plain text. Any eavesdropper can catch the cipher text. But it's impossible for him to recover the plain text without Alice's private key. Classical public-key cryptosystem has been widely used in modern society, such as commercial affairs, military affairs, network communications et al. As known NP problems are some difficult problems which have no polynomial time algorithm until now. Most of the classical public-key algorithms are based on NP-hard problems which are the most difficult problems of NP problems [18-19]. But as known in 1994 Peter Shor proved that RSA algorithm can be cracked in polynomial time on future quantum computer [20]. It means that the classical public-key cryptosystems based on RSA algorithm will collapse inevitably faced with a quantum computer. So do several most popular classical public-key cryptosystem. Quantum public-key technology may be a good solution to resist such danger. It originated form a paper of Gottesman which present a quantum one-way function to design quantum digital signature protocol [21]. This idea may be used in a public-key system. In 2008 Nikolopoulos presented the first unconditionally secure quantum public-key protocol [22] which based on the single-particle rotation of unknown quantum states. Since then people have developed several public-key protocols one after another [23-26].

Until now most of the quantum public-key protocols require that users accomplish communication by exchange qubits, which means that there must be a quantum channel between any two users. It may be difficult to guarantee in reality. Moreover quantum channel is more fragile, which reduces the robustness of quantum public-key cryptosystems. In this paper we provide a quantum public-key cryptosystem without quantum channel between any two users which is based on the non-locality of entangled states. It is based on the non-local correlation in an entangled quantum system. With the help of the key management center, N users can communicate with each other securely. Moreover digital signature of the message can be fulfilled naturally by the public-key cryptosystem. Any two users don't need to exchange qubits with each other. So it's easier to carry out and more robust in practice.

2 Basic Idea

As known a two-state quantum system is called a qubit. A qubit may be in any state in the set {|0>, |1>, |+>, |->} in which

$$|+>=\frac{1}{\sqrt{2}}(|0>+|1>), \quad |->=\frac{1}{\sqrt{2}}(|0>-|1>). \tag{1}$$

Obviously the four states aren't orthogonal to each other, or in other words, there are no ways to determine in which state a qubit is from the four possible states with certainty. On the other hand the four states form two complete orthogonal basic vector sets respectively

$$B_{01}=\{|0>,|1>\} \qquad B_{+-}=\{|+>,|->\} \tag{2}$$

in which people can measure a qubit. There are four maximumly entangled states in a two-qubit system

$$
\begin{aligned}
|\Phi^+> &= \frac{1}{\sqrt{2}}(|00> + |11>) \\
|\Phi^-> &= \frac{1}{\sqrt{2}}(|00> - |11>) \\
|\Psi^+> &= \frac{1}{\sqrt{2}}(|01> + |10>) \\
|\Psi^-> &= \frac{1}{\sqrt{2}}(|01> - |10>)
\end{aligned}
\tag{3}
$$

which are also called the Bell states. A two-qubit system in one of the Bell states is often called an EPR pair. Notice that the Bell state $|\Phi^+>$ can be rewritten as

$$|\Phi^+> = \frac{1}{\sqrt{2}}(|++> + |--|) \tag{4}$$

From (3) and (4) we can get an important conclusion. If two people measure the two qubits respectively in basis B_{01}, they will get the same measurement results. At the same time if they measure the two qubits respectively in basis B_{+-}, they will get the same results, too. Such non-local correlations are the fundamental property of entangled states. This fact is the main idea on which our public-key cryptosystem is based.

Now let's introduce a public-key cryptosystem which includes a key management center (KMC) and N users. Every user, such as Alice, share M EPR pairs with KMC in the state

$$|\Phi^+>_{12} = \frac{1}{\sqrt{2}}(|0>_1|0>_2 + |1>_1|1>_2) \tag{5}$$

in which qubit 1 is hold by Alice while qubit 2 is hold by KMC. So KMC has an M-qubit sequence denoted as Q^K while Alice has an M-qubit sequence denoted as Q^A. Q^K and Q^A are just Alice's public key and private key. To communicate with each other, all users first agree to the following key rule.

Key rule.

$$|0> \to 0, \quad |1> \to 1, \quad |+> \to 0, \quad |-> \to 1 \tag{6}$$

In this quantum public-key cryptosystem, there are no quantum channels needed connecting any two users. What we need is a communal public classical channel through which any user can exchange classical information with another one. On the other hand there is a quantum channel through every user can exchange qubits with KMC to get another user's public key. Both the classical channel and the quantum channel are insecure to which everyone can listen to. But the classical channel is authenticated so that one user can affirm the identity of the other one communicating with him.

If one user, such as Bob, wants to sends an n-bit string denoted as P to Alice, he first asks KMC for Alice's public key Q^K. Then Alice measures all qubits in Q^A in basis B_{01} or B_{+-} at random. Now the basis sequence can be denoted as

$$B = (B_1 B_2 ... B_M), \quad B_i \in \{B_{01}, B_{+-}\} \tag{7}$$

At the same time Alice records her measurement results according to the Key Rule. So she gets an M-bit string denoted as S. Next Alice sends B to Bob through the classical channel. When Bob receives B, he measures Q^K according to B and records his measurement result according to the Key Rule. So Bob also gets an M-bit string denoted as S'. Then Alice chooses t=M-n bits from S at random and declares them. Bob chooses the corresponding t bits from S' and declares them, too. If there are too many disagreements, they abandon the communications process and turn to the beginning. Or they can assure that the left n-bit substring of S (denoted as K) and the left n-bit substring of S' (denoted as K') must be exactly equal, or in other words, K=K'. The process in which Bob and Alice get the same string K' or K by the help of KMC can be described in Fig. 1.

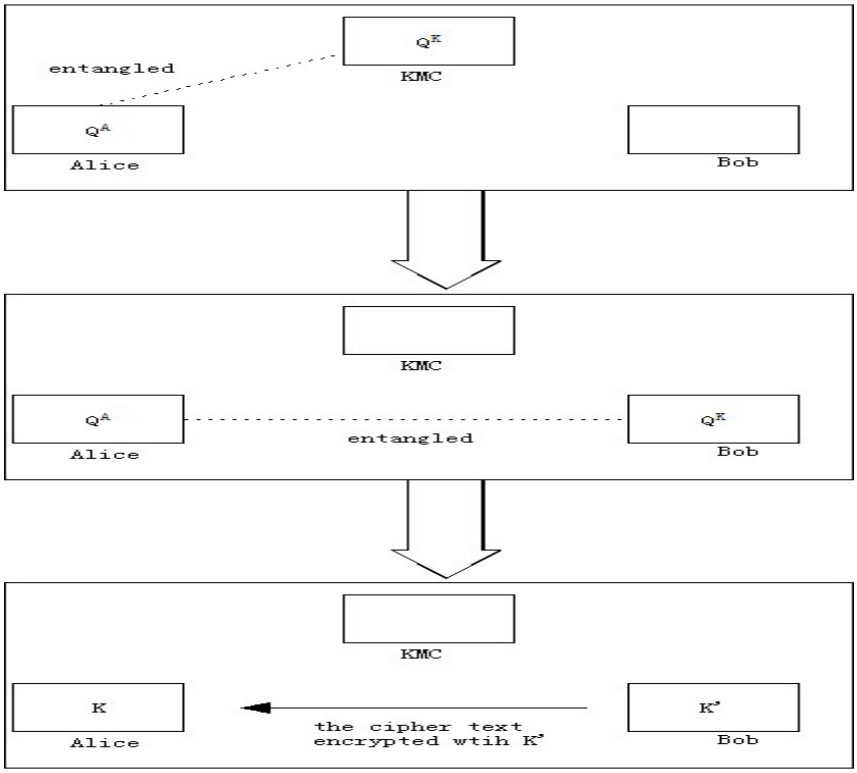

Fig. 1. Process of establishing the key

Now Bob performs an XOR operation on P and K' to get a string EP

$$EP = P \oplus K' . \tag{8}$$

EP is just the cipher text of P. Then Bob sends EP to Alice through the classical channel. When Alice receives EP, she performs an XOR operation on EP and K to get a string P'.

$$P' = EP \oplus K \tag{9}$$

Since K=K', it's obvious that we have P'=P. So Alice gets the message which Bob wants to send her. In section 4 we will prove that no one including KMC except Alice and Bob can get the message. So users can achieve secret communications in the public-key cryptosystem.

It must be pointed out that KMC should keep the M-qubit sequence Q^A for some time until a user asks for it. So KMC should have a quantum storage which is difficult but not impossible. People can make it using today's technology now although the storage time and the storage capacity need to be improved. There is another problem which must be solved. After a communication process the EPR pairs are consumed. So both Alice's public key and private key no longer exist. They can be used for only one time! In fact maybe many users want to send message to Alice or one user wants to sends message to Alice for many times. So Alice must share many (public key, private key) pairs with KMC. Each (public key, private key) pair is given a unique id number.

So we can design a complete quantum public-key cryptosystem based on the idea above.

3 Quantum Public-key Cryptosystem without Exchanging Qubits

Now we give the quantum public-key cryptosystem as follows.

3.1 Building the Public-key Cryptosystem

There are N users and a key management center (KMC) in the public-key cryptosystem. Any two users can exchange classical information through an authenticated public classical channel. The classical

channel is public so that everyone can listen to it and catch the classical information transmitted through it. But a user can affirm the identity of the other at the end of the channel, or in other words, no one can impersonate other to send fake information through the classical channel. As known such an authenticated public classical channel is necessary to nearly all quantum cryptographic protocols including famous BB84 protocol et al. On the other hand every user can exchange qubits with KMC through an insecure quantum channel. But two users needn't to exchange qubits so that no quantum channels are needed to connect them.

Every user creates $L \times M$ EPR pairs in the state $|\Phi^+>$ in which every M EPR pairs are given a unique id numbers as a (public key, private key) pair. Then she shares all the EPR pairs with KMC in which to each EPR pair the first qubit is hold by the user himself while the second qubit is hold by KMC. So a user's public keys set can be denoted as

$$K_{PK} = \{ (i, Q_i^K), \quad i = 1, 2, ..., L \} \tag{10}$$

in which Q_i^K is a M-qubit sequence with the id number i. the user's private key set can be denoted as

$$K_{PA} = \{ (i, Q_i^A), \quad i = 1, 2, ..., L \} \tag{11}$$

in which Q_i^K is the corresponding M-qubit sequence with the id number i. All users' public keys are kept by KMC and open to everyone who wants get them. But a user must keep his private keys absolutely secret by himself so that no other one including KMC can get.

3.2 Process of Communication

If a user, such as Bob, wants to send an *n*-bit string *P* to another user Alice. They do according to the following steps.

Step 1. Bob asks KMC for one of Alice's public keys.

Step 2. KMC chooses one public keys (j, Q_j^K) from Alice's public key set K_{PK} at random and sends it to Bob.

Step 3. When Bob receives (j, Q_j^K), he sends the id number *i* to Alice.

Step 4. When Alice receives the id number *i*, she measure the corresponding private key (j, Q_j^A) in basis B_{01} or B_{+-} at random and records her measurement basis sequence as *B*. Then Alice records her measurement results according to Key Rule. Finally she gets an *M*-bit string *S*'.

Step 5. Alice send her basis sequence *B* to Bob.

Step 6. When Bob receives *B*, he measures (j, Q_j^K) according to *B* and records his measurement basis according to Key Rule. Finally Bob also gets an *M*-bit string *S'*.

Step 7 (error-checking). Alice chooses *t* bits (*t*=*M*-*n*) at random from *S* and Bob chooses the corresponding bits from *S*' respectively. Then they compare them. If there are too many disagreements, they abandon the communication process and turn back to step 1. Or Alice and Bob keep the left *n*-bit string *K* and *K*' respectively and continue into next step.

Step 8. Bob performs an XOR operation on *P* and *K*' to get the cipher text *EP*. Then Bob sends *EP* to Alice.

Step 9. When Alice receives *EP*, she performs an XOR operation on *EP* and *K* to get a string *P*'.

Obviously they know *P*'=*P*. So Alice has obtained the message that Bob sends her.

If Alice wants to send a secret message to Bob, they need only exchange the roles in the process above. So any two users can achieve secret communications by this public-key cryptosystem.

3.3 Digital Signature

If Bob sends a secret message *P* to Alice, he can sign the message to prove his identity to Alice. What Bob needs to do is to attach a classical message (the signed message) with the original message that he wants send to Alice. To produce the signed message, Bob performs the following steps.

Step 1. Bob produces an m-bit abstract *PA* from *P* which he wants to send Alice by a hash algorithm, such as SHA-1 algorithm.

Step 2. Bob chooses one of his private keys at random, such as (i, R_i^A). Then he performs measurement the first m qubits of (i, R_i^A) in basis $\{|0>, |1>\}$ and records his results according to Key Rule. So Bob gets a string *PK*.
Step 3. Bob performs XOR operation between *PA* and *PK*. Finally he gets an *m*-bit string *PS* which is just the signed message.
Step 4. Bob attaches *PS* and the id number *i* with *P*. So he gets a string *PX* which is the plain text to be submitted to Alice.
Step 5. Bob sends *PA* to KMC through the classical channel and declares that it's the abstract of the message which he will send Alice.

Notice that now the length of *PX* should be *n*. So the length of the original message *P* added with the length of *k* should be *n-m*. If *P* can't satisfy it, we can always make it by dividing it into several parts or adding supplementary bits.

Then Bob and Alice can finish the communication as the steps in section III.

After Alice gets the plain text *PX*, she can extract the original message *P*, the signed message *PS* and the id number *i*. To verify the signature, she does the following steps.
Step 1. Alice asks KMC for Bob's no. *i* public key (i, R_i^K).

Step2. Alice measure (i, R_i^K) in basis $\{|0>, |1>\}$. Then she takes the first m measurement results and records according to Key Rule. Finally she gets an *m*-string *PK*' which is just equal to *PK*.
Step 4. Alice performs XOR operation between *PK*' and *PS*. So she gets an *m*-bit string *PA*'.
Step 5. Alice produces the abstract *PA* of *P* using SHA-1 algorithm just as Bob does.
Step 6. Alice compare *PA*' and *PA*. If they are identical, the verification succeeds. Alice can be sure that the message is really from Bob.

On the other hand if Bob denies that *P'* is sent from him to Alice, Alice can easily prove that Bob is lying. What she needs to do is to send *P'* to KMC. After receiving *P'*, KMC produces the abstract denoted as *KPA* from it by SHA-1 algorithm. If *KPA* is exactly equal to *PA* which Bob has sent him, KMC can affirm that Bob has really sent *P* to Alice. So Bob can't deny his signature.

3.4 Experimental Program

First every user builds his or her (public key, secret key) set with KMC. To make it, KMC creates EPR pairs in the state $|\Phi^+>$ by many methods [27-29] and sends them to the user. This can be easily achieved by today's technology. But there is still a difficulty left. KMC must keep all users' (public key, secret key) sets for a relative long time until they are used. Or in other words, KMC needs quantum memory which can maintain quantum coherence for a relative long time. Recently Sprague et al provided a broadband quantum memory in a hollow- core photonic-crystal fibre [30] which is just what this quantum public-key cryptosystem needs. So in this public-key cryptosystem KMC creates photon pair in two-photon entangled state as the EPR pair needed in the cryptosystem. Then KMC saves all users' (public key, secret key) sets by the technology [30].

Second when a user Alice wants to send a secret message to another one Bob, he asks KMC for the counterpart's public key. KMC can export the photons by the method that Sprague issued, which can be accomplished with a probability of success near to 100%. Then KMC sends the photons to Alice. As known entanglement between these photons and those photons hold by Bob can preserve very well because that photons have little interactions with the environment so that quantum coherence can preserve in transmission.

Third in the rest steps of the process of the communication, all that Alice and Bob need to do are performing local measurement on the photons and exchanging classical information. It's known that there are no difficulties at all.

So users can fulfill secure communications in this public-key cryptosystem by today's technology.

4 Security of the Public-Key Cryptosystem

This public-key cryptosystem is secure. If one user Bob sends a message to another user Alice, no one including KMC except Alice can get the message. On the other hand digital signature is also secure. No

one can fake Bob's signature while Bob can't deny his signature. We prove it as follows. First we assume that an eavesdropper, such as Eve, wants to get the message transmitted from Bob to Alice.

4.1 Impossibility for Eve to Get the Message

To get the message sent from Bob to Alice, Eve may listen to both the classical channel through which Bob communicates with Alice and the quantum channel through which Bob gets Alice's public key from KMC. She can get the cipher text EP sent form Bob to Alice in step 8. On the other hand, she also knows that the cipher text is produced by no. j public key with the plain text. But she can't get the plain text *P* at all because *EP* is produced by *P* XOR *K*'. As known *K*' is determined by the results of Bob's measurements on (j,Q_j^K) which are kept secret by Bob. Eve can't get Bob's measurements results. So she can't get *K*'. On the other hand the private (j,Q_j^A) is kept absolutely secret by Alice so as to no one can get it. Eve can never get the *K* by performing measurements on the qubits just as Alice does. Now we have the following preconditions.

(1) *EP*=*P* XOR *K*';

(2) Eve has only *EP*;

(3) *K*' is a random string produced from Bob's measurement result about which Eve knows nothing.

From the preconditions above we can easily come to such conclusion. Whatever Eve does, the probability to get the plain text *P* for her is no more than just guessing it. Since *P* is a n-bit string, the probability for Eve to get *P* is no more than

$$P_{error}=\left(\frac{1}{2}\right)^n. \tag{12}$$

If n=1000, we have

$$P_{error}=\left(\frac{1}{2}\right)^{1000}\approx 10^{-300}. \tag{13}$$

It's a number too small to imagine. So Eve's attack is sure to fail.

Let's consider another strategy of attack. When KMC sends Alice's public key (j,Q_j^K) to Bob, Eve may catches it and tries to get the key. We can prove that Eve can't succeed. First Eve can't measure (j,Q_j^K) because she doesn't know the correct basis sequence B which is determined after Alice receives Bob's notification in step 3. Since Bob doesn't receive (j,Q_j^K) because Eve has intercepted it, he won't send any message to Alice at all! If Eve measures (j,Q_j^K), she can just get some random measurement results. It's of no use. On the other hand Eve may think that she can make a copy of (j,Q_j^K). Then she sends (j,Q_j^K) to Bob and listens to the communicated information between Alice and Bob so that she may decrypt the cipher text just as Alice. Such strategy of attack is infeasible at all. Quantum no-cloning theorem forbids anyone to clone an unknown state. So it's impossible for Eve to make a copy of (j,Q_j^K), not mention to getting the plain text P.

Finally Eve can take the strategy of entanglement attack. When KMC sends Alice's public key (j,Q_j^K) to Bob, Eve catches it. Then she creates M auxiliary qubit and performs CNOT operation on each auxiliary qubit (denoted as qubit E) and a qubit in (j,Q_j^K) in which the former is the target qubit and the latter is the control qubit. So the state of the whole three-qubit system composed of an EPR pair and an auxiliary qubit turns into

$$|S>=\frac{1}{\sqrt{2}}(|0>_1|0>_2|0>_E+|1>_1|1>_2|1>_E). \tag{13}$$

In step 4 ~ step 7 Alice and Bob respectively measure qubit 1 and qubit 2 according to *B*. So Eve measure qubit E, too. If the measure basis is B_{01}, all the three people will get the same result, or in other

words, Eve will get the same result as Alice and Bob, which may make her get the bit of *K*'. Moreover Alice and Eve can't find Eve's existing. It seems like a serious danger to the public-key cryptosystem. But if the measurement basis is B_{+-} the result is complete different. Equation (13) can be rewritten as

$$\begin{aligned} |S>=\frac{1}{2\sqrt{2}}[(|+>_1|+>_2+|+>_1|->_2+|->_1|+>_2 \\ +|->_1|->_2)|0>_E+(|+>_1|+>_2-|+>_1|->_2 \\ +|->_1|+>_2+|->_1|->_2)|1>_E] \end{aligned} \tag{14}$$

It can also be rewritten as

$$|S>=\frac{1}{\sqrt{2}}[(|+>_1|+>_2+|->_1|->_2)|+>_E+(|+>_1|->_2+|->_1|+>_2)|->_E] \tag{15}$$

So the probability that Alice and Bob get the same measurement results is 1/2 no matter whatever basis Eve measures qubit E in. Since the measurement basis is B_{01} or B_{+-} at random, so the average probability that Alice and Bob get the same measurement results is 3/4. In step 7 Alice and Bob compare t bit from S and S'. So the probability that all these t bits are same is

$$P_{error}=(3/4)^t. \tag{16}$$

If t=200, the probability is

$$P_{error}=(3/4)^{200}\approx 10^{-25}. \tag{17}$$

This is just the probability that Eve escaping from be found by Alice and Bob. It's too small a number to imagine. In fact we can be sure that Alice and Bob will find Eve's existing in step 7 and abandon the communications process, that is to say, Eve's entanglement attack fails.

4.2 Impossibility for KMC to Get the Message

In this public-key cryptosystem KMC can't get the message sent from Bob to Alice, either. We prove it as follows.

First KMC will be on the same boat as Eve when Bob sends the cipher text *EP* to Alice. KMC can get *EP*, too. But without *K*' it's impossible for it to recover the plain text *P* from *EP* just as Eve.

Second since KMC keeps Alice's public keys, it may try to get something about the key to help itself in getting the plain text. But KMC can't measure Alice's public key (j,Q_j^K) because doing such measurements not only doesn't help it to get *K*' but also makes it to be found by Alice and Bob. As know one qubit in (j,Q_j^K) and the corresponding qubit in (j,Q_j^A) form an EPR pair, which is just the reason that Alice and Bob get the same measurement results. If KMC measures (j,Q_j^K), the two qubit will collapse into eigenstates and lose the correlations of measurement results between them. So in step 7 to a bit from S and the corresponding bit from S', the probability that they are equal is at most 1/2. Since Alice and Bob compare t bits in step 7, the probability to all the t bits is

$$P_{error}=\left(\frac{1}{2}\right)^t. \tag{18}$$

If t=200,

$$P_{error}=\left(\frac{1}{2}\right)^{200}\approx 10^{-60}. \tag{19}$$

So Alice and Bob are sure to find something wrong and abandon the process of communication, that is to say, KMC can't get the message sent from Bob to Alice.

Third KMC may also take entanglement attack, too. It's easy to find that it can do nothing more than Eve at all. So such attack form KMC is determined to fail just as such attack from Eve.

Finally we consider a complex attack. When Bob asks for Alice's public key (j,Q_j^K), KMC give him a fake key denoted as (j,FQ_j^K) which is a part of the EPR pair sequence $[(j,FQ_j^K),(j,FQ_j^A)]$. Then KMC measures (j,FQ_j^A) and (j,Q_j^K) respectively in basis B in order to get message P. So KMC's measurement results on (j,FQ_j^A) are same as Bob's measurement result on (j,FQ_j^K) while KMC's measurement results on (j,Q_j^K) are same as Alice's measurement result on (j,Q_j^A). But this fact is of no help for KMC. There are no correlations between Alice's measurement results (j,Q_j^A) and Bob's measurement result on (j,FQ_j^K) because (j,FQ_j^K) aren't entangled with (j,Q_j^A). So in step 7 to a bit from *S* and the corresponding bit from *S*', the probability that they are equal is 1/2, too, that is to say, the probability to all the t bits is

$$P_{error}=\left(\frac{1}{2}\right)^t. \tag{19}$$

If t=200,

$$P_{error}=\left(\frac{1}{2}\right)^{200}\approx 10^{-60}. \tag{20}$$

So suck attack fails, too.

4.3 Impossibility for Eavesdroppers to Distort the Message

Eve may catch the cipher text *EP* sent from Bob to Alice and sends a fake message denoted as F*P* to Alice. We prove that it's impossible.

The cipher text *EP* is transmitted through the public classical channel. It's easy for Eve to catch *EP* and keep it. But Eve has no *K*' to encode *FP*. No matter what Eve does to treat *FP*, the probability that Alice just get *FP* after decoding is no more than the probability that Alice guesses every bit of *K*' correctly. So the probability that Eve make Alice to accept a fake message is

$$P_{error}=\left(\frac{1}{2}\right)^n. \tag{21}$$

If n=1000,

$$P_{error}=\left(\frac{1}{2}\right)^{1000}\approx 10^{-300}. \tag{22}$$

It's an extreme small probability so that we can say that Eve can't succeed.

On the other hand Eve may catch the public key (j,Q_j^K) when it's sent from KMC to Bob. But she still has no way in making Alice receive a fake message. Moreover Eve will be found as long as she performs any operation on (j,Q_j^K) because Alice and Bob do error-checking in step 4-7 by comparing their measurement results. It's easy to prove that the probability Eve escaping from being found is equal to equation (19) in subsection 4.1, or in other words, about 10^{-60}.

So we can conclude that Eve can't have Alice to accept a distort message.

4.4 Impossibility for KMC to Distort the Message

Although KMC keeps Alice's public keys, it can't make Alice to accept a distort message, either. If Bob asks KMC for Alice's public key (j,Q_j^K), how can KMC do? First KMC can't measure (j,Q_j^K) because Alice and Bob will find it and abandon the process of communication, which has been proved in subsection 4.3. Second quantum no-cloning theorem forbids it to produce a copy of (j,Q_j^K). So what

KMC can do is no more than what Eve can do. We have proved that Eve can't make Alice to accept a distort message. So KMC can't make it likewise.

4.5 Security of Digital Signature

We can prove that in this cryptosystem digital signature is secure. No one can counterfeit Bob's digital signature and Bob can't deny his signature. To produce the signed message, Bob chooses the first m qubits from his private (i, R_i^A) and measures them in basis $\{|0>, |1>\}$. So he will get a random binary string *PK*. The signed message PS is produced by $PK \otimes PA$ in which *PA* is the abstract of the plain text. When Alice receive the cipher text and decrypted it, she will extract PS and the id number *i*. Then she ask KMC for Bob' public key (i, R_i^K) and choose the first m qubits. By measuring these qubits Alice gets *PK*'. The property of entangled states guarantees that *PK*= *PK*'. So when Alice performs $PK' \otimes PA$ to get *PS*', she can be sure that *PS*'= *PS*. The digital signature is verified. It's easy to find that *PK*= *PK*' is necessary to produce the correct signed message. But no one except Bob holds Bob's private key (i, R_i^A). This means any eavesdropper, such as Eve, can't get the same measurement results as that of Alice's measurement on (i, R_i^K). So it's impossible for Eve to get *PK*. Without *PK* the probability that Eve just produce the correct sign message PS is no more than

$$P_{error} = \left(\frac{1}{2}\right)^m . \tag{23}$$

If m=100,

$$P_{error} = \left(\frac{1}{2}\right)^{100} \approx 10^{-30} . \tag{24}$$

So we can say that no one can counterfeit Bob's signature. At the same time Bob can't deny his signature if the signed message is verified by Alice.

4.6 Security against Forward Search Attack

Forward search attack is a powerful strategy of attack against classical public-key cryptosystem. Eve can use Alice's public key to encrypt many plain texts and save the (plain text, cipher text) pairs in her database. Next Eve listens to the channel and gets all the cipher texts sent to Alice. Then Eve queries the cipher text in her database. If she does find the (plain text, cipher text) pair, she is sure that the plain text is just the secret message which Bob wants to send to Alice. But forward search attack is pointless in this public-key cryptosystem. Alice has many (public key, private key) pairs in which one pair can be used for only one time. If Eve uses one of Alice's public keys (j, Q_j^K) to produce the cipher text *EP*, both (j, Q_j^K) and (j, Q_j^A) have been consumed. If a user wants to send a message to Alice, he must choose another public key $(j', Q_{j'}^K)$ to encrypt the plain text, that is to say, Eve can't query it in her database at all. So forward search attack is due to fail.

4.7 Security against Resend Attack

Let's discuss another attack, resend attack. In classical public-key cryptosystem Eve can catch the cipher text and make a copy of it. Then Eve resends the cipher text again to Alice. Alice can decrypt it successfully without feeling anything wrong. So Eve succeeds in making Alice to receive a repeated message although she even knows nothing about the message. But our quantum public-key cryptosystem is immune to such attack. When Alice receives a repeated cipher text from Eve, she can't use the correct private key to decrypt it because the key has been consumed after the first communication process. Without the correct private key, Alice can never recover the plain text. What Alice can receive is not a repeated message but a random string, or in other words, Eve's resend attack can never succeed.

4.8 Security against Chosen Plain Text Attack

In classical public-key cryptosystem chosen plain text attack is a very serious threat. The public key is open to everyone. So Eve can use it to encrypt any plain text to get the corresponding cipher text. Then Eve may analyze all the (plain text, cipher text) pairs in order to find some principles which may help her to get the private key. If the number of the available (plain text, cipher text) pairs is large enough, the probability that Eve succeeds in defeating the cryptosystem can't be ignored. It's easy to prove that in our public-key cryptosystem chosen plain text isn't a danger at all. Since one public key can only be used for one time, any two (plain text, cipher text) pairs are produced by different public keys. There are no principles between the (plain text, cipher text) pairs at all! Eve can't get any valuable information. So chosen plain text attack is invalid.

5 Feasibility Analysis of the Public-key Cryptosystem

First an advantage of our public-key cryptosystem is that what the users need to do are performing the single particle measurement on a qubit and transmitting qubits through a quantum channel, which have been realized in laboratory for many years. So there are no fundamental technical difficulties which prevent this quantum public-key cryptosystem from coming into practice.

Second another significant advantage is that two user needn't exchange quantum information in this cryptosystem. So there are no quantum channels needed between them, which greatly depresses the resource requirements. So our cryptosystem is easier to carry out. Moreover no quantum channels needed means that it doesn't need to face a series of technical problem, such as control of quantum system, quantum decoherence, quantum noise et al. So it's more robust.

Third as known the power of quantum cryptography comes from the quantum coherence of quantum system.

But in practice quantum systems are sure to undergo decoherence over time which makes them to lose quantum coherence inevitably. It's the most serious threat to quantum cryptographic protocol. In traditional quantum cryptographic protocols people can avoid this difficulty by completing the protocol before decoherence occurs. But in public-key cryptosystem, KMC must keep all users' public keys which are quantum system for a relative long time until a user asks for them. So decoherence is unavoidable. To solve this problem, we can use the quantum system which has bigger time length of decoherence, such as photon in single-mode fiber et al. On the other hand users can update their public keys periodically before they become invalid. Taking such methods, it's possible for our cryptosystem to work well for a long time.

6 Discussion and Conclusion

A defect of our public-key cryptosystem is that a public key can be used for only one time. This puts restriction on user's communications. Once a user' public keys are used up, no one can sends message to him again. Such limit can be removed by developing cryptosystem in which the public key is reusable. It will be discussed in future work.

In this paper we provide a quantum public-key cryptosystem using entangled states. Using EPR pairs as (public key, private key) pair, N users can achieve secret communications by the help of a key manage center (KMC). The principles of quantum mechanics guarantee that our cryptosystem is unconditionally secure. No one including KMC can get the secret message. The message can be signed so that the receiver can verify the truth of the message. No exchanging qubits are needed between any two users. So our quantum public-key cryptosystem is easier to carry out and more robust in practice.

Acknowledgements

This work is supported by Natural Science Foundation of China (Grants 61472412), Natural Science Foundation of the Education Department of Henan Province of China (Grants 14A520012) and Natural Science Basic Research Plan in Shannxi Province of China (No. 2014JM2-6103). The authors wish to thank Ruqian Lu for directing us into this research.

References

[1] C.H. Bennett, G. Brassard, Quantum cryptography: public-key distribution and tossing, in: Proc. IEEE International conference on Computers, Systems and Signal Processing, 1984.

[2] A.K. Ekert, Quantum cryptography based on Bell's theorem, Physical Review Letters 67(6)(1991) 661-663.

[3] C.H. Bennett, G. Brassard, N.D. Mermin, Quantum cryptography without Bell's theorem, Physical Review Letters 68(5)(1992) 557-559.

[4] H.-K. Lo, H.F. Chau, .Unconditional security of quantum key distribution over arbitrarily long distances, Science 283(1999) 2050-2056.

[5] A. Cabello, Quantum key distribution in the holevo limit, Physical Review Letters 85(26)(2000) 5635-5638.

[6] T. Nguyen, M.A. Sfaxi, S. Ghernaouti-Hélie, 802.11i encryption key distribution using quantum cryptography, Journal of Networks 1(5)(2006) 9-20.

[7] R. Namiki, T. Hirano, Efficient-phase-encoding protocols for continuous-variable quantum key distribution using coherent states and postselection, Physical Review A 74(3)(2006) 032301.

[8] B. Qi, Y. Zhao, X.F. Ma, H.K. Lo, L. Qian, Quantum key distribution with dual detectors, Physical Review A 75(5)(2007) 052304.

[9] R. MatsumotoH, Quantum multiparty key distribution protocol without use of entanglement, Physical Review A 76(6)(2007) 062316.

[10] Y. Zhao, B. Qi, H.K. Lo, Quantum key distribution with an unknown and untrusted source, Physical Review A 77(5)(2008) 052327.

[11] T. Choi, M.S. Choi, Quantum key distribution using quantum faraday rotators, Journal of Physics: Condensed Matter 20(2008) 275242.

[12] K. Horodecki, M. Horodecki, P. Horodecki, D. Leung, J. Oppenheim, Quantum key distribution based on private states: unconditional security over untrusted channels with zero quantum capacity, IEEE Transaction on Information Theory 54(6)(2008) 2604-2620.

[13] J. Barrett, R. Colbeck, A. Kent, Unconditionally secure device-independent quantum key distribution with only two devices, Physical Review A 86(2012) 062326.

[14] C.H. Bennett, F. Bessette, G. Brassard, L. Salvail, J. Smolin, Experimental quantum cryptography, Journal of Cryptology 5(1)(1992) 3-28.

[15] T. Kimura, Y. Nambu, T. Hatanaka, A. Tomita, H. Kosaka, K. Nakamura, Single-photon interference over 150-km transmission using silica-based integrated-optic interferometers for quantum cryptography, Japanese Journal of Applied Physics 43(9A/B)(2004) L1217.

[16] W.T. Buttler, R.J. Hughes, P.G. Kwiat, S.K. Lamoreaux, G.G. Luther, G.L. Morgan, J.E. Nordholt, C.G. Peterson, C.M. Simmons, Practical free-space quantum key distribution over 1 km, Physical Review Letters 81(15)(1998) 3283-3286.

[17] R. Rivest, A. Sharmir, L. Adleman, A method for obstaining digital signature and public-key cryptosystem, Communications of ACM 21(2)(1978) 120-126.

[18] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness", W.H. Freeman, New York, 1979.

[19] L.J. van, Handbook of Theoretical Computer Science, Vol. A, Algorithms and Complexity, Elsevier, Amsterdam, 1998.

[20] P.W. Shor, Algorithms for quantum computation: discrete logarithm and factoring, in: Proc. 35th Annual IEEE Symposium on Foundations of Computer Science, 1994.

[21] D. Gottesman, I. Chuang, Quantum digital signatures. <https://arxiv.org/abs/quant-ph/0105032>, 2001 (accessed 08.03.01).

[22] G. Nikolopoulos, Applications of single-qubit rotations in quantum public-key cryptography, Physical Review A 77(2008) 032348.

[23] G. Nikolopoulos, L. Ioannou, Deterministic quantum-public-key encryption: forward search attack and randomization, Physical Review A 79(2009) 042327.

[24] L. Ioannou, M. Mosca, Public-key cryptography based on bounded quantum reference frames. <https://arXiv:quant-ph/0903.5156>, 2009 (accessed 30.03.09).

[25] L. Ioannou, M. Mosca, Unconditionally-secure and reusable public-key authentication, in: Proc. the 6th Conference on the Theory of Quantum Computation, Communication and Cryptography, 2011.

[26] U. Seyfarth, G. Nikolopoulos, G. Alber, Symmetries and security of a quantum-public-key encryption based on single-qubit rotations, Physical Review A 85(2012) 022342.

[27] J.W. Pan, C. Simon, C. Brukner, A. Zeilinger, Entanglement purification for quantum communication, Nature 410(2001) 1067-1070.

[28] J.W. Pan, S. Gasparoni, R. Ursin, G. Weihs, A. Zeilinger, Experimental entanglement purification of arbitrary unknown state, Nature 423(2003) 417-422.

[29] J.L. Romero, L. Roa, J.C. Retamal, C. Saavedra, Entanglement purification in cavity QED using local operation, Physical Review A 65(2002) 052319.

[30] M.R. Sprague, P.S. Michelberger, T.F.M. Champion, D.G. England, J. Nunn, X.-M. Jin, W.S. Kolthammer, A. Abdolvand, P.S. Russell, I.A. Walmsley, Broadband quantum memory in a hollow-core photonic-crystal fibre, Nature Photonics 8(4)(2014) 287-291.