Quantum Public-Key Cryptosystem Using Non-orthogonal States

Xiaoyu Li
School of Information Engineering, Zhengzhou University, Zhengzhou City, P. R. China
Email: iexyli@zzu.edu.cn

Lei Li
School of Physics Science and Engineering, Zhengzhou University, Zhengzhou City, P. R. China
Email: lilei@zzu.edu.cn

***Abstract*—In this paper we provide a quantum public-key cryptosystem using non-orthogonal states. A user Alice uses a set of particles which are in non-orthogonal quantum states as the public key kept by a key management center (KMC) while she keeps the states of the particles secret as the private key. By the help of KMC any other user can send encrypted message to Alice. Any one including KMC except Alice can't get the message. On the other hand digital signature can also be achieved by this public-key cryptosystem. There are no entangled states and complex operations needed in our cryptosystem. So it's easier to carry out in practice and more robust against possible attacks**.

***Index Terms*—public-key, quantum cryptography, non-orthogonal states, digital signature, security**

I. Introduction

The aim of cryptography is to send secret information through an insecure channel. To keep the information secret, people often integrate the original information (called "plain text") with some auxiliary information (called "key") to produce the encrypted information (called "cipher text"). Only the cipher text is transmitted so anyone can get the cipher text. But no one can recover the plain text except the authenticated user who has the key. Then two users who share the key can achieve secret communications. But how to distribute the key is the most important and most difficult problem. In fact there are nearly no unconditionally secure key distribution protocols in classical cryptography.

Quantum key distribution (QKD) protocol is a good way to solve this problem. In QKD protocols we can achieve unconditional secure key distribution with an insecure quantum channel and an insecure but authenticated classical channel. The first quantum key distribution protocol is proposed by C. H. Bennett and G. Brassard in 1984 (so called BB84 protocol) [1]. Since then many quantum key distribution protocols have been established and their securities have been studied, such as the EPR schemes [2], B92 [3], the scheme of Lo-Chau [4], and so on [5-13]. On the other hand experimental work for OKD has also succeeded. In 1992 Bennett, Bessette and Brassard first realized BB84 scheme in laboratory [14]. Recently QKD in optical fiber has been achieved beyond 150 km [15] and in free space has been implemented over a distance of 1 km [16].

All the QKD protocols above belong to symmetrical key protocols. There is a serious difficulty in symmetrical key protocols: how to distribute and manage keys if many users want to communicate with each other? If there are N users in a cryptosystem, a 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. On the other hand, N(N-1)/2 key distribution processes should be fulfilled before the cryptosystem begins to work. Obviously it's too tedious and too complex when N is a large number! Moreover in practice maybe the users don't trust each other, which make key distribution impossible. In classical cryptography a solution to overcome such difficulties is public-key cryptosystem, for example RSA algorithm [17]. Every user has a public key and a private key. The cipher text encrypted by the public key can only be decrypted by the private key while cipher text encrypted by the private key can only be decrypted by the public key. Moreover the public key and the private key are independent that we can't deduce one key from the other. A key management center (KMC) keeps all users' public keys which every user can get while every user keeps his or her private key secret which no other people can get. When a user Alice wants to send a secret message to another user Bob, she first asks KMC for the latter's public key .and encrypt the message by the public key. Then Alice sends the encrypted message (the cipher text) to Bob. When Bob receives the cipher text, he decrypted the cipher text by his private key. Finally Bob get the plain text. Any eavesdropper who catches the cipher text can't recover the plain text because he or she hasn't Bob's private key. Public-key cryptosystem has been widely used in all aspects of modern society, such as commercial affairs, military affairs, network communications, and et al. But as known in 1994 RSA algorithm has been proved to be unsafe on future quantum computer by Peter Shor [18]. So the classical public-key cryptosystems based on RSA algorithm will become unreliable inevitably. Quantum public-key

doi:10.4304/jsw.8.8.1906-1913

technology may be a good alterative solution. Gottesman first provided a quantum one-way function to design quantum digital signature protocol in [19], which may be used in a public-key system. In [20] a similar scheme is provided. In 2008 Nikolopoulos presented the first unconditional quantum public-key scheme [21] which uses single-particle rotation of unknown quantum states. Since then several public-key schemes have been developed [22-25].

In this paper we provide a quantum public-key cryptosystem using non-orthogonal states. It is based on the indistinguishability of non-orthogonal quantum states. 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. Our cryptosystem doesn't need entangled states and complex operations. So it's easier to carry out in practice and more robust against possible attacks.

II. BASIC IDEA

In quantum information science a quantum two-state particle is often called a qubit. A qubit's state space is a two-dimension Hilbert space. Such states are possible states of a qubit.

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

In which

$$|+>=\frac{1}{\sqrt{2}}(|0>+|1>)$$

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

We can notice that they aren't orthogonal to each other. As known the four states form two complete orthogonal basic vector sets

$$B_{01}=\{|0>,|1>\},$$

$$B_{+-}=\{|+>,|->\}\tag{3}$$

in which we can measure the qubit. It's known that non-orthogonal quantum states can't be discriminated with certainty, that is to say, there are no ways to determine one state of the four states in (1) with certainty. Now let's consider a public-key cryptosystem which includes a key management center (KMC) and N users. KMC keeps every user's public key which anyone can get to encrypt plain text while every user keeps his private key secret to decrypt cipher text. A user, for example Alice, creates an n-qubits sequence in which a qubit is in one of the state |0>, |1>, |+>, |-> at random. They constitute one public key of Alice's. At the same time Alice records the n qubits' states as a state sequence

$$\varphi=(|\varphi_1>|\varphi_2>...|\varphi_n>).\tag{4}$$

This is just Alice's private key. Then Alice gives her public key to KMC which is open to any user while she keeps her private key secret in order that no one except herself can get it. Now another user, for example Bob, wants to send a secret message to Alice. The message may be an n-bit string denoted *P* which we call the plain text. To encrypt the plain text to get the cipher text, Bob asks KMC for Alice's public key. After getting the qubit sequence, he sends a message through a public classical channel to Alice asking for the correct base to measure the qubits. Then Alice replies to him through the classical channel, asking him to measure the qubits according to the following key rule.

Key Rule:

If a qubit of the public key is in the state |0> or |1>, measure it in basis B_{01}; If a qubit of the public key is in the state |+> or |->, measure it in basis B_{+-}, Or in other words, Alice sends Bob a sequence of basis $B=(B_1B_2...B_n)$ in which $B_1,B_2,...,B_n\in\{B_{01},B_{+-}\}$.

The coding rules can be described as the following table.

TABLE I

KEY RULE

State	Bob's basis	Bob's measurement result
|0>	B_{01}	|0>
|1>	B_{01}	|1>
|+>	B_{+-}	|+>
|->	B_{+-}	|->

After finishing the measurements Bob will get φ at last. Moreover the state of any qubit hasn't changed. Now if Bob wants to send Alice an n-bit string, he can encode the information using the n qubits. To any bit of the string, if it's "0", Bob performs according to coding rule 1.

Coding Rule 1:

If the state of qubit is |0>, Bob do nothing; If the state of qubit is |1>, Bob reverses it to |0>; If the state of qubit is |+>, Bob do nothing; If the state of qubit is |->, Bob reverses it to |+>.

On the other hand, if the bit is "1", Bob performs according to coding rule 2.

Coding Rule 2:

If the state of qubit is |0>, Bob reverses it to |1>; If the state of qubit is |1>, Bob do nothing; If the state of qubit is |+>, Bob reverses it to |->; If the state of qubit is |->, Bob do nothing.

The process of the coding rules can be described as the following tables.

TABLE II

CODING RULE 1

Bit of the string	Original state	Bob's operation	Coded state
0	|0>	nothing	|0>
	|1>	reverse	|0>
	|+>	nothing	|+>
	|->	reverse	|+>

TABLE III

CODING RULE 2

Bit of the string	Original state	Bob's operation	Coded state
1	|0>	reverse	|1>
	|1>	nothing	|1>
	|+>	reverse	|->
	|->	nothing	|->

So the sequence of the n qubits is just the cipher text. Then Bob sends the n-qubit sequence to Alice. When Alice receives the qubits, she measures them according to key rule and records the results. Finally she gets a new sequence φ'. Then Alice compares every bit of φ' with the corresponding bit of φ and records according to decoding rule.

Decoding Rule:

If the two bits from φ' and from φ are same, she records it as "0"; if the two bits from φ' and form φ are converse, she records it as "1".

The coding rules can be described as the following table.

TABLE IV

DECODING RULE

Qubit (φ)	|0>	|0>	|1>	|1>	|+>	|+>	|->	|->
Qubit (φ')	|0>	|1>	|0>	|1>	|+>	|->	|+>	|->
Bit (P')	0	1	1	0	0	1	1	0

Finally Alice gets an n-bit string denoted as P'. Obviously we have $P'=P$, or in other words, Alice gets the plain text Bob wants to send her. In section 4 we will prove that by a well-designed scheme no we can affirm that no one except Alice and Bob can get the plain text. So the communication between Bob and Alice is secure.

There is still a problem left. The public key, or in other words, the n-qubit sequence is consumed after a communication process. So it can be used for only one time. If all the N-1 users want to send secret message to Alice, KMC must preserve at least N-1 public keys for Alice. Moreover Bob has gotten φ after a communication process with Alice! So the N-1 public keys must not be N-1 copies of a n-qubit sequence but N-1 different n-qubit sequences otherwise Bob will be able to get any message other user sends to Alice in future. In practice a user maybe needs to communicate with Alice for many times. So we can assume that KMC should keep M(M>>N) public key for Alice. So does every user in our cryptosystem. In order to discriminate the M public keys of Alice, every public key should be given a unique id number.

So we can design a feasible public-key cryptosystem based on this idea.

III. QUANTUM PUBLIC-KEY CRYPTOSYSTEM USING NON-ORTHOGONAL STATES

Now we present our public-key cryptosystem.

A. Building the Public-key Cryptosystem

First we assume that there are N users and a KMC in our public-key cryptosystem. They can communicate with each other through a classical channel and a quantum channel. KMC is trusted by every user while any two users don't trust each other. Every user creates M (M>>N) public keys. Every public key is an n-qubit sequence in which a qubit is in one of the states |0>, |1>, |+> or |-> at random. For example, KMC keeps Alice's M public keys denoted as

$$K_{PU} = \{ (i, Q_i), \quad i = 1,2,...M \} \tag{5}$$

in which Q_i is an n-qubit sequence and i is the id number. Every qubit of Q_i is in one of the states |0>, |1>, |+> or |-> at random. On the other hand, Alice keeps her private keys denoted as

$$K_{PR} = \{ (i, \varphi_i), \quad i = 1,2,...M \}$$

$$\varphi_i = (|\varphi_1>|\varphi_2>...|\varphi_n>)$$

$$|\varphi_1>, |\varphi_2>, ..., |\varphi_n> \in \{|0>, |1>, |+>, |->\}. \tag{6}$$

So does every user in the cryptosystem. All users' public keys are open to any user, in other words, any user can asks KMC for any public key of any other user. But a public key can only be given to one user because it will be consumed and no longer exist. At the same time every user must keep his or her private keys absolutely secret. A private key can also be used for one time so that it will be abandoned after finishing a communication process.

B. Process of the Secret Communication

If Bob wants to send a secret message denoted as a binary string *P* to Alice, they perform the following steps.

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

Step 2: KMC chooses a public key (j,Q_j) from Alice's K_{PU} at random and gives it to Bob.

Step 3: After receiving the public key, Bob gets the id number *j* and sends it to Alice through the classical channel.

Step 4: After receiving the id number j, Alice queries it in her K_{PR} and gets the corresponding φ_j. Then she replies to Bob through the classical channel, telling Bob to measure all the qubit of Q_j according to Key Rule, or in other words, she sends the measurement basis sequence B_j to Bob.

Step 5: When Bob receives Alice's dictates, Bob performs measurements as Alice asks. At last Bob gets φ_j.

Step 6: Bob encodes *P* on Q_j according to Coding Rule 1 and Coding Rule 2. So he gets a new n-qubit sequence Q_j'. Then Bob sends Q_j' to Alice.

Step 7: When Alice receives Q_j', she measures all the qubits according to Key Rule and records the measurement results. So she will get a new qubit sequence φ_j'.

Step 8: Alice compares every bit of φ_j' with the corresponding bit of φ_j and records according to Decoding Rule. Finally she will get a string *P*'. Obviously we have *P'*=*P*. So Alice gets the message which Bob wants to send 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 using our public-key cryptosystem.

C. Digital Signature

First all users agree to the following rule.

Signature Rule:

If the state is |0> or |+>, we records as "0"; If the state is |1> or |->, we records as "1".

If Bob sends secret a 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, for example SHA-1 algorithm.

Step 2: Bob chooses one of his private keys at random, for example φ_k. Then he produces an m-bit string *PK* from the first m items of φ_k according to Signature Rule.

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 *k* with *P'*. So he gets a string *P* which is the plain text to be submitted to Alice.

Notice that now the length of *P* should be n. So the length of the original message *P*' added with the length of the number 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 by doing the steps in section III.

After Alice gets the plain text, she can extract the original message *P*, the signed message *PS* and the id number *k.* To verify the signature, she does the following steps.

Step 1: Alice asks KMC for Bob's no. *k* public key Q_k.

Step 2: Alice asks Bob for the sequence of measurement basis B_k through the public classical channel.

Step 3: After receive B_k, Alice measure Q_k according to B_k. Then she takes the first m measurement results and records according to Signature Rule. Finally she gets an m-string *PK*' which is just equals 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*' by SHA-1 algorithm just as Bob does.

Step 6: Alice compare *PA*' and *PA.* If they are same, the verification succeeds. Alice can be sure that the message is just from Bob.

IV. SECURITY OF THE PUBLIC-KEY CRYPTOSYSTEM

Our public-key cryptosystem is secure. Two users can communicate with each other secretly. Any other people including KMC can not get the message. We prove it as follows.

Let's assume that an eavesdropper, for example, Eve, wants to get the message transmitted from Bob to Alice.

A. Impossibility for Eavesdropper to Get theMessage

Eve may listen to both the classical channel and the quantum channel, trying to get the message from Bob to Alice. She can not only get Alice's dictates to Bob in step 4 but also get the n-qubit sequence Q_j' sent form Bob to Alice in step 6. From Alice's dictates, she can get the basis sequence B_j to measure the qubit sequence. Now Eve can also measure Q_j' and get φ_j' just like Alice. But she can't get the message *P* that Bob wants to send Alice at all because she has no φ_j which is kept secret by Alice. The message *P* is encoded in the difference between φ_j' and φ_j. Eve can get no information about

P only from φ_j'. As known φ_j is a random state sequence. So the probability that Eve succeeds in getting *P* is no more than the probability she just guess the every bit of *P* which is

$$P_{error} = \left(\frac{1}{2}\right)^n . \quad (7)$$

If *n=1000,* we have

$$P_{error} = \left(\frac{1}{2}\right)^{1000} \approx 10^{-300} . \quad (8)$$

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

B. Impossibility for KMC to Get the Message

It's easy to prove that KMC can't get message that Bob sends to Alice even though it keeps the public keys and joins in the communications process. Alice's public key is a qubit sequence in which a qubit may be in one of the four states $\{|0>,|1>,|+>,|->\}$ at random. But Alice doesn't tell KMC in which state any qubit of the sequence is. Moreover the four states aren't orthogonal to each other. So it's impossible for KMC to find in which state the qubit is with certainty. The probability that KMC fortunately get to know a qubit's state correctly is

$$P_{KMC} = \frac{1}{4} \times (\frac{1}{2} \times 1 + \frac{1}{2} \times \frac{1}{2}) \times 4 = \frac{3}{4} . \quad (9)$$

If *n=1000*, we have

$$P_{KMC} = (\frac{3}{4})^{1000} \approx 10^{-125} . \quad (10)$$

That is to say, KMC can't get Alice's private key from measuring Alice's public key which it keeps. On the other hand quantum no-cloning thorem forbids KMC to make a copy of the public key (j,Q_j). When Alice sends B_j to step 4, KMC also can get it. But KMC has no (j,Q_j) now. So it's possible to get φ_j' for KMC. What KMC can do is nothing more than what Eve can do. So we can conclude that KMC also can't get the message communicated between Alice and Bob.

C. Impossibility for Eavesdropper to Distort the Message

We prove that the Eve can't distort the secret message from Bob to Alice. Eve may catch the qubit sequence Q_j' from Bob to Alice and try to produce a fake message to Alice. First Eve listens to Alice's dictates in step 4 and gets B_j. Then she catches the qubit sequence Q_j' in step 6. So Eve can get φ_j' by measuring Q_j' according to B_j. Obviously she can perform any operations on the qubits of Q_j' as she wants. Finally Alice will get a state sequence φ_j''. But Eve can't make Alice get the specified fake message because the message Alice will get is determined by the difference between φ_j and φ_j'' while Eve hasn't φ_j. This makes it impossible for Eve to design the correct φ_j'' to let Alice get the fake message by comparing φ_j with φ_j''. In fact Eve can only guess which state the item in φ is. The probability that she chances to guess correctly for all items is

$$P_{error} = \left(\frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times \frac{1}{2}\right)^n = \left(\frac{1}{2}\right)^n . \quad (11)$$

If *n=1000*, we have

$$P_{error} = \left(\frac{1}{2}\right)^{1000} \approx 10^{-300} . \quad (12)$$

That is to say, such attack also fails.

D. Impossibility for KMC to distort the Message

We can prove that the KMC can't distort the secret message from Bob to Alice, too. KMC may also catch the qubit sequence Q_j' from Bob to Alice and try to produce a fake message to Alice. First KMC can listen to Alice's dictates in step 4 and get B_j. Then it catches the qubit sequence Q_j' in step 6. So KMC can get φ_j' by measuring Q_j' according to B_j. All that above is just the same as what Eve can do. Can KMC get more information to help it in cheating? We can prove that it is impossible in fact. KMC keeps all Bob's public key. But it can't get φ_j at all because non-orthogonal states can't be distinguished with certainty. Moreover if KMC tries to performs measurement on the qubit of Q_j, the state of the qubit may be destroyed. The secret communication can't be finished correctly. Alice will find that something is wrong and abandon what she receives. So KMC can't get anything about the private key. What KMC can perform is only to perform operations on Q_j directly just as Eve may do in subsection C. Of course it also can't make Alice to accept a distort message just like Eve.

On the other hand KMC may try to produce a distort message by providing fake public key to Bob. We can prove that such attack can't succeed, either. First KMC produces a fake public key (j,FQ_j). When Bob asks KMC for Alice's public key, KMC give (j,FQ_j) to him. Then Bob sends *j* to Alice and Alice returns B_j to Bob. Since KMC knows nothing about φ_j, FQ_j is sure to be different from Q_j. Or in other words, B_j isn't the correct basis sequence for FQ_j. If Bob measures FQ_j according to B_j, he is sure to change the states of some qubits of FQ_j randomly. Now even

KMC doesn't know the state of the qubits in FQ_j. So it's impossible for KMC to make Alice to get a message which is just the one that it wants Alice to get. What KMC can achieve is only to make Alice getting a disordered and meaningless binary string. Of course Alice will find that something is wrong at once. So KMC's cheating can't succeed.

E. Security against Forward Search Attack

In classical public-key cryptosystem, how to defeat forward search attack is an important problem which can't be ignored. The forward search attack can be described as follows. Since Alice's public key is kept by KMC, every user who wants to sends a message to Alice must ask KMC for Alice's public key. All cipher texts are encrypted by Alice's public key. So Eve may encrypt many plain texts by Alice's public key to produce many cipher texts and save them in her database. Then Eve catches all cipher texts sent to Alice and queries them in her database. If she just finds that a cipher text which a user sends to Alice is the same as one cipher text in her database, she can conclude that the plain text which the user wants to send Alice is just the plain text she used to produce the cipher text in her database. Finally Eve gets the secret message transmitted to Alice. But in our quantum public-key cryptosystem, forward search attack is meaningless because Alice has many public keys in which a public key can be used only one time. Encrypting the same plain texts by different public keys of course produces different cipher texts.

So forward search attack is sure to be unsuccessful. This is a big advantage of our public-key system.

F. Security of Digital Signature

Now we prove that our cryptosystem can solve digital signature problem, too. How does Alice assure that the message is really from Bob? If Eve wants to impersonate Bob, she must produce signed message to cheat Alice. It's easy for Eve to produce the abstract *PA* from the message she wants to send Alice by SHA-1 algorithm. But Eve doesn't know Bob's private key at all which it's necessary to produce the signed message *PS*. Since Bob keeps his private key secret, what Eve can do is only to guess *PK*. So the probability for Eve to guess correctly for all the m bit of *PK* is

$$P_{error} = \left(\frac{1}{2}\right)^m . \tag{13}$$

If *m=100*, we have

$$P_{error} = \left(\frac{1}{2}\right)^{100} \approx 10^{-30} . \tag{14}$$

It's such a small probability. So Eve has no chances to cheat Alice successfully. Or in other words, Alice can assure that the message is from Bob. So we can say that our public-key cryptosystem provides a reliable signature method.

G. Security against Resend Attcak

In classical public-key cryptosystem, Eve may take the strategy of resend attack. She can catch the message sent form Bob to Alice and make a copy of it. Then she resends the message after some time, for example two days or two months. Obviously Alice has no means to percept such attack because the message is indeed from Bob. So Eve can make Alice to receive an outdated and repeated message although Eve doesn't know the message at all. To solve this problem, people should add timestamp to the original plain text so as Alice can find that the message is outdated. Obviously users have to pay more cost to producing and verifying timestamp.

In our quantum public-key cryptosystem, resend attack is not a problem at all. First Eve can catch Q_j' when it is sent from Bob to Alice. At the same time Eve can get B_j when Alice sends it to Bob through the public classical channel. So she can measure Q_j' according to B_j. Finally Eve can get φ_j'. Eve can make a copy of Q_j' without any difficulty by creating qubits according to φ_j'. But if Eve wants to fulfill a resend attack by resend to Q_j' to Alice, she can't achieve her goal. The reason is that in our public-key cryptosystem the public key Q_j and the private key φ_j are also used for one time. Alice won't measure Q_j' according to B_j to get φ_j' at all because B_j and φ_j have been abandoned. On the other hand Alice can only get a random string if she tries to use any private key φ_k ($k \neq j$) to decrypt the cipher text Q_j'.

So resend attack can't succeed in our public-key cryptosystem.

H. Security against Chosen Plain Text Attack

Our public-key cryptosystem is secure under chosen plain text attack. We prove it as follows.

In a chosen plain text attack, Eve is allowed to obtain a random number (plain text, cipher text) pairs of her choice. Then she tries to find some information about the key. In classical cryptography chosen plain text attack is a power tool to crash the cryptographic system if the number is large enough. But in our public-key cryptosystem the public key can be used for only one time. Different cipher texts are produced by different public keys. So there are no correlations between them. Eve can't find any laws which can help her to find some information about the key. Although Eve may get as many as possible (plain text, cipher text) pairs, she is still unable to get any information helpful to break our public-key cryptosystem. So chosen plain text attack is invalid to our public-key cryptosystem.

Now we have proved that our public-key cryptosystem is unconditionally secure.

V. Feasibility analysis of the public-key cryptosystem

First our public-key cryptosystem isn't an imaginary plan based on the technology which doesn't exist or the technology difficult to carry out. All that the users need to do are performing measurement on a qubit, reversing a qubit whose states is known and transmitting qubits through a quantum channel. There are no entangled states and complex quantum operations needed at all. So it is easier to carry out in practice.

Second as known quantum cryptography depends on the special properties of quantum system. But in practice quantum systems often undergo decoherence over time which makes them to lose quantum coherence and to turn into classical systems inevitably. It's the most important problem for quantum cryptographic protocol to work in practice. Especially in public-key cryptosystem, KMC needs to keep all users' public keys which are quantum systems for some time until a user asks for them. This brings a serious challenge for quantum public-key cryptosystem. To overcome this difficulty, we can use the quantum system which has bigger time length of decoherence, such as photon in Single-mode fiber. On the other hand users can update their public keys periodically. By means of such methods, our cryptosystem can perform well to satisfy all users.

Third all these discussions above are based on that Alice and Bob always using noiseless channels to build a key in our protocol. If there are no noiseless channels, can this protocol work? We can study it, too. Let's consider noisy classical channel first. In step 3, step 4 and step5, Bob and Alice exchange classical information which is necessary to the next step. If there errors in transmission, Bob is sure to fail. Fortunately classical error-correcting coding technology has been a mature and powerful tool. We can fulfill information transmission through a noisy classical channel with very low error rate by error-correcting coding, which guarantees the classical information to be correctly exchanged between Alice and Bob. On the other hand, in step 6 Bob sends the qubit sequence Q_j' to Alice through the quantum channel. If there are random errors existing, Alice will get mistaken bits, which also means communication failure. The solution is error-correcting coding, too. Although quantum error-correcting coding technology is not as mature as classical error-correcting coding technology, it can provide rather satisfying results for most quantum channel.

VI. Discussion and Conclusion

We have pointed out that a public key can be used for only one time in our cryptosystem. This limits the number of user. If KMC keeps M public keys for Alice, M users can send message to Alice at most. If one user needs to communicate with Alice for many times, the number who can exchange information with Alice will be further depressed. Such limit can be removed by developing cryptosystem in which public key can be reused. We will discuss it in future work.

In this paper we provide a quantum public-key cryptosystem using non-orthogonal states. N users can achieve secret communications by the help of a key management center. The laws of quantum mechanics guarantee that our cryptosystem is unconditionally secure. No one except the two parts involved in communication can get the message. The message can be signed so that the sender's identity can be verified. No entangled states and complex operations are needed. So our cryptosystem is easy to carry out in practice. Moreover it is proved to be secure against possible attacks.

Acknowledgment

The authors wish to thank Ruqian Lu for directing us into this research. This work is supported by Natural Science Foundation of China (Grants 61073023);

References

[1] C H. Bennett and G. Brassard, "Quantum cryptography: Public-key distribution and tossing", P*roceedings of IEEE International conference on Computers, Systems and Signal Processing,* Bangalore, India, pp.175, December 1984.

[2] A. K. Ekert, "Quantum cryptography based on Bell's theorem", *Physical Review Letters*, vol. 67, no. 6, pp.661-663, August 1991.

[3] C. H. Bennett, G. Brassard and N. D. Mermin, "Quantum cryptography without Bell's theorem", *Physical Review Letters*, vol. 68, no. 5, pp.557-559, February 1992.

[4] Hoi-Kwong Lo and H. F. Chau, "Unconditional Security of Quantum Key Distribution over arbitrarily long distances", *Science*, vol. 283, pp.2050-2056, February 1999.

[5] A. Cabello, "Quantum Key Distribution in the Holevo Limit", *Physical Review Letters,* vol. 85, no. 26, pp.5635-5638, December 2000.

[6] T. Nguyen, M. A. Sfaxi, S. Ghernaouti-Hélie, "802.11i encryption key distribution using quantum cryptography", *Journal of Networks*, v 1, nol. 5, pp. 9-20, September 2006.

[7] R. Namiki, T. Hirano, "Efficient-phase-encoding protocols for continuous-variable quantum key distribution using coherent states and postselection", *Physical Review* A, vol. 74, no. 3, pp.032301, September 2006.

[8] B. Qi, Y. Zhao, X. F. Ma, H. K. Lo, L. Qian, "Quantum key distribution with dual detectors", *Physical Review* A, vol. 75, no. 5, pp.052304, May 2007.

[9] R. MatsumotoH, "Quantum multiparty key distribution protocol without use of entanglement", *Physical Review* A, vol. 76, no. 6, pp.062316, June 2007.

[10] Y. Zhao, B. Qi, H. K. Lo, "Quantum key distribution with an unknown and untrusted source", *Physical Review* A, vol. 77, no. 5, pp.052327, May 2008.

[11] T. Choi, M. S. Choi, "Quantum Key Distribution Using Quantum Faraday Rotators", *Journal of Physics: Condensed Matter,* vol. 20, pp. 275242, May 2008.

[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*, vol. 54, no. 6, pp.2604-2620, June 2008.

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

[14] Charles H. Bennett, F. Bessette, G. Brassard, L. Salvail, J. Smolin, "Experimental quantum cryptography", *Journal of Cryptology*, vol. 5, no.1, pp.3-28, January 1992.
[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", *arXiv:quant-ph*/0403104.
[16] W. T. Buttler et al., "Practical Free-Space Quantum Key Distribution over 1 km", *Physical Review Letters*, vol. 81, no. 15, pp.3283-3286, October 1998.
[17] R. Rivest, A. Sharmir, L. Adleman, "A Method for Obstaining Digital Signature and Public-Key Cryptosystem", *Communications of ACM,* vol. 21, no. 2, pp. 120-126, February 1978.
[18] P. W. Shor, "Algorithms for quantum computation: Discrete logarithm and Factoring", *Proceedings of 35th Annual IEEE Symposium on Foundations of Computer Science,* Santa Fe, US, pp.124-134, 1994.
[19] D. Gottesman, I. Chuang, "Quantum Digital Signatures", *arXiv:quant-ph*/0105032.
[20] J. Zhang, "Arbitrated quantum signature protocol using EPR Pairs", *Journal of Networks*, v 7, n 11, p 1803-1810, November 2012.
[21] G. Nikolopoulos, "Applications of single-qubit rotations in quantum public-key cryptography", *Physical Review* A, 77, pp. 032348, March 2008.
[22] G. Nikolopoulos, L. Ioannou, "Deterministic quantum-public-key encryption: forward search attack and randomization", *Physical Review* A, 79, pp. 042327, April 2009.
[23] L. Ioannou, M. Mosca, "Public-key cryptography based on bounded quantum reference frames", *arXiv:quant-ph/*0903.5156.
[24] L. Ioannou, M. Mosca, "Unconditionally-secure and reusable public-key authentication", *Proceedings of the 6th Conference on the Theory of Quantum Computation, Communication and Cryptography*, pp.13-27, May 2011.
[25] U. Seyfarth, G. Nikolopoulos, G. Alber ,"Symmetries and security of a quantum-public-key encryption based on single-qubit rotations", *Physical Review* A, 85, pp. 022342, February 2012.

Xiaoyu Li was born in Nanyang, China in 1974. He received the Ph. D. degree in computer software and theory from Institute of Computing Technology, Chinese Academy of Sciences, China in 2004. He majors in quantum information and quantum computing; mobile computing.

He is an associate professor at School of Information Engineering, Zhengzhou University, China. Dr Li is now the member of Chinese Computer Federation.

Lei Li was born in Nanyang, China in 1981. He received the Ph. D. degree in information and communication Engineering and from Institute of Acoustics, Chinese Academy of Sciences, China in 2009. He majors in quantum information; photoelectric inspect & signal processing; embedded systems and applications.

He is a lecture at School of Physics Science and Engineering, Zhengzhou University, China.