PEA: Polymorphic Encryption Algorithm Based on Quantum Computation

Nikos Komninos and Georgios Mantas

Algorithms and Security Group
Athens Information Technology
GR-19002 Peania Attiki, Greece
nkom@ait.edu.gr

Abstract

In this paper, a polymorphic encryption algorithm (PEA), based on basic quantum computations, is proposed for the encryption of binary bits. PEA is a symmetric key encryption algorithm that applies different combinations of quantum gates to encrypt binary bits. PEA is also polymorphic since the states of the shared secret key control the different combinations of the ciphertext. It is shown that PEA achieves perfect secrecy and is resilient to eavesdropping and Trojan horse attacks. A security analysis of PEA is also described.

Keywords. *Encryption algorithm, polymorphism, quantum computations, CNOT and SWAP quantum gates.*

1. Introduction

Conventional cryptography consists of two types of cryptosystems; symmetric key cryptosystems and asymmetric key, or public key cryptosystems. Symmetric key cryptosystems, as the name implies, use identical keys for encryption and decryption and both the sender and the receiver share the same key(s). For this reason, such cryptosystems rely on the secrecy of the key. Otherwise, any malicious actors can deduce the plaintext from the ciphertext. The major disadvantage of such cryptosystems is the need of frequent and reliable key distribution. On the contrary, cryptosystems based on asymmetric key cryptography use different keys for ciphering and for deciphering. In ciphering, the key is announced publicly, but in deciphering, the key remains secret. The security of public key cryptosystems stems from the fact that the key pair generation is based on computational complexity of certain difficult mathematical problems. However, the main drawback of this cryptosystem is the computational cost of generating the key pair [19, 20].

In the early 1970s, a new type of Cryptography was proposed by Stephen Wiesner [10]. He proposed the idea of Quantum Cryptography as he introduced the concept of quantum conjugate coding at the same year. Quantum Cryptography is based on quantum computation. In contrast to Conventional Cryptography that uses digital bits for encoding information, Quantum Cryptography uses quantum particles (i.e. photons) and their quantized properties (i.e. photon’s polarization) to do that. Each photon carries one bit of quantum information, called qubit. A qubit defines not only the two binary eigenstates "0" and "1" but also the superposition of the two. Furthermore, the security of Quantum Cryptography is based on fundamental quantum mechanical principles. It uses the quantum mechanics in order to overcome the key distribution problem which is the most serious problem in the both types of conventional cryptography [6, 8, 9 10, 18]. Quantum Cryptography is considered as the basis for next generation cryptographic systems that may be able to replace the public key cryptosystems [3, 4, 7, 11, 16].

Conventional cryptosystems are susceptible to advanced technology, as new more powerful computers (e.g. quantum computers) may be able to override the burden of computationally complex problems in the future. Thus, no matter how strong a cryptographic algorithm is, soon or later it will be broken. On the contrary, Quantum Cryptography is independent of technological advance in the future as its security is provable information theoretically [4, 5].

During the last two decades, the Quantum Cryptography has focused on three fields concerned to key sharing between transmitter and receiver. These fields are the following: quantum key distribution [8], quantum secret sharing [10] and quantum bit commitment [1]. Furthermore, a new breakthrough appeared regarding the evolution of quantum encryption algorithms. The concept of this new field was first introduced by Boykin and Roychowdhury [12] in 2000. Thus, Quantum Cryptography does not only provide a secure key exchange between two parties, but also provides a special type of quantum encryption at transmitter and recipient.

Moreover, Quantum Cryptography solves the two major practical problems of one-time pad encryption scheme as it provides secure key distribution and is also able to generate sets of random numbers at the two communicating parties,. One – time pad is the only provably unconditionally secure cryptosystem, which can not be compromised even in the face of unlimited time and computational power [17, 19, 20]. Therefore, the existing encryption algorithms for quantum information take the form of one-time pad encryption method. Several quantum encryption algorithms for classical binary bits from different aspects were proposed in [2, 13, 14]. A realizable quantum encryption algorithm for qubits was also presented in [15]. Motivated by these and the challenge of quantum computing field, we propose a polymorphic encryption algorithm (PEA), which uses different combinations of quantum CNOT and SWAP gates in order to encrypt messages without changing the key. The Controlled NOT gate (CNOT) is the quantum analogue of the XOR gate. It has two input qubits, the first is the controlled qubit and the second is the target qubit. If the controlled qubit is zero then the target qubit is intact. If the controlled qubit is one then the target qubit is flipped. Generally, the notation $\mathrm{CNOT}_{x,y}$, means that the index x is the controlled qubit and the index y is the target qubit. The quantum SWAP gate has two input qubits and swaps them. The notation $\mathrm{SWAP}_{x,y}$, means that this gate swaps the qubit x with the qubit y. The security of the encryption algorithm is analyzed from several aspects and it is shown that the PEA can prevent quantum and classical attack.

Following this introduction, the paper is organized as follows. Section 2 describes the polymorphic encryption algorithm along with its encryption and decryption processes. Section 3, presents a security analysis of PEA and shows how resilient is to quantum and classical attacks. Finally, we conclude our algorithm in section 4 and propose future work.

2. Polymorphic Encryption Algorithm

The polymorphic encryption algorithm (PEA) is a symmetric key algorithm based on quantum computation for encryption of classical messages. PEA requires four groups of keys to be exchanged between sender and recipient before the encryption takes place. The first group key is responsible for the choice of quantum ancilla bits. The second group key is responsible for the choice of the first level of encryption. The third group key is responsible for the second level of encryption and finally, the fourth group key is responsible for leading the encrypted data to non-orthogonality. Hence, the pre-sharing of these keys is essential as they are used during encryption of PEA. The polymorphism of this algorithm is based on the pre-shared keys, as they are the factors that designate the internal paths that the algorithm should follow in order to give the encrypted output.

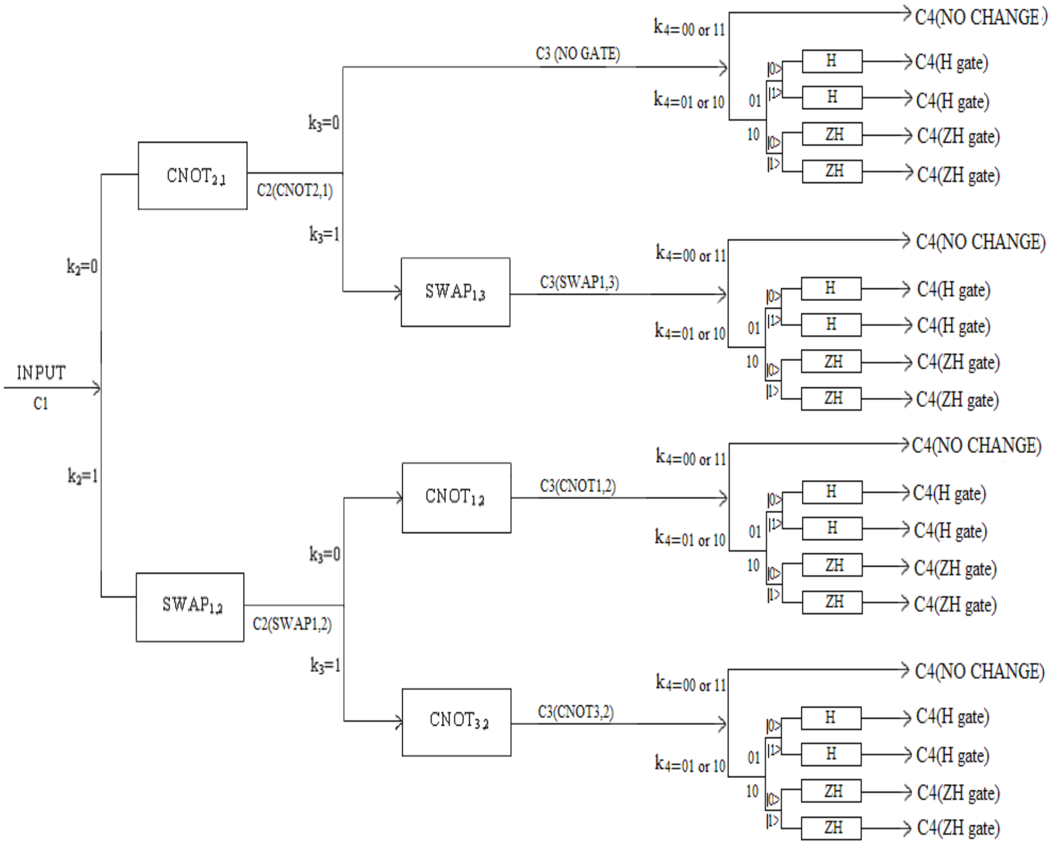

Figure 1. PEA Encryption Algorithm

Generally, the classical message, which is encrypted by PEA, consists of *m* bits and for each bit of the message one quantum state, C_1, is created based on the current bit and the key. So, *m* quantum states are going to be created. As illustrated in figure 1, the input of PEA is $C_{1,}$ which according to the second group of key element k_2 can follow one of the two paths. If k_2 is equal to 0, the quantum state will go through $CNOT_{2,1}$ gate. Whereas k_2 is equal to 1, the quantum state will go through $SWAP_{1,2}$ gate. Next, the output of $CNOT_{2,1}$ gate, $C_2(CNOT_{2,1})$, will pass either by the path which has no gate or by the $SWAP_{1,3}$, according to the third group key element k_3. Likewise, the output of $SWAP_{1,2}$ gate, $C_2(SWAP_{1,2})$, will pass either by $CNOT_{1,2}$ gate or by $CNOT_{3,2}$, according to the third group key element k_3.

2.1. Encryption

The encryption process consists of the following phases: preparation; first and second level of encryption; and non-orthogonality phases.

Phase 1: Preparation

In Phase 1, the key k_1 is used for the definition of the valid quantum states. Let we define the quantum ancilla state represented as $|k_1^1 k_1^2\rangle$. The k_1^1 and the k_1^2 are two key elements of the i_{th} key pair in k_1. When the classical key element pairs are 00, 01, 10 and 11 we have the corresponding quantum ancilla states $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$, respectively. For each bit, m (0 or 1), of the classical message, we have the corresponding quantum state $|m\rangle$. Now, the combination of the quantum ancilla state with the corresponding quantum state $|m\rangle$ of each bit, will give all valid quantum states. This combination is achieved by using the tensor product which generates the tensor product state $|mk_1^1 k_1^2\rangle$. To simplify our calculations, we have assumed an 8-bit classical message as a block size for our algorithm. Thus, we are going to have the following eight valid quantum states:

$|000\rangle$, $|001\rangle$, $|010\rangle$, $|011\rangle$, $|100\rangle$, $|101\rangle$, $|110\rangle$, $|111\rangle$.

These valid states can be represented by Equation 1:

$$C_1 = |\, m k_1^1\, k_1^2 >, \ where \ \ m, k_1^1, k_1^2 \in \{0,1\}\,. \qquad (\mathbf{0})$$

Phase 2: First Level of Encryption

In Phase 2, the second key, k_2, is responsible for the first level of encryption. The input to the first level of encryption is one of the eight valid values defined by Equation 1. According to the values of k_2, there is a different path that the input can follow. When k_2 is equal to 0, the input will pass by the $CNOT_{2,1}$ gate and when k_2 is equal to 1, the input will pass by the $SWAP_{1,2}$ gate. The index 2,1 of

CNOT gate defines the second qubit as control and the first qubit as target. Thus, the CNOT$_{2,1}$ gate inverts the first qubit when the second qubit is equal to 1. Likewise, the index 1,2 of SWAP gate defines that the first qubit will be swapped with the second qubit. We have selected the CNOT$_{2,1}$ and SWAP$_{1,2}$ gates to create diffusion to the data bit found in the first qubit of the quantum states, $|mk_1^1k_1^2\rangle$, of Phase 1.

The output C$_2$ that results from CNOT$_{2,1}$ and SWAP$_{1,2}$ quantum gates (see Figure 1), denoted as C$_2$(CNOT$_{2,1}$) and C$_2$(SWAP$_{1,2}$) respectively, is represented by Equation 2:

$$\begin{gathered} C_2(\text{CNOT}_{2,1}) = |t_m\, k_1^1\, k_1^2 >,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\ \ and\ t_m = m \oplus k_1^1, \\ \text{C}_2(\text{SWAP}\ _{1,2}) = |k_1^1\, m, k_1^2 >,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\,. \end{gathered} \tag{\textbf{0}}$$

Based on the second group of key element *k*$_2$, all possible combinations of C$_2$(CNOT$_{2,1}$) and C$_2$(SWAP$_{1,2}$) states are shown in Table 1:

Table 1. All possible C$_2$ output.

C$_1$	*k*$_2$	C$_2$(CNOT$_{2,1}$)	*k*$_2$	C$_2$(SWAP$_{1,2}$)
$\vert 0_m00\rangle$	0	$\vert 0_m00\rangle$	1	$\vert 00_m0\rangle$
$\vert 0_m01\rangle$	0	$\vert 0_m01\rangle$	1	$\vert 00_m1\rangle$
$\vert 0_m10\rangle$	0	$\vert 1_m10\rangle$	1	$\vert 10_m0\rangle$
$\vert 0_m11\rangle$	0	$\vert 1_m11\rangle$	1	$\vert 10_m1\rangle$
$\vert 1_m00\rangle$	0	$\vert 1_m00\rangle$	1	$\vert 01_m0\rangle$
$\vert 1_m01\rangle$	0	$\vert 1_m01\rangle$	1	$\vert 01_m1\rangle$
$\vert 1_m10\rangle$	0	$\vert 0_m10\rangle$	1	$\vert 11_m0\rangle$
$\vert 1_m11\rangle$	0	$\vert 0_m11\rangle$	1	$\vert 11_m1\rangle$

Phase 3: Second Level of Encryption

In Phase 3, the third key, *k*$_{3,}$ is responsible for the second level of encryption. In the second level of encryption, its inputs are the outputs of Phase 2. The outputs, C$_2$(CNOT$_{2,1}$) and C$_2$(SWAP$_{1,2}$), are the inputs to the second level of encryption and according to the values of *k*$_3$, there is a different path that each input can follow.

C$_2$(CNOT$_{2,1}$) will pass either from the path which has no gate, for *k*$_3$ equal to 0, or from the SWAP$_{1,3}$ gate, for *k*$_3$ equal to 1. When C$_2$(CNOT$_{2,1}$) follow the path which has no gate, the output will be the same as the input. The index 1,3 of SWAP gate defines that the first qubit will be swapped with the third qubit. The same as in Phase 1, we have selected the SWAP$_{1,3}$ gate to create diffusion to the inverted data bit which is found in the first qubit in the quantum states, $|\, t_m \, k_1^1 \, k_1^2 >$, of Phase 2.

The output C_3 that results from the path without quantum gate and the path with $SWAP_{1,3}$ quantum gate (see Figure 1), denoted as C_3(NO GATE) and $C_3(SWAP_{1,3})$ respectively, is represented by Equation 3:

$$\begin{aligned} &C_3\,(\text{NO GATE}) = |\, t_m \, k_1^1 \, k_1^2 >, \ \ where \ \ m, k_1^1, k_1^2 \in \{0,1\} \ \ and\, t_m \ = m \oplus k_1^1 \,, \\ &C_3\,(\text{SWAP}_{\ 1,3}) = | k_1^2, k_1^1, t_m >, \ \ where \ \ m, k_1^1, k_1^2 \in \{0,1\} \ and\, t_m \ = m \oplus k_1^1 \end{aligned} \tag{\textbf{0}}$$

Based on the third group of key, k_3, all possible combinations of C_3(NO GATE) and $C_3(SWAP_{1,3})$ states are shown in Table 2:

Table 2. All possible C_3 output with $C_2(CNOT_{2,1})$ input.

$C_2(CNOT_{2,1})$	k_3	C_3 (NO GATE)	k_3	$C_3(SWAP_{1,3})$
$\lvert 0_m00\rangle$	0	$\lvert 0_m00\rangle$	1	$\lvert 000_m\rangle$
$\lvert 0_m01\rangle$	0	$\lvert 0_m01\rangle$	1	$\lvert 100_m\rangle$
$\lvert 1_m10\rangle$	0	$\lvert 1_m10\rangle$	1	$\lvert 011_m\rangle$
$\lvert 1_m11\rangle$	0	$\lvert 1_m11\rangle$	1	$\lvert 111_m\rangle$
$\lvert 1_m00\rangle$	0	$\lvert 1_m00\rangle$	1	$\lvert 001_m\rangle$
$\lvert 1_m01\rangle$	0	$\lvert 1_m01\rangle$	1	$\lvert 101_m\rangle$
$\lvert 0_m10\rangle$	0	$\lvert 0_m10\rangle$	1	$\lvert 010_m\rangle$
$\lvert 0_m11\rangle$	0	$\lvert 0_m11\rangle$	1	$\lvert 110_m\rangle$

$C_2(SWAP_{1,2})$ will pass either from the $CNOT_{1,2}$ gate, when k_3 is equal to 0 or by $CNOT_{3,2}$ when k_3 is equal to 1. The index 1,2 of CNOT gate defines the first qubit as control and the second qubit as target. Thus, the $CNOT_{1,2}$ gate inverts the second qubit when the first qubit is equal to 1. Likewise, the index 3,2 of CNOT gate defines the third qubit as control and the second qubit as target. Thus, the $CNOT_{3,2}$ gate inverts the second qubit when the third qubit is equal to 1. We have selected the $CNOT_{1,2}$ and $CNOT_{3,2}$ gates to create diffusion to the data bit which is found in the second qubit in the quantum states, $|\, k_1^1 m k_1^2 \,\rangle$, of Phase 2.

The output C_3 that results from $CNOT_{1,2}$ and $CNOT_{3,2}$ quantum gates (see Figure 1), denoted as $C_3(CNOT_{1,2})$ and $C_3(CNOT_{3,2})$ respectively, is represented by Equation 4:

$$C_3\,(\text{CNOT}_{1,2}) = | k_1^1 \, t_m, k_1^2 >, \ \ where \ \ m, k_1^1, k_1^2 \in \{0,1\} \ \ and\, t_m \ = k_1^1 \oplus m \tag{\textbf{0}}$$

$$C_3(\text{CNOT}_{3,2}) = |k_1^1\, t_m, k_1^2 >, \ \ where \ \ m, k_1^1, k_1^2 \in \{0,1\} \ \ and\, t_m = k_1^2 \oplus m$$

Based on the third group of key element k_3, all possible combinations of $C_3(\text{CNOT}_{1,2})$ and $C_3(\text{CNOT}_{3,2})$ states are shown in Table 3:

Table 3. All possible C_3 output with $C_2(\text{SWAP}_{1,2})$ input.

$C_2(\text{SWAP}_{1,2})$	k_3	$C_3(\text{CNOT}_{1,2})$	k_3		$C_3(\text{CNOT}_{3,2})$
$\|00_m0\rangle$	0	$\|00_m0\rangle$	1		$\|00_m0\rangle$
$\|00_m1\rangle$	0	$\|00_m1\rangle$	1		$\|01_m1\rangle$
$\|10_m0\rangle$	0	$\|11_m0\rangle$	1		$\|10_m0\rangle$
$\|10_m1\rangle$	0	$\|11_m1\rangle$	1		$\|11_m1\rangle$
$\|01_m0\rangle$	0	$\|01_m0\rangle$	1		$\|01_m0\rangle$
$\|01_m1\rangle$	0	$\|01_m1\rangle$	1		$\|00_m1\rangle$
$\|11_m0\rangle$	0	$\|10_m0\rangle$	1		$\|11_m0\rangle$
$\|11_m1\rangle$	0	$\|10_m1\rangle$	1		$\|10_m1\rangle$

Phase 4: Non-Orthogonality

In Phase 4, the fourth group key k_4 is responsible for leading the encrypted data to non-orthogonality. The encrypted data derived from the second level of encryption are states which are orthogonal. However, orthogonality is a property which should be avoided so that the encrypted data are securely transferred through the communication channel. Orthogonality permits states to be distinguished and thus, phase 4 makes the outputs states non-orthogonal. Non-orthogonality is a preferable property for PEA.

Non-orthogonality is achieved by using the fourth key, k_4. According to the key, $k_{4,}$ we change the second qubit in the following approach: If the key element is 00 or 11, then we do not change the second qubit of state C_3. Thus, the second qubit remains in the state $|0\rangle$ or $|1\rangle$. However, when the key element is 01 or 10, then we perform the following computations to the second qubit: when the key element has value equal to 01, we apply Hadamard (H) gate [10] to the second qubit and the resulting output state is $|+\rangle$ if the input state is $|0\rangle$, or $|-\rangle$ if the input state is $|1\rangle$. If the key element is 10, we apply ZH gate [10] to the second qubit and the resulting output state is $|-\rangle$ if the input state is $|0\rangle$, or $|+\rangle$ if the input state is $|1\rangle$. We select the second qubit to be passed by H or ZH gates because in the second qubit there is the quantum representation of a data bit after Phase 3, and particularly into two outputs out of four.

The Non-Orthogonality process is shown in Figure 2. For each output C_3(NO GATE), $C_3(SWAP_{1,3})$, $C_3(CNOT_{1,2})$, and $C_3(CNOT_{3,2})$, non-orthogonality is also illustrated in Figure 1.

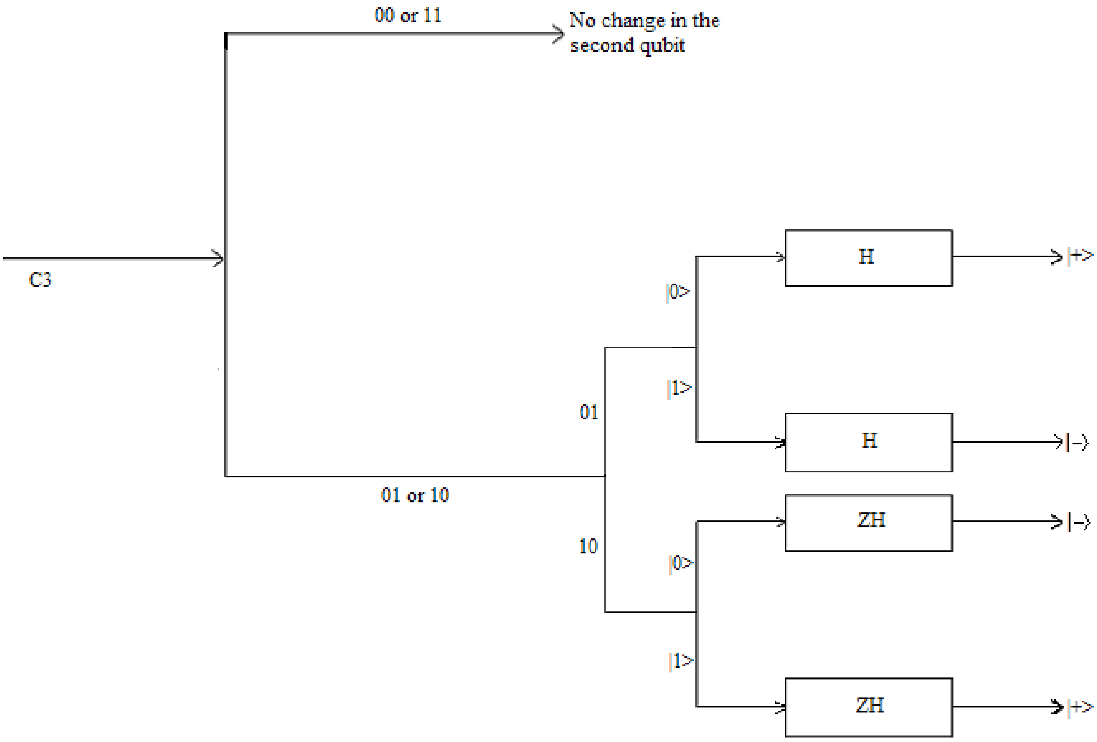

Figure 2. Non-Orthogonality Process

To prove that the utilization of Hadamard gates and ZH gates offers non-orthogonality, we need to follow the next steps:

Firstly, it is known that Hadamard gate is given by Equation 5 [10]:

$$\hat{H} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \textbf{(0)}$$

It is also known that the matrix representations of the $|0\rangle$ and the $|1\rangle$ are given by Equation 6 [10]:

$$|0>=\begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } |1>=\begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad \textbf{(0)}$$

From equations 5 and 6 we can calculate the following:

$$\hat{H}|0>=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix}=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix}=\frac{1}{\sqrt{2}}(|0>+|1>)\equiv|+>$$

$$\hat{H}|1>=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix}=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix}=\frac{1}{\sqrt{2}}(|0>-|1>)\equiv|->$$

$$\textbf{(0)}$$

Furthermore, the ZH gate is given by Equations [10]:

$$Z\hat{H} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \qquad (\mathbf{0})$$

and

$$|0> = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } |1> = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\,. \qquad (\mathbf{0})$$

From equations 8 and 9 we can calculate the following:

$$\begin{aligned} Z\hat{H}\,|0> &= \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix} = \frac{1}{\sqrt{2}}(|0> - |1>) \equiv |-> \\ Z\hat{H}\,|1> &= \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}(|0> + |1>) \equiv |+>\,. \end{aligned} \qquad (\mathbf{0})$$

For input C_3(NO GATE), the output C_4 results from not changing the second qubit of the input C_3(NO GATE) or applying H / ZH quantum gates to the second qubit of the input C_3(NO GATE). These results denoted as C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) are represented by Equation 11:

$$\begin{aligned} C_4(\text{NO CHANGE}) &= |m, k_1^1, k_1^2>,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\ \ when\ \ k_4 = 00\ \ or\ \ 11 \\ C_4(\text{H gate}) &= |m, \pm, k_1^2>,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\ \ when\ \ k_4 = 01 \\ C_4(\text{ZH gate}) &= |m, \mp, k_1^2>,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\ \ when\ \ k_4 = 10\,. \end{aligned} \qquad (\mathbf{0})$$

Based on the fourth group of key element k_4, all possible combinations of C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) states are shown in Table 4:

Table 4. All possible C_4 output with C_3(NO GATE) input.

C_3 (NO GATE)	k_4	C_4(NO CHANGE)	k_4	C_4(H gate)	k_4	C_4(ZH gate)
$\|0_m00\rangle$	00 or 11	$\|0_m00\rangle$	01	$\|0_m+0\rangle$	10	$\|0_m-0\rangle$
$\|0_m01\rangle$	00 or 11	$\|0_m01\rangle$	01	$\|0_m+1\rangle$	10	$\|0_m-1\rangle$
$\|1_m10\rangle$	00 or 11	$\|1_m10\rangle$	01	$\|1_m-0\rangle$	10	$\|1_m+0\rangle$
$\|1_m11\rangle$	00 or 11	$\|1_m11\rangle$	01	$\|1_m-1\rangle$	10	$\|1_m+1\rangle$
$\|1_m00\rangle$	00 or 11	$\|1_m00\rangle$	01	$\|1_m+0\rangle$	10	$\|1_m-0\rangle$
$\|1_m01\rangle$	00 or 11	$\|1_m01\rangle$	01	$\|1_m+1\rangle$	10	$\|1_m-1\rangle$
$\|0_m10\rangle$	00 or 11	$\|0_m10\rangle$	01	$\|0_m-0\rangle$	10	$\|0_m+0\rangle$
$\|0_m11\rangle$	00 or 11	$\|0_m11\rangle$	01	$\|0_m-1\rangle$	10	$\|0_m+1\rangle$

For input $C_3(SWAP_{1,3})$, the output C_4 results from not changing the second qubit of the input $C_3(SWAP_{1,3})$, or applying H or ZH quantum gates to the second qubit of the input $C_3(SWAP_{1,3})$. These results denoted as C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) respectively are represented by Equation 12:

$$\begin{aligned} C_4(\text{NO CHANGE}) &= |\, k_1^2, k_1^1, m>,\;\; where\;\; m, k_1^1, k_1^2 \in \{0,1\}\;\; when\;\; k_4 = 00\;\; or\;\; 11 \\ C_4(\text{H gate}) &= | k_1^2 \pm, m>,\;\; where\;\; m, k_1^1, k_1^2 \in \{0,1\}\;\; when\;\; k_4 = 01 \qquad (\mathbf{0}) \\ C_4(\text{ZH gate}) &= | k_1^2, \mp, m>,\;\; where\;\; m, k_1^1, k_1^2 \in \{0,1\}\;\; when\;\; k_4 = 10 \end{aligned}$$

Based on the fourth group of key element k_4, all possible combinations of C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) states are shown in Table 5:

Table 5. All possible C_4 output with $C_3(\text{SWAP}_{1,3})$ input.

$C_3(\text{SWAP}_{1,3})$	k_4	C_4(NO CHANGE)	k_4	C_4(H gate)	k_4	C_4(ZH gate)
$\|000_m\rangle$	00 or 11	$\|000_m\rangle$	01	$\|0{+}0_m\rangle$	10	$\|0{-}0_m\rangle$
$\|100_m\rangle$	00 or 11	$\|100_m\rangle$	01	$\|1{+}0_m\rangle$	10	$\|1{-}0_m\rangle$
$\|011_m\rangle$	00 or 11	$\|011_m\rangle$	01	$\|0{-}1_m\rangle$	10	$\|0{+}1_m\rangle$
$\|111_m\rangle$	00 or 11	$\|111_m\rangle$	01	$\|1{-}1_m\rangle$	10	$\|1{+}1_m\rangle$
$\|001_m\rangle$	00 or 11	$\|001_m\rangle$	01	$\|0{+}1_m\rangle$	10	$\|0{-}1_m\rangle$
$\|101_m\rangle$	00 or 11	$\|101_m\rangle$	01	$\|1{+}1_m\rangle$	10	$\|1{-}1_m\rangle$
$\|010_m\rangle$	00 or 11	$\|010_m\rangle$	01	$\|0{-}0_m\rangle$	10	$\|0{+}0_m\rangle$
$\|110_m\rangle$	00 or 11	$\|110_m\rangle$	01	$\|1{-}0_m\rangle$	10	$\|1{+}0_m\rangle$

For input $C_3(\text{CNOT}_{1,2})$, the output C_4 results from not changing the second qubit of the input $C_3(\text{CNOT}_{1,2})$, or applying H / ZH quantum gates to the second qubit of the input $C_3(\text{CNOT}_{1,2})$. These results denoted as C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) are represented by Equation 13:

$$\begin{aligned} C_4(\text{NO CHANGE}) &= | k_1^1, m, k_1^2 >,\;\; where\;\; m, k_1^1, k_1^2 \in \{0,1\}\;\; when\;\; k_4 = 00\;\; or\;\; 11 \\ C_4(\text{H gate}) &= | k_1^1, \pm, k_1^2 >,\;\; where\;\; m, k_1^1, k_1^2 \in \{0,1\}\;\; when\;\; k_4 = 01 \qquad (\mathbf{0}) \\ C_4(\text{ZH gate}) &= | k_1^1, \mp, k_1^2 >,\;\; where\;\; m, k_1^1, k_1^2 \in \{0,1\}\;\; when\;\; k_4 = 10 \end{aligned}$$

Based on the fourth group of key element k_4, all possible combinations of C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) states are shown in Table 6:

Table 6. All possible C_4 output with $C_3(\text{CNOT}_{1,2})$ input.

$C_3(\text{CNOT}_{1,2})$	k_4	C_4(NO CHANGE)	k_4	C_4(H gate)	k_4	C_4(ZH gate)
$\|00_m0\rangle$	00 or 11	$\|00_m0\rangle$	01	$\|0{+}_m0\rangle$	10	$\|0{-}_m0\rangle$
$\|00_m1\rangle$	00 or 11	$\|00_m1\rangle$	01	$\|0{+}_m1\rangle$	10	$\|0{-}_m1\rangle$

$\|11_m0\rangle$	00 or 11	$\|11_m0\rangle$	01	$\|1-_m0\rangle$	10	$\|1+_m0\rangle$
$\|11_m1\rangle$	00 or 11	$\|11_m1\rangle$	01	$\|1-_m1\rangle$	10	$\|1+_m1\rangle$
$\|01_m0\rangle$	00 or 11	$\|01_m0\rangle$	01	$\|0-_m0\rangle$	10	$\|0+_m0\rangle$
$\|01_m1\rangle$	00 or 11	$\|01_m1\rangle$	01	$\|0-_m1\rangle$	10	$\|0+_m1\rangle$
$\|10_m0\rangle$	00 or 11	$\|10_m0\rangle$	01	$\|1+_m0\rangle$	10	$\|1-_m0\rangle$
$\|10_m1\rangle$	00 or 11	$\|10_m1\rangle$	01	$\|1+_m1\rangle$	10	$\|1-_m1\rangle$

For input $C_3(CNOT_{3,2})$, the output C_4 results from not changing the second qubit of the input $C_3(CNOT_{3,2})$, or applying H / ZH quantum gates to the second qubit of the input $C_3(CNOT_{3,2})$,. These results denoted as C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) are represented by Equation 14:

$$\begin{gathered}
C_4(\text{NO CHANGE}) = |k_1^1, m, k_1^2>,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\ \ when\ \ k_4 = 00\ \ or\ \ 11 \\
C_4(\text{H gate}) = |k_1^1, \pm, k_1^2>,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\ \ when\ \ k_4 = 01 \\
C_4(\text{ZH gate}) = |k_1^1, \mp, k_1^2>,\ \ where\ \ m, k_1^1, k_1^2 \in \{0,1\}\ \ when\ \ k_4 = 10
\end{gathered} \tag{\mathbf{0}}$$

Based on the fourth group of key element k_4, all possible combinations of C_4(NO CHANGE), C_4(H gate) and C_4(ZH gate) states are shown in Table 7:

Table 7. All possible C_4 output with $C_3(CNOT_{3,2})$ input.

$C_3(CNOT_{3,2})$	k_4	C_4(NO CHANGE)	k_4	C_4(H gate)	k_4	C_4(ZH gate)
$\|00_m0\rangle$	00 or 11	$\|00_m0\rangle$	01	$\|0+_m0\rangle$	10	$\|0-_m0\rangle$
$\|01_m1\rangle$	00 or 11	$\|01_m1\rangle$	01	$\|0-_m1\rangle$	10	$\|0+_m1\rangle$
$\|10_m0\rangle$	00 or 11	$\|10_m0\rangle$	01	$\|1+_m0\rangle$	10	$\|1-_m0\rangle$
$\|11_m1\rangle$	00 or 11	$\|11_m1\rangle$	01	$\|1-_m1\rangle$	10	$\|1+_m1\rangle$
$\|01_m0\rangle$	00 or 11	$\|01_m0\rangle$	01	$\|0-_m0\rangle$	10	$\|0+_m0\rangle$
$\|00_m1\rangle$	00 or 11	$\|00_m1\rangle$	01	$\|0+_m1\rangle$	10	$\|0-_m1\rangle$
$\|11_m0\rangle$	00 or 11	$\|11_m0\rangle$	01	$\|1-_m0\rangle$	10	$\|1+_m0\rangle$
$\|10_m1\rangle$	00 or 11	$\|10_m1\rangle$	01	$\|1+_m1\rangle$	10	$\|1-_m1\rangle$

2.2. Decryption

All gates (CNOT gates, SWAP gates, H gates, ZH gates) used in the encryption and non-orthogonality processes are unitary operators. Hence, the decryption is the inverse process of the encryption using the pre-shared four group keys. The decryption process takes place at the recipient side. Thus, the decryption process consists of the following inverse phases:

Phase 1: Inverse process of Phase 4 of Encryption process

In Phase 1, the fourth group key k_4 is responsible for the first level of decryption. The recipient receives the non-orthogonal states (ciphertext) from the sender and applies the same process with the one applied in the Phase 4 during encryption. When the key element is 00 or 11 we do not change the second qubit of state C_4 and thus, the second qubit remains in state $|0\rangle$ or $|1\rangle$. However, when the key element is 01 or 10, then we perform the following computations to the second qubit of the ciphertext: If the key element has value equal to 01, we apply H gate to the second qubit and the output state is $|0\rangle$ when the input state is $|+\rangle$, or $|1\rangle$ when the input state is $|-\rangle$. If the key element is 10, we apply ZH gate to the second qubit and the resulting output state is $|0\rangle$ when the input state is $|-\rangle$, or $|1\rangle$ when the input state is $|+\rangle$. Therefore, the output of this phase is the state C_3 of the encryption process.

Phase 2: Inverse process of Phase 3 of Encryption process

In Phase 2, the third and second group keys k_3, k_2 are responsible for the second level of decryption. The inputs of phase 2 are the outputs of the first level of decryption. Here the recipient applies the same process with the process applied in phase 3 during the encryption process. For example, when k_2 is equal to 0, state C_3 passes either by the path which has no gate for k_3 equal to 0, or by $\text{SWAP}_{1,3}$, for k_3 equal to 1. When k_2 is equal to 1 state C_3 passes either by the $\text{CNOT}_{1,2}$ gate for k_3 equal to 0, or by $\text{CNOT}_{3,2}$ for k_3 equal to 1. Hence, the output of this phase is the state C_2 of the encryption process.

Phase 3: Inverse process of Phase 2 of Encryption process

In Phase 3, the second group of key element k_2 is responsible for the third level of decryption. The inputs of phase 3 are the outputs of the second level of decryption. Same as before the receiver applies the same process with the one applied in phase 2 during encryption. For example, when k_2 is equal to 0, state C_2 passes by the $\text{CNOT}_{2,1}$ gate, but when k_2 is equal to 1, state C_2 passes by the $\text{SWAP}_{1,2}$ gate. Hence the output of this phase is the state C_1 of the encryption process and the first qubit of each quantum state corresponds to the initial data bit.

Phase 4: Acquisition of plaintext

In Phase 4, we obtain all the first qubits of each state that correspond to the original bits and acquire the bit sequence of the original plaintext.

3. Security Analysis of PEA

To evaluate PEA we have conducted a security analysis to examine its resilience to Trojan horse and eavesdropping attacks and to study its homogeneity and perfect secrecy.

3.1. Homogeneity and Perfect Secrecy

It is essential an encryption algorithm to produce homogeneous ciphertexts and ideally have perfect secrecy, i.e. the ciphertext gives no information about the plaintext [20]. To prove that our algorithm is homogeneous and achieves perfect secrecy, it is enough to show that the density matrix of *n* ciphertexts states related to the *n* bits classical message is the identity matrix.

To prove it, let $|\psi_{wz}\rangle$ be the linear combination of all possible states with equal probability in the ciphertext set C_w (w=1, 2, 3), which corresponds to the z_{th} bit of message. The density matrices are calculated by Equation 15:

$$|\psi_{1z}\rangle\langle\psi_{1z}| = I \qquad |\psi_{2z}\rangle\langle\psi_{2z}| = I \qquad |\psi_{3z}\rangle\langle\psi_{3z}| = I \qquad \textbf{(0)}$$

The density matrix of *n* ciphertext states $|\psi_3\rangle$ for a message which consists of *n* bits is given by Equation 16:

$$|\psi_3\rangle\langle\psi_3| = |\psi_{31}\rangle\langle\psi_{31}| \otimes |\psi_{32}\rangle\langle\psi_{32}| \otimes |\psi_{33}\rangle\langle\psi_{33}| . . . \otimes |\psi_{3n}\rangle\langle\psi_{3n}| \qquad \textbf{(0)}$$

To calculate Equation 16, firstly it is required to calculate the outer product of each element $|\psi_{3i}\rangle$: $|\psi_{31}\rangle\langle\psi_{31}|$, $|\psi_{32}\rangle\langle\psi_{32}|$, $|\psi_{33}\rangle\langle\psi_{33}|$, $|\psi_{34}\rangle\langle\psi_{34}|$, $|\psi_{35}\rangle\langle\psi_{35}|$, $|\psi_{36}\rangle\langle\psi_{36}|$, $|\psi_{37}\rangle\langle\psi_{37}|$, $|\psi_{38}\rangle\langle\psi_{38}|$

Then, it is required to calculate the tensor products of the above outer products. We know that the elements $|\psi_{3i}\rangle$ represented as the following: $|000\rangle$, |

001⟩, |010⟩, |011⟩, |100⟩, |101⟩, |110⟩, |111⟩, can be also represented as vectors. By using vectors we can make simple calculations and prove that the ciphertext is homogenous. For example, the vector representation of the first states |000⟩ is given by Equation 17:

$$|000\rangle = |0\rangle \otimes |0\rangle \otimes |0\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1\\0\\0\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1\\0\\0\\0\\0\\0\\0\\0\end{bmatrix} \qquad (\mathbf{0})$$

Likewise, we can construct the rest of the states |001⟩, |010⟩, |011⟩, |100⟩, |101⟩, | 110⟩, and |111⟩ as vectors.

According to Figure 1, when the output is given by C_3 (NO GATE) we have the following possible states:

Table 8. C_3 (NO GATE) Output.

$\|0_m00\rangle$	$\|0_m01\rangle$	$\|1_m10\rangle$	$\|1_m11\rangle$	$\|1_m00\rangle$	$\|1_m01\rangle$	$\|0_m10\rangle$	$\|0_m11\rangle$

From Table 8, we have:

$$|\psi_{31}\rangle=|0_m00\rangle,\quad |\psi_{32}\rangle=|0_m01\rangle,\quad |\psi_{33}\rangle=|1_m10\rangle,\quad |\psi_{34}\rangle=|1_m11\rangle,$$
$$|\psi_{35}\rangle=|1_m00\rangle,\quad |\psi_{36}\rangle=|1_m01\rangle,\quad |\psi_{37}\rangle=|0_m10\rangle,\quad |\psi_{38}\rangle=|0_m11\rangle.$$

Next, the outer products can be calculated by Equation 18:

$$\begin{gathered} |\psi_{31}\rangle\langle\psi_{31}| = |0_m00\rangle\langle 0_m00| = ee^T = E \\ \text{where} \\ e = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T, \\ E = \mathrm{Diag}\{1, 0, 0, 0, 0, 0, 0, 0\} \end{gathered} \qquad (\mathbf{18})$$

Likewise, we can calculate all the outer products $|\psi_{32}\rangle\langle\psi_{32}|$, $|\psi_{33}\rangle\langle\psi_{33}|$, $|\psi_{34}\rangle\langle\psi_{34}|$, $|\psi_{35}\rangle\langle\psi_{35}|$, $|\psi_{36}\rangle\langle\psi_{36}|$, $|\psi_{37}\rangle\langle\psi_{37}|$, and $|\psi_{38}\rangle\langle\psi_{38}|$.

According to Figure 1, when the output is given by C_3(SWAP 1,3) we have the following possible states:

Table 9. C_3 (SWAP 1,3) Output.

$\|000_m\rangle$	$\|100_m\rangle$	$\|011_m\rangle$	$\|111_m\rangle$	$\|001_m\rangle$	$\|101_m\rangle$	$\|010_m\rangle$	$\|110_m\rangle$

From Table 9, we have:

$$|\psi_{31}\rangle=|000_{m}\rangle,\ |\psi_{32}\rangle=|100_{m}\rangle,\ |\psi_{33}\rangle=|011_{m}\rangle,\ |\psi_{34}\rangle=|111_{m}\rangle,$$

$$|\psi_{35}\rangle=|001_{m}\rangle,\ |\psi_{36}\rangle=|101_{m}\rangle,\ |\psi_{37}\rangle=|010_{m}\rangle,\ |\psi_{38}\rangle=|110_{m}\rangle.$$

The outer product of $|\psi_{31}\rangle\langle\psi_{31}|$ is given by Equation 19:

$$|\psi_{31}\rangle\langle\psi_{31}| = |000_{m}\rangle\langle 000_{m}| = ee^{T}= E$$

where

$$e = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^{T},$$

$$E = \mathrm{Diag}\{1, 0, 0, 0, 0, 0, 0, 0\}$$

(**19**)

Likewise, we can calculate $|\psi_{32}\rangle\langle\psi_{32}|$, $|\psi_{33}\rangle\langle\psi_{33}|$, $|\psi_{34}\rangle\langle\psi_{34}|$, $|\psi_{35}\rangle\langle\psi_{35}|$, $|\psi_{36}\rangle\langle\psi_{36}|$, $|\psi_{37}\rangle\langle\psi_{37}|$, and $|\psi_{38}\rangle\langle\psi_{38}|$.

According to Figure 1, when the output is given by $C_3(CNOT_{1,2})$ we have the following possible states:

Table 10. $C_3(CNOT_{1,2})$ Output.

$\|00_{m}0\rangle$	$\|00_{m}1\rangle$	$\|11_{m}0\rangle$	$\|11_{m}1\rangle$	$\|01_{m}0\rangle$	$\|01_{m}1\rangle$	$\|10_{m}0\rangle$	$\|10_{m}1\rangle$

From Table 10, we have:

$$|\psi_{31}\rangle=|00_{m}0\rangle,\ |\psi_{32}\rangle=|00_{m}1\rangle,\ |\psi_{33}\rangle=|11_{m}0\rangle,\ |\psi_{34}\rangle=|11_{m}1\rangle,$$

$$|\psi_{35}\rangle=|01_{m}0\rangle,\ |\psi_{36}\rangle=|01_{m}1\rangle,\ |\psi_{37}\rangle=|10_{m}0\rangle,\ |\psi_{38}\rangle=|10_{m}1\rangle.$$

The outer product of $|\psi_{31}\rangle\langle\psi_{31}|$ is given by Equation 20:

$$|\psi_{31}\rangle\langle\psi_{31}| = |00_{m}0\rangle\langle 00_{m}0| = ee^{T}= E$$

where

$$e = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^{T},$$

$$E = \mathrm{Diag}\{1, 0, 0, 0, 0, 0, 0, 0\}$$

(**0**)

Likewise, we can calculate $|\psi_{32}\rangle\langle\psi_{32}|$, $|\psi_{33}\rangle\langle\psi_{33}|$, $|\psi_{34}\rangle\langle\psi_{34}|$, $|\psi_{35}\rangle\langle\psi_{35}|$, $|\psi_{36}\rangle\langle\psi_{36}|$, $|\psi_{37}\rangle\langle\psi_{37}|$, and $|\psi_{38}\rangle\langle\psi_{38}|$.

According to Figure 1, when the output is given by $C_3(CNOT_{3,2})$ we have the following possible states:

Table 11. $C_3(CNOT_{3,2})$ Output.

$\|00_{m}0\rangle$	$\|01_{m}1\rangle$	$\|10_{m}0\rangle$	$\|11_{m}1\rangle$	$\|01_{m}0\rangle$	$\|00_{m}1\rangle$	$\|11_{m}0\rangle$	$\|10_{m}1\rangle$

From Table 11, we have:

$$|\psi_{31}\rangle = |00_m0\rangle, |\psi_{32}\rangle = |01_m1\rangle, |\psi_{33}\rangle = |10_m0\rangle, |\psi_{34}\rangle = |11_m1\rangle,$$
$$|\psi_{35}\rangle = |01_m0\rangle, |\psi_{36}\rangle = |00_m1\rangle, |\psi_{37}\rangle = |11_m0\rangle, |\psi_{38}\rangle = |10_m1\rangle.$$

The outer product of $|\psi_{31}\rangle\langle\psi_{31}|$ is given by Equation 21:

$$\begin{gathered} |\psi_{31}\rangle\langle\psi_{31}| = |00_m0\rangle\langle 00_m0| = ee^T = E \\ \text{where} \\ e = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T, \\ E = \mathrm{Diag}\{1, 0, 0, 0, 0, 0, 0, 0\} \end{gathered} \qquad (\mathbf{0})$$

Likewise, we can calculate $|\psi_{32}\rangle\langle\psi_{32}|$, $|\psi_{33}\rangle\langle\psi_{33}|$, $|\psi_{34}\rangle\langle\psi_{34}|$, $|\psi_{35}\rangle\langle\psi_{35}|$, $|\psi_{36}\rangle\langle\psi_{36}|$, $|\psi_{37}\rangle\langle\psi_{37}|$, and $|\psi_{38}\rangle\langle\psi_{38}|$.

We notice that the first output ($|00_m0\rangle$) given by $C_3(CNOT_{1,2})$ (see Table 10) is the same with the first output ($|00_m0\rangle$) given by $C_3(CNOT_{3,2})$ (see Table 11). Consequently, the linear combination $|\psi_{31}\rangle$ corresponds to the state $|00_m0\rangle$ is the same for the both cases. Thus, equation (20) and equation (21) are identical.

According to the assumption that k_1, k_2, k_3, k_4, are uniformly distributed and based on the above matrices of the outer products, we can calculate the tensor products and obtain the following density matrix:

$$|\psi_3\rangle\langle\psi_3| = |\psi_{31}\rangle\langle\psi_{31}| \otimes |\psi_{32}\rangle\langle\psi_{32}| \otimes |\psi_{33}\rangle\langle\psi_{33}| \ldots \otimes |\psi_{3n}\rangle\langle\psi_{3n}| = I \qquad (\mathbf{0})$$

Equation 22 demonstrates that the ciphertext is homogeneous and has perfect secrecy, since the density matrix is equal to the identity matrix.

3.2. Trojan horse attacks tolerance

The Trojan horse attacks are the most serious threats to computer networking security. However, the proposed algorithm is tolerant against Trojan horse attacks. If a Trojan horse, denoted as T, has invaded in the sender or the recipient in order to identify the quantum states $|0\rangle$ and $|1\rangle$, then the Trojan horse is going to take the following ciphertext state:

$$\begin{aligned}|\psi_{4i}(T)\rangle = \frac{1}{48}(&|0^{\|}_{m}0^{\|}0^{\|}> + |0^{\|}_{m}0^{\|}1^{\perp}> + |1^{\perp}_{m}1^{\perp}0^{\|}> + |1^{\perp}_{m}1^{\perp}1^{\perp}> + |1^{\perp}_{m}0^{\|}0^{\|}> \\ &+|1^{\perp}_{m}0^{\|}1^{\perp}> + |0^{\|}_{m}1^{\perp}0^{\|}> + |0^{\|}_{m}1^{\perp}1^{\perp}> + |0^{\|}0^{\|}0^{\|}_{m}> + |1^{\perp}0^{\|}0^{\|}_{m}> \\ &+|0^{\perp}1^{\perp}1^{\perp}_{m}> + |1^{\perp}1^{\perp}1^{\perp}_{m}> + |0^{\|}0^{\|}1^{\perp}_{m}> + |1^{\perp}0^{\|}1^{\perp}_{m}> + |0^{\|}1^{\perp}0^{\|}_{m}> \\ &+|1^{\perp}1^{\perp}0^{\|}_{m}> + |0^{\|}0^{\|}_{m}0^{\|}> + |0^{\|}1^{\perp}_{m}1^{\perp}> + |1^{\perp}0^{\|}_{m}0^{\|}> + |1^{\perp}1^{\perp}_{m}1^{\perp}> \\ &+|0^{\|}1^{\perp}_{m}0^{\|}> + |0^{\|}0^{\|}_{m}1^{\perp}> + |1^{\perp}1^{\perp}_{m}0^{\|}> + |1^{\perp}0^{\|}_{m}1^{\perp}> + |0^{\|}_{m} {+}^{?} 0^{\|}> \\ &+|0^{\|}_{m} {+}^{?} 1^{\perp}> + |1^{\perp}_{m} {-}^{?} 0^{\|}> + |1^{\perp}_{m} {-}^{?} 1^{\perp}> + |1^{\perp}_{m} {+}^{?} 0^{\|}> + |1^{\perp}_{m} {+}^{?} 1^{\perp}> \\ &+|0^{\|}_{m} {-}^{?} 0^{\|}> + |0^{\|}_{m} {-}^{?} 1^{\perp}> + |0^{\|} {+}^{?} 0^{\|}_{m}> + |1^{\perp} {+}^{?} 0^{\|}_{m}> + |0^{\perp} {-}^{?} 1^{\perp}_{m}> \\ &+|1^{\perp} {-}^{?} 1^{\perp}_{m}> + |0^{\|} {+}^{?} 1^{\perp}_{m}> + |1^{\perp} {+}^{?} 1^{\perp}_{m}> + |0^{\|} {-}^{?} 0^{\|}_{m}> + |1^{\perp} {-}^{?} 0^{\|}_{m}> \\ &+|0^{\|} {+}^{?}_{m} 0^{\|}> + |0^{\|} {-}^{?}_{m} 1^{\perp}> + |1^{\perp} {+}^{?}_{m} 0^{\|}> + |1^{\perp} {-}^{?}_{m} 1^{\perp}> + |0^{\|} {-}^{?}_{m} 0^{\|}> \\ &+|0^{\|} {+}^{?}_{m} 1^{\perp}> + |1^{\perp} {-}^{?}_{m} 0^{\|}> + |1^{\perp} {+}^{?}_{m} 1^{\perp}>)\end{aligned} \tag{\mathbf{0}}$$

In Equation 23, we have used three symbols. The symbol || is the information obtained by the Trojan horse when the quantum state is |0⟩. The symbol $\perp$ is the information obtained by the Trojan horse when the quantum state is |1⟩. Whereas, the symbol ? is the information obtained by the Trojan horse when the quantum state is |+⟩ or |-⟩. It means that when the state is |+⟩ or |-⟩ due to non-orthogonality that is applied in the fourth phase of encryption process, the Trojan horse can not decide which the valid state is. Moreover, according to the design of encryption algorithm, the quantum representation of a data bit can be in the first qubit or in the second qubit or in the third qubit. Thus, the Trojan horse is able to take any information related to the plaintext as well as which qubit is related to the ancilla qubit.

3.3. Eavesdropping attacks tolerance

According to the design of PEA, the ciphertext states are non-orthogonal leading the ciphertext states to be undistinguishable by an eavesdropping attacker. The non-orthogonality is applied in the fourth phase of encryption process and we can prove it by calculating the $\langle\psi_{4i} \mid \psi_{4i}\rangle$. If the value of inner product $\langle\psi_{4i} \mid \psi_{4i}\rangle$ is larger than 0, it means that the ciphertext states are non-orthogonal [10].

We know that $|\psi_{4i}\rangle$ (Equation 23) must be represented as a vector in order to calculate the value of $\langle\psi_{4i} \mid \psi_{4i}\rangle$. The states |000⟩, |001⟩, |010⟩, |011⟩, |100⟩, |101⟩, |110⟩, and |111⟩ have already been formed as vectors and it is necessary to also represent the states |0+0⟩, |0-0⟩, |0+1⟩, |0-1⟩, |1+0⟩, |1+1⟩, |1-0⟩, and |1-1⟩ as vectors. For example, the vector representation of |0+0⟩ is given by Equation 24:

$$|0{+}0\rangle = |0\rangle \otimes |+\rangle \otimes |0\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\0\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}}\\0\\\frac{1}{\sqrt{2}}\\0\\0\\0\\0\\0\end{bmatrix} \qquad (\mathbf{0})$$

Likewise, we can calculate |0-0⟩, |0+1⟩, |0-1⟩, |1+0⟩, |1+1⟩, |1-0⟩, and |1-1⟩. Hence, according to Equations 17, 25 and their further calculations we have:

$$|\psi_{4i}> = \frac{3}{48}\left\{\begin{bmatrix}1\\1\\1\\1\\1\\1\\1\\1\end{bmatrix} + \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\0\\0\\0\\0\end{bmatrix} + \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\-\frac{1}{\sqrt{2}}\\-\frac{1}{\sqrt{2}}\\0\\0\\0\\0\end{bmatrix} + \begin{bmatrix}0\\0\\0\\0\\\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} + \begin{bmatrix}0\\0\\0\\0\\\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\\-\frac{1}{\sqrt{2}}\\-\frac{1}{\sqrt{2}}\end{bmatrix}\right\} = \frac{1}{16}\left\{\begin{bmatrix}1+\sqrt{2}\\1+\sqrt{2}\\1\\1\\1+\sqrt{2}\\1+\sqrt{2}\\1\\1\end{bmatrix}\right\} \qquad (\mathbf{0})$$

Next, we can calculate the inner product ⟨ψ_{4i} | ψ_{4i}⟩ as shown in Equation 26:

$$<\psi_{4i}\,|\,\psi_{4i}> = \frac{1}{256}\left\{\begin{bmatrix}1+\sqrt{2} & 1+\sqrt{2} & 1 & 1 & 1+\sqrt{2} & 1+\sqrt{2} & 1 & 1\end{bmatrix}\cdot\begin{bmatrix}1+\sqrt{2}\\1+\sqrt{2}\\1\\1\\1+\sqrt{2}\\1+\sqrt{2}\\1\\1\end{bmatrix}\right\} = \frac{1}{32}\left(2+\sqrt{2}\right) \qquad (\mathbf{0})$$

and:

$$<\psi_{4i} | \psi_{4i} >= \frac{1}{32}\left(2+\sqrt{2}\right) > 0 \qquad (\mathbf{0})$$

By proving that the value of inner product $<\psi_{4i} | \psi_{4i}>$ is larger than 0, it means that the ciphertext states are non-orthogonal.

4. Conclusion

To our knowledge there are still not many good and feasible quantum encryption algorithms proposed. With the rapid progress of quantum information theory and technology, quantum information comes into real life quietly. When the quantum computers come true some day, it will be necessary and not always possible to transfer the existing encryption algorithms into quantum information. Based on the basic principle of quantum computation, a quantum cryptographic algorithm to encrypt the classical binary bits was proposed. The security of the encryption algorithm was analyzed in detail. It was shown that the proposed algorithm can prevent quantum as well as classical attacks.

PEA has several properties. First of all, no quantum state is pre-shared or stored making PEA possible and efficient in real applications. Second, it achieves perfect secrecy with the condition that the key is uniformly distributed. Third, both encryption and decryption are based on simple quantum computation, and particularly on a combination of the quantum CNOT and SWAP gates. Fourth, its implementation is feasible with the existing technology. At last, the same algorithm can be extended to encrypt quantum information.

References

[1] Nanrum Zhou, Guihua Zeng, Yiyou Nie, Jin Xiong and Fuchen Zhu, "A novel quantum block encryption algorithm based on quantum computation", *Physica A: Statistical Mechanics and its Applications*, Volume 362, Issue 2, 1 April 2006, Pages 305-313

[2] Osamu Hirota, Masaki Sohma, Masaru Fuse, Kentaro Kato, "Quantum stream cipher by the Yuen 2000 protocol: Design and experiment by an intensity-modulation scheme", *The American Physical Society*, 2005

[3] N. Gisin, S. Wolf, "Quantum Cryptography on noisy Channels: Quantumversus Classical Key-Agreement Protocols", *The American Physical Society*, 1999

[4] Rob Pike, "An Introduction to Quantum Computation and Quantum Communication", *Bell Labs, Lucent Technologies*, June 23, 2000

[5] Yoshihiro NAMBU and Hideo KOSAKA, "Introduction of Quantum Cryptography and Its Development", *NEC Res. & Develop*., Vol 44, No. 3, July 2003

[6] Emanuel Knill, Raymond Laflamme, Howard N. Barnum, Diego A.Dalvit, Jacek J. Dziarmaga, James E. Gubernatis, Leonid Gurvits, Gerardo Ortiz, Lorenza Viola, and Wojciech H. Zurek, "Quantum Information Processing: *A* hands-on primer", *Los Alamos Science* Number 27, 2002

[7] Ari Y. Benbasat, "A Survey of Current Optical Security Techniques", *MIT Media Lab*, April 15, 1999

[8] Steven Linton, "Quantum Cryptography-Quantum Key Distribution", July 13, 2003

[9] Dheera Venkatraman, "Methods and implementation of quantum cryptography", *MIT Department of Physics*, 27 April 2004

[10] Michael A. Nielsen and Isaac L. Chuang, *Quantum Computation and Quantum Information*, Cambridge Press, 2004

[11] X. Y. Li, H. Barnum, "Quantum authentication using entangled states", *International Journal of Foundations Computer Science*, Volume 15, Issue 4, 2004, Pages: 609-617.

[12] P.O. Boykin and V. Roychowdhury, "Optimal encryption of quantum bits", *Physics Review A*, Vol. 67, 2003.

[13] H. Bechmann-Pasquinucci and N. Gisin**,** "Intermediate states in quantum cryptography and Bell inequalities", *Physics Review A*, Vol 67, 062310, 2003

[14] N. Zhou, G. Zeng, "Progress on Cryptography", *The Kluwer International Series in Engineering and Computer Science*, Kluwer Academic Publishers, Boston, 2004, Pages: 195–200.

[15] N. Zhou, G. Zeng, "A realizable quantum encryption algorithm for qubits", China Physics, Vol. 14, Issue 2164, 2005

[16] G. Benenti and G. Casatti, *Principles of Quantum Computation, vol. I: Basic Concepts*, World Scientific Publishing, New Jersey, 2004.

[17] Justin Mullins, "Making Unbreakable Code", IEEE Spectrum, May 2002.

[18] A.M. Steane, E.G. Rieffel, "Beyond Bits: The Future of Quantum Information Processing", IEEE Computer, January 2000.

[19] William Stallings, *Cryptography and Network Security Principles and Practices*, Prentice Hall, November 2005.

[20] A. J. Menezes, S. A. Vanstone, and P. C. Van Oorschot, *Handbook of Applied Cryptography*, CRC Press, Inc., 2004