Quantum Forgery Attacks on COPA, AES-COPA and Marble Authenticated Encryption Algorithms

Yinsong Xu · Wenjie Liu* · Wenbin Yu

Abstract Since the classic forgery attacks on COPA, AES-COPA and Marble authenticated encryption algorithms need to query about $2^{n/2}$ times and their success probability are not high. To solve this problem, the corresponding quantum forgery attacks on COPA, AES-COPA and Marble authenticated encryption algorithms are presented. In the quantum forgery attacks on COPA and AES-COPA, we use Simon's algorithm to find the period of the tag generation function in COPA and AES-COPA by querying in superposition, and then generate a forged tag for a new message. While in the quantum forgery attack on Marble, Simon's algorithm is used to recover the secret parameter L, and the forged tag can be computed with L. Compared with classic forgery attacks on COPA, AES-COPA and Marble, our attack can reduce the number of queries from $O(2^{n/2})$ to $O(n)$ and improve success probability close to 100%.

Y. Xu

School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, P. R. China
E-mail: mugongxys@foxmail.com

W. Liu
*Corresponding Author
Engineering Research Center of Digital Forensics, Ministry of Education, Nanjing 210044, P. R. China
School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, P. R. China
E-mail: wenjiel@163.com

W. Yu
Engineering Research Center of Digital Forensics, Ministry of Education, Nanjing 210044, P. R. China
School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, P. R. China
E-mail: ywb1518@126.com

1 Introduction

In symmetric cryptography, an authenticated encryption algorithm is an algorithm that transforms an arbitrary-length data stream, called a message or plaintext, into another data stream of the same length, called a ciphertext, and generates an authentication tag for the message at the same time, under the control of a secret key [1]. The purpose of authenticated encryption algorithms is to provide data privacy and integrity. In 2013, the CAESAR competition [2] was launched to provide good authenticated encryption schemes as better alternatives to current options such as AES-GCM [3], which have received 57 submissions in first round.

The COPA authenticated encryption algorithm [4] is designed by Andreeva *et al.*, which combines OCB's offsets with an internal dependency chain in order to achieve some security in the case of nonce repetition. Since its birth is earlier than the CAESAR competition, it did not participate in the competition. But, its instantiation with the AES [7] block cipher under 128 key bits (called AES-COPA [5,6]) has been a CAESAR candidate. The Marble authenticated encryption algorithm (v1.0/1.1/1.2) [8–10] is also a CAESAR submission by Jian Guo inspired by COPA, which uses two internal chains to prevent birthday attacks on the internal chain and uses reduced-round AES as building blocks. So far, only AES-COPA is a CAESAR candidate.

Although the COPA designers proved that it has a birthday-bound security on integrity (which is mainly associated with existential forgery) under the assumption that the underlying block cipher is a strong pseudorandom permutation. And the Marble designer claimed that Marble achieved a full security beyond the birthday-bound due to the choice of $TRANS$ function (see Table 1). However, Nandi [11] presented an existential forgery attack on the case of COPA that processes fractional messages, which produce the correct ciphertext and tag for an unspecified message whose ciphertext and tag are not given. Later, Lu [12] also proposed an almost universal forgery attack on the COPA and Marble, which produce the correct ciphertext and tag for any specified message whose ciphertext and tag are not given. Note that almost universal forgery attack allows the forger to make a minimal change from the given message M to a modified one M' by replacing some of its blocks before producing its tag [13]. Both of Nandi's and Lu's forgery attack on the COPA indicate that the probability of a forgery is much larger than $2/2^n$ when the number of queries q is close to birthday-bound $2^{n/2}$ and even smaller than $2^{n/2}$.

Moreover, Lu's forgery attack on the marble also indicates that the Marble (v1.0/1.1/1.2) are incorrectly far overestimated in the sense of full security. The probability of a forgery would be 32% when the number of queries q is close to birthday-bound $2^{n/2}$. Due to that Lu's forgery attack on the Marble

do not use associated data, Fuhr *et al.*'s forgery attack [14] make up for this consideration. Finally, for AES-COPA with nonce, Lu's forgery attack requires slightly less than 2^{63} encryption queries and its success probability is about 6%.

On the other hand, in the quantum world, many quantum algorithms are constantly being used in cryptanalysis [15, 24, 25], machine learning [16–18], blockchain [19, 20] and so on. Since Shor's algorithm [15] was proposed, it has been announced that quantum computers would be a severe threat for public key cryptography. More and more researchers have began to use quantum algorithms to break symmetric cryptosystems, such as Simon's algorithm [21, 24–27], Grover algorithm [22, 31], Bernstein-Vazirani algorithm [23, 32] and so on. In addition, they also have proposed some new quantum algorithms [30, 33, 34], and even extended classical cryptanalysis methods to the quantum field [35, 36]. Among them, Simon's algorithm was first used to break the 3-round Feistel construction [24] and then to prove that the Even-Mansour construction [25] is insecure with superposition queries. Inspired from them, Kaplan *et al.* [26] show that several classical attacks based on finding collisions can be dramatically sped up using Simon's algorithm. And Shi *et al.* [27] also use analogous way to implement collision attacks on authenticated encryption AEZ from CAESAR competition. To improve the efficiency of classic forgery attack on COPA, AES-COPA and Marble, we present quantum forgery attacks based on Simon's algorithm. Note that, the attack of Ref. [23–27, 30–36] and our forgery attack all belong to the Q2 model proposed by Kaplan [29]. In Q2 model, the adversary is allowed to perform quantum superposition queries to a remote quantum cryptographic oracle [30]. The opposite of Q2 model is Q1 model, i.e., the adversary can query a quantum random oracle with arbitrary superpositions of the inputs, but is only able to make classical queries to a classical encryption oracle. Therefore, this model is not considered in this article.

Our contributions. In this paper, we use Simon's algorithm to find the period of the function of tag generation in COPA firstly. When we get the period, we can compute the forged message for the same tag. Since the length of the associated data block and the message block will affect the period value, there will be a small difference in the process of applying Simon's algorithm (see Sect. 3). Secondly, the encryption of AES-COPA is analogous to COPA, an additional (public) input parameter called nonce is appended to associated data (if any), and then the resulting value is treated as associated data in COPA. Therefore, our quantum forgery attack on AES-COPA is similar on COPA. Moreover, due to the tag generation has 3-round encryption with different keys and the function $TRANS$, it is difficult to find the period of tag by Simon's algorithm. But, we use Simon's algorithm to find the period of internal state S_1, and then obtain the secret parameter L through the got period. Finally, we can use the secret parameter L to compute forged tag and message pairs. In summary, the query times and the success probability of our quantum forgery attack are mainly reflected in the number of executing Simon's algorithm and the success probability of finding period. That is, our

quantum forgery attack only needs $O(n)$ queries with high success probability (the probability is close to 1).

This paper is organized as follows. Sect. 2 provides a brief description of the COPA, AES-COPA, Marble authenticated encryption algorithms and Simon's algorithm. And our quantum forgery attacks on COPA, AES-COPA and Marble authenticated encryption algorithms are shown in Sects. 3, 4 and 5, respectively. Then, the comparison with other forgery attacks on COPA, AES-COPA and Marble is analyzed in Sect. 6, followed by a discussion and conclusion in Sect. 7.

2 Preliminaries

In this section, we would briefly describe the COPA, AES-COPA, Marble authenticated encryption algorithms and Simon's algorithm.

2.1 COPA, AES-COPA and Marble Authenticated Encryption Algorithms

For the sake of clarity, we list some explanations for variables and notations in Table 1, which are frequently used in COPA and Marble authenticated encryption algorithms. The COPA authenticated encryption algorithm [4] was published in 2013. Its internal state, key and tag have the same length. Therefore, in order to facilitate following analysis, we default the length of them to 128 bits. To generate ciphertexts and tag, it has three phases: processing associated data, message encryption, and tag generation, which is shown in Fig. 1. During the process of processing associated data, if there is no associated data, then we set $V \overset{def}{=} 0$. Besides, if the last block $A[a]$ or $M[d]$ are not a multiple of n bits, they need to be padded by a one and as many zeroes as necessary to obtain a multiple of the block size n, i.e., $A[a]10^*$ and $M[d]10^*$. Finally, decryption is the inverse of encryption, and tag verification is identical to tag generation. Please refer to [4] for the specification of COPA.

AES-COPA authenticated encryption algorithm is an extended version of COPA, with some differences. First, a public message number N called nonce is appended to associated data, like $A[1]||A[2]||\cdots||A[a]||N$, and as part of associated data. Besides, AES-COPA can accept "fractional" messages M, i.e., the length $|M|$ is not necessarily a positive multiple of the block size n. And AES-COPA has two versions v.1 [5] and v.2 [6], where the process of fractional message encryption in v.1 is slightly different from v.2 (as shown in Table 2). For simplicity, we roughly introduce the encryption process of these two versions. Note that $XLS_d()$ is invertible in v.1.

The Marble authenticated encryption algorithms [8–10] are like COPA, which has four phases: initialization, processing associate data, message encryption, and tag generation. Fig. 2 illustrates the message encryption and tag generation phase of newest version (i.e. v1.2) of Marble. Its decryption is the inverse of encryption, and tag verification is identical to tag generation. Please

Table 1: Variables and notations

Variables and notations	Explanations
$A[1]\|\|A[2]\|\|\cdots\|\|A[a]$, $M[1]\|\|M[2]\|\|\cdots\|\|M[d]$	$A[1]\|\|A[2]\|\|\cdots\|\|A[a]$ and $M[1]\|\|M[2]\|\|\cdots\|\|M[d]$ are represented as associated data of a n-bit blocks and messages of d n-bit blocks, respectively, where "$\|\|$" is bit connection and n generally defaults to 128.
$C[1], C[2], \cdots, C[d]; T$	$C[1], C[2], \cdots, C[d]$ and T are the ciphertext and the tag for $M[1]\|\|M[2]\|\|\cdots\|\|M[d]$, repestively.
$\|A\|$	$\|A\|$ represents the number of bits in A.
S, S_1, S_2	S, S_1 and S_2 are n-bit ($n = 128$) internal states.
$E_k(), L$	$E_k()$ is an n-bit block cipher, i.e., $E : k \times \{0,1\}^n \rightarrow \{0,1\}^n$, where the key k generally consists of 128 bit. And $L \stackrel{def}{=} E_k(0)$ in COPA.
$+, \oplus$	"$+$" or "$\oplus$" are bitwise logical exclusive (XOR) operation.
$\cdot$	"$\cdot$" represents polynomial multiplication modulo the polynomial $x^{128} + x^7 + x^2 + x + 1$ in $\mathrm{GF}(2^{128})$. We can abbreviate $A \cdot B$ as AB.
E_1, E_2, E_3	Each of the operations E_1, E_2 and E_3 is a 4-round reduced version of the AES [29] block cipher, with four fixed round subkeys chosen from the eleven round subkeys of the AES with 128 key bits.
⧈	⧈ (see Fig. 2) represents a function $TRANS(x,y) = (x + y, 3x + y)$, where x and y are 128-bit inputs.
$Const_0$, $Const_1$, $Const_2$	$Const_0$, $Const_1$ and $Const_2$ are three 128-bit constants.
τ	τ is 128-bit secret parameters.

refer to [8–10] for the specification of Marble. The main differences between the newest version of Marble and the other two versions are as follows: In the second version (i.e. v1.1 [9]), the mask parameter before E_1 is $2^{a-1} \cdot 3^2 \cdot L$ for the last block of associated data; and in the initial version (i.e. v1.0 [8]), the mask parameter before E_1 is $2^{a-1} \cdot 3^3 \cdot L$ for the last block of associated data if it is full, and is $2^{a-1} \cdot 3^4 \cdot L$ if it is not full. Thus, the latest version of Marble is identical to the initial version when the last block of associated data is full.

2.2 Simon's Algorithm

Simon's algorithm was proposed by Daniel R. Simon [21] in 1997, which is a quantum algorithm to solve Simon's problem (also proposed by Daniel R. Simon). The definition of Simon's problem is presented as below.

Simon's Problem: Given a Boolean function $f : \{0,1\}^n \rightarrow \{0,1\}^n$ and the promise that there exists $s \in \{0,1\}^n$ such that for any $(x,y) \in \{0,1\}^n$, $[f(x) = f(y)] \Leftrightarrow [x \oplus y \in \{0^n, s\}]$, the goal is to find s.

This problem can be solved classically by searching for collisions. The optimal time to solve it is therefore $\Theta(2^{n/2})$. On the other hand, Simon's algorithm solves this problem with quantum complexity $O(n)$. Note that to run

Table 2: Two versions of AES-COPA

AES-COPA v.1-ENCRYPT:	AES-COPA v.2-ENCRYPT:
if $d \geqslant 2$ and $1 \leqslant |M[d]| \leqslant n-1$ **then**	$M = M[1]||M[2]||\cdots||M'[d]$
$V \leftarrow$ Processing associated data $A||N$ (see Fig. 1)	$M[d] \leftarrow M'[d]$ **if** $|M'[d]| = n$ **else** $M'[d]|10^*$
$(C', S') \leftarrow$ Message encryption $(V, M[1]||M[2]||\cdots||M[d-1])$ (see Fig. 1)	$P \leftarrow 0$ **if** $|M'[d]| = n$ **else** 1
$\Sigma' \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus M[d-1]$	$V \leftarrow$ Processing associated data $A||N$ (see Fig. 1)
$T' \leftarrow E_k(E_k(\Sigma' \oplus 2^{d-2}3^2L) \oplus S') \oplus 2^{d-2}7L$	$(C, S) \leftarrow$ Message encryption $(V, M[1]||M[2]||\cdots||M[d], P)$ (see Fig. 1)
$C[d]T \leftarrow XLS_d(M[d]T')$	$\Sigma \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus M[d]$
$C \leftarrow C'C[d]$	$T \leftarrow E_k(E_k(\Sigma \oplus 2^{d-1}3^27^PL) \oplus S) \oplus 2^d7L$
Output (C,T)	Output (C,T)
end if	

Simon's algorithm, it is required that the function f can be queried quantum-mechanically.

In Simon's algorithm (as shown in Fig. 3), we need to prepare a $2n$-qubit state $|0\rangle^{\otimes n}|0\rangle^{\otimes n}$ and apply Hadamard transform $H^{\otimes n}$ to the first n qubits to obtain the quantum superposition $\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle |0\rangle^{\otimes n}$. Then, the quantum superposition would be input into the function f to get the state $\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle |f(x)\rangle$. After that, we apply Hadamard transform $H^{\otimes n}$ to the first n qubits again to get $\frac{1}{2^n} \sum_{x \in \{0,1\}^n} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle |f(x)\rangle$. Finally, we perform measurements on all qubits, where the vector y (measured from the first n qubits) be orthogonal to s, i.e., $y \cdot s = 0$. By repeating this subroutine $O(n)$ times, one obtains $n-1$ independent vectors orthogonal to s with high probability, and s can be recovered using basic linear algebra.

3 Quantum Forgery Attacks on COPA by Simon's Algorithm

3.1 Attack Strategy

Due to that our forgery attack belongs to Q2 model, the adversary can be able to access the quantum cryptographic oracle and queries in superposition. To execute Simon's algorithm, the quantum cryptographic oracle would be used as quantum oracle Q_f (also as the function f in Simon's problem) in the circuit of Simon's algorithm. According to the specific situation, we select the associated data or message as the input of Simon's algorithm, and select the tag or other data (like S_1 in Marble) as the algorithm's output. By repeatedly executing Simon's algorithm $O(n)$ times, we can obtain the corresponding period value.
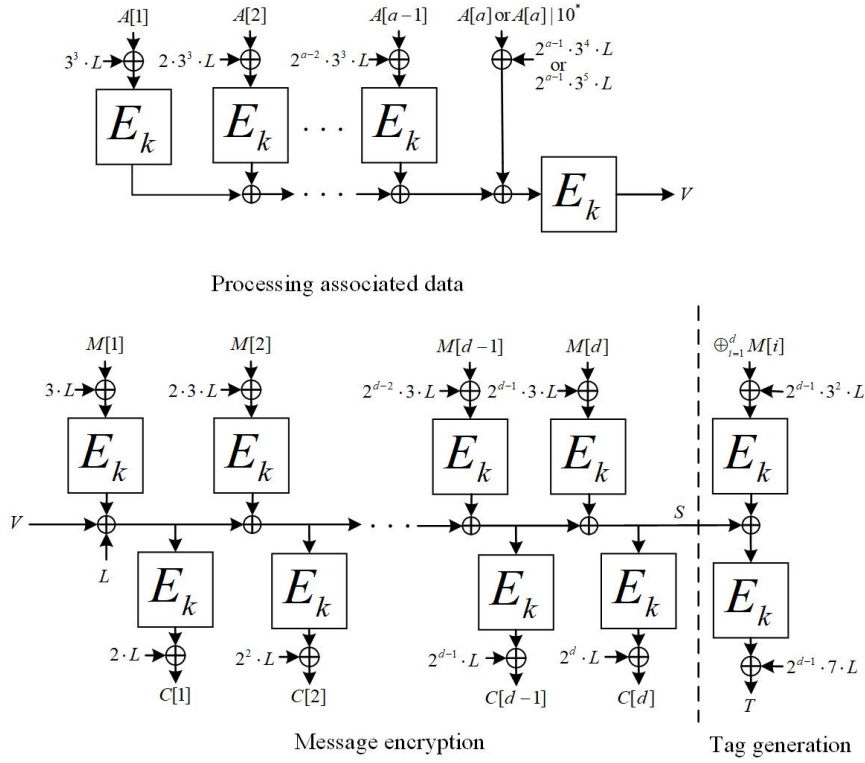
Fig. 1: The process of processing associated data, message encryption and tag generation in COPA.

Finally, we can get the collision of the tag with different associated data or messages by the period value. The entire attack process is called as a quantum forgery attack.

Since COPA can set the associated data to 0, we conduct quantum forgery attacks from two cases: without associated data and with associated data, which is demonstrated as below.

3.2 Quantum Forgery Attacks on COPA without Associated Data

Since there is no associated data, so $V = 0$ and $T = E_k(E_k(\Sigma \oplus 2^{d-1}3^2L) \oplus S) \oplus 2^{d-1}7L$, where $\Sigma = M[1] \oplus \cdots \oplus M[d]$. We can see that the size of the message length d affects the period value s of the function T. Therefore, we will calculate the period s of T from 2 cases: $d = 1$ and $d \geqslant 2$.

Case 1: When $d = 1$, i.e., message $M = M[1]$. Then,

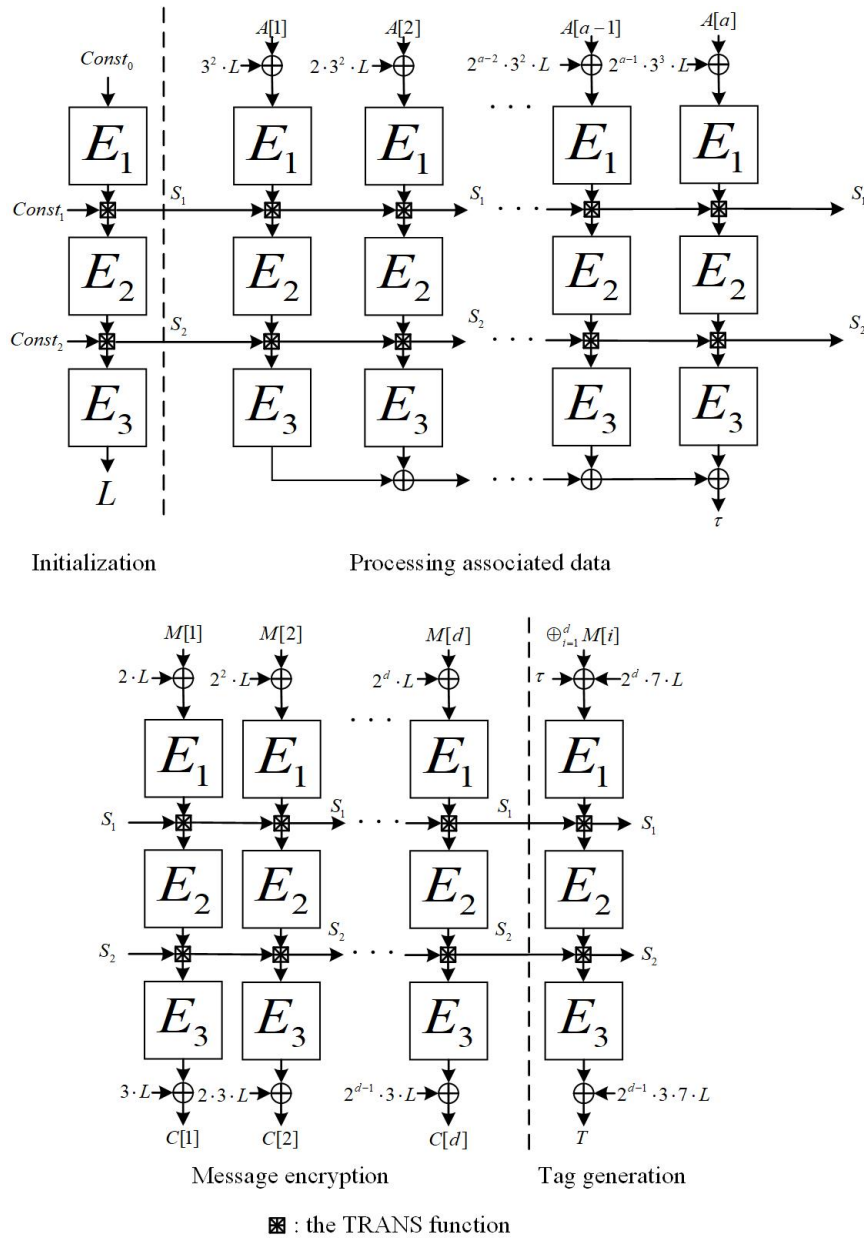$$T = E_k(E_k(M[1] \oplus 3^2L) \oplus S) \oplus 7L, \tag{1}$$

Fig. 2: The process of initialization, processing associated data, message encryption and tag generation in Marble.
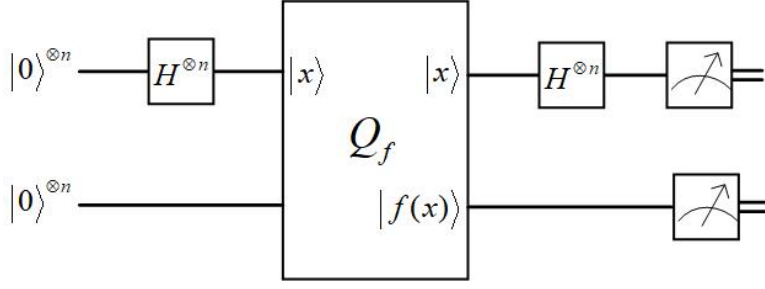
Fig. 3: The circuit of Simon's algorithm.

where $S = E_k(M[1] \oplus 3L) \oplus V \oplus L = E_k(M[1] \oplus 3L) \oplus L$ and $3^2 = 3 \cdot 3 = (x+1)(x+1) = (x^2 + x + x + 1) mod(x^{128} + x^7 + x^2 + x + 1) = x^2 + 1 = 5$. Note that, the multiplication and addition operations in this paper are the addition and multiplication operations in GF(2^{128}). So,

$$T = E_k(E_k(M[1] \oplus 5L) \oplus E_k(M[1] \oplus 3L) \oplus L) \oplus 7L. \qquad (2)$$

Firstly, we define the following function:

$$f : \{0,1\}^n \to \{0,1\}^n$$
$$x \to Tag\_COPA(x) = E_k(E_k(x \oplus 5L) \oplus E_k(x \oplus 3L) \oplus L) \oplus 7L. \qquad (3)$$

The function f can be computed with a single call to the cryptographic oracle, and we can build a quantum circuit for f given a quantum oracle for COPA (as shown in Fig. 4). Moreover, f satisfies the requirement of Simon's problem with the period $s = 6L$:

$$f(x) = E_k(E_k(x \oplus 5L) \oplus E_k(x \oplus 3L) \oplus L) \oplus 7L,$$
$$f(x \oplus s) = E_k(E_k(x \oplus s \oplus 5L) \oplus E_k(x \oplus s \oplus 3L) \oplus L) \oplus 7L$$
$$= E_k(E_k(x \oplus 6L \oplus 5L) \oplus E_k(x \oplus 6L \oplus 3L) \oplus L) \oplus 7L$$
$$= E_k(E_k(x \oplus 3L) \oplus E_k(x \oplus 5L) \oplus L) \oplus 7L$$
$$= f(x).$$

where $6L \oplus 5L = (6 \oplus 5) \cdot L = (x^2 + x + x^2 + 1) \cdot L = (x+1) \cdot L = 3L$ and $6L \oplus 3L = 5L$.

Therefore, the tag of an arbitrary block $M[1]$ is valid for $M[1] \oplus 6L$.

Case 2. When $d \geqslant 2$, i.e., message $M = M[1]||M[2]||\cdots||M[d]$. Then,

$$T = E_k(E_k(M[1] \oplus M[2] \oplus \cdots \oplus M[d] \oplus 2^{d-1}3^2) \oplus S) \oplus 2^{d-1}7L, \qquad (4)$$

where

$$S = E_k(M[d] \oplus 2^{d-1}3L) \oplus \cdots \oplus E_k(M[2] \oplus 6L) \oplus E_k(M[1] \oplus 3L) \oplus L. \qquad (5)$$
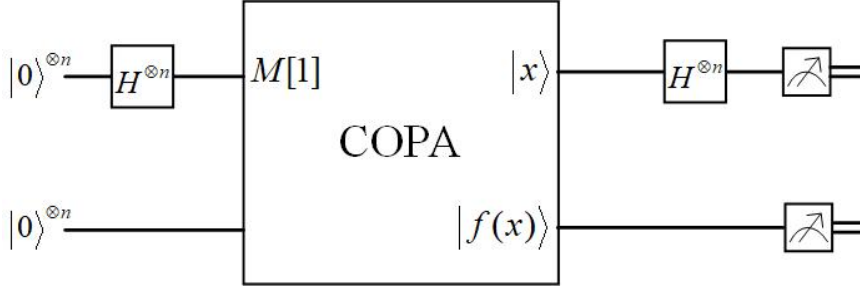
Fig. 4: The circuit of quantum forgery attack on COPA without associated data ($d = 1$).

By observing Eqs. (4) and (5), we find that if we find the period s which meets $M[1] \oplus s \oplus 3L = M[2] \oplus 6L$, then the tag will not change for different messages $M = M[1]||M[2]||\cdots||M[d]$ and $M' = M[1]\oplus s||M[2]\oplus s||\cdots||M[d]$, i.e.,

$$
\begin{aligned}
S' &= E_k(M[d] \oplus 2^{d-1}3L) \oplus \cdots \oplus E_k(M[2] \oplus s \oplus 6L) \oplus E_k(M[1] \oplus s \oplus 3L) \\
&\quad \oplus L \\
&= E_k(M[d] \oplus 2^{d-1}3L) \oplus \cdots \oplus E_k(M[1] \oplus 3L) \oplus E_k(M[2] \oplus 6L) \oplus L \\
&= S, \\
T' &= E_k(E_k(M[1] \oplus s \oplus M[2] \oplus s \oplus \cdots \oplus M[d] \oplus 2^{d-1}3^2) \oplus S') \oplus 2^{d-1}7L \\
&= E_k(E_k(M[1] \oplus M[2] \oplus \cdots \oplus M[d] \oplus 2^{d-1}3^2) \oplus S) \oplus 2^{d-1}7L \\
&= T.
\end{aligned}
$$

Therefore, we only need to intercept the first two message blocks $M[1]||M[2]$ as input to the function f, and define the function as below:

$$
\begin{aligned}
f : \{0,1\}^n &\to \{0,1\}^n \\
x &\to Tag\_COPA(x||x \oplus \sigma) = \\
&E_k(E_k(\sigma \oplus 10L) \oplus E_k(x \oplus \sigma \oplus 6L) \oplus E_k(x \oplus 3L) \oplus L) \oplus 14L,
\end{aligned}
\tag{6}
$$

where $10 = 2^{d-1} \cdot 3^2 = 2 \cdot 3^2$, $\sigma = M[1] \oplus M[2]$ and σ can be viewed as an arbitrary constant. It is obvious to see that $f(x) = f(x \oplus s)$ with $s = \sigma \oplus 5L$:

$$
\begin{aligned}
f(x) &= E_k(E_k(\sigma \oplus 10L) \oplus E_k(x \oplus \sigma \oplus 6L) \oplus E_k(x \oplus 3L) \oplus L) \oplus 14L \\
f(x \oplus s) &= E_k(E_k(\sigma \oplus 10L) \oplus E_k(x \oplus s \oplus \sigma \oplus 6L) \oplus E_k(x \oplus s \oplus 3L) \oplus L) \\
&\quad \oplus 14L \\
&= E_k(E_k(\sigma \oplus 10L) \oplus E_k(x \oplus \sigma \oplus 5L \oplus \sigma \oplus 6L) \oplus E_k(x \oplus \sigma \oplus 5L \\
&\quad \oplus 3L) \oplus L) \oplus 14L \\
&= E_k(E_k(\sigma \oplus 10L) \oplus E_k(x \oplus 3L) \oplus E_k(x \oplus \sigma \oplus 6L) \oplus L) \oplus 14L \\
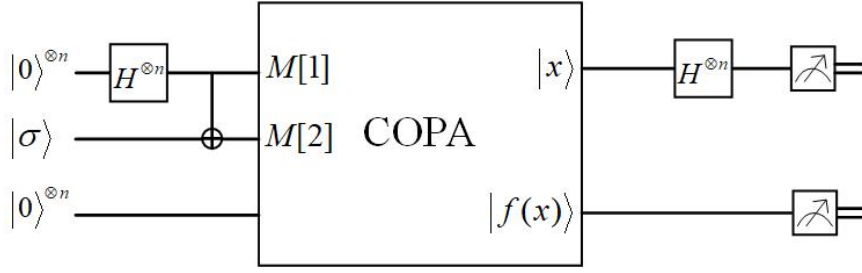&= f(x)
\end{aligned}
$$

Fig. 5: The circuit of quantum forgery attack on COPA without associated data ($d = 2$).

Therefore, we can apply Simon algorithm on this function f (i.e., the COPA cryptographic oracle), to get period $s = \sigma \oplus 5L$ (as shown in Fig. 5). Finally, we query the tag of $M = M[1]||M[2]||\cdots||M[d]$, and the same tag is valid for $M' = M[2] \oplus 5L||M[1] \oplus 5L||\cdots||M[d]$.

3.3 Quantum Forgery Attacks on COPA with Associated Data

Due to the existence of associated data, $V = V(A[1]||A[2]||\cdots||A[a])$ and $T = E_k(E_k(\Sigma \oplus 2^{d-1}3^2 L) \oplus E_k(M[d] \oplus 2^{d-1}3L) \oplus \cdots \oplus E_k(M[1] \oplus 3L) \oplus V \oplus L) \oplus 2^{d-1}7L$. Through the control variable method, we can find that as long as the values of V and V' calculated from two different associated data (A and A', $A \neq A'$) are equal, i.e., $V = V'$, the corresponding tags T and T' with two same constant messages are also equal to each other. So, we can calculate the period s of the function T with the constant message and variable associated data, which is also the period of the function V. And by observing the process of processing associated data, we find that whether $|A|$ is a multiple of n will affect the process of calculating V, and $a, d \geqslant 2$. Therefore, We would consider the following three cases: 1) $|A[a]| \% n = 0$, $a = d = 2$; 2) $|A[a]| \% n \neq 0$, $a = d = 2$; 3) $a, b > 2$.
Case 1: When $|A[a]| \% n = 0$ and $a = d = 2$, i.e., $A = A[1]||A[2]$ and $M = M[1]||M[2]$. Then,

$$T = E_k(E_k(M[1] \oplus M[2] \oplus 10L) \oplus E_k(M[2] \oplus 6L) \oplus E_k(M[1] \oplus 3L) \\ \oplus E_k(E_k(A[1] \oplus 3^3 L) \oplus A[2] \oplus 2 \cdot 3^4 L) \oplus L) \oplus 14L. \quad (7)$$

In order to prevent the message $M[1]||M[2]$ from affecting the period s of the function T (which is also the period of function V), we set the value of $M[1]||M[2]$ to an arbitrary constant $m||m$. $E_k(M[1] \oplus M[2] \oplus 10L) \oplus E_k(M[2] \oplus 6L) \oplus E_k(M[1] \oplus 3L)$ would be a constant, which is abbreviated as $c_{m||m}$. And Eq. (7) can be abbreviated as below.

$$T = E_k(E_k(E_k(A[1] \oplus 3^3 L) \oplus A[2] \oplus 2 \cdot 3^4 L) \oplus c_{m||m} \oplus L) \oplus 14L. \quad (8)$$
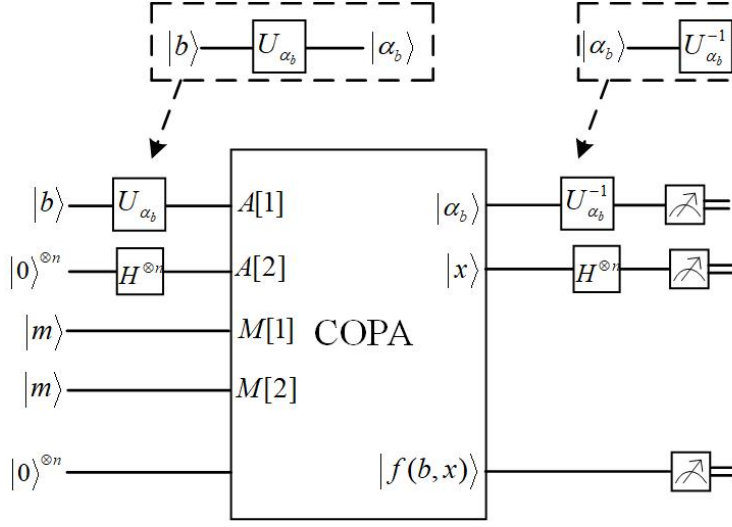
Fig. 6: The circuit of quantum forgery attack on COPA with associated data $(|A[a]| \% n = 0, a = d = 2)$.

We firstly fix two arbitrary associated data blocks α_0 and α_1 (one of them is $A[1]$ and $\alpha_0 \neq \alpha_1$), and define the following function:

$$f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$
$$b, x \to Tag\_COPA(\alpha_b||x, m||m) =$$
$$E_k(E_k(E_k(\alpha_b \oplus 3^3 L) \oplus x \oplus 2 \cdot 3^4 L) \oplus c_{m||m} \oplus L) \oplus 14L, \tag{9}$$

where $3^3 = (x+1)^3 mod(x^{128} + x^7 + x^2 + x + 1) = (x^3 + x^2 + x + 1)mod(x^{128} + x^7 + x^2 + x + 1) = 15$.

Then, we apply Simon's algorithm on function f (as shown in Fig. 6). The function f has the period $s = 1||E_k(\alpha_0 \oplus 15L) \oplus E_k(\alpha_1 \oplus 15L)$:

$$f(1, x \oplus s) = E_k(E_k(E_k(\alpha_1 \oplus 15L) \oplus x \oplus E_k(\alpha_0 \oplus 15L)$$
$$\oplus E_k(\alpha_1 \oplus 15L) \oplus 2 \cdot 3^4 L) \oplus c_{m||m} \oplus L) \oplus 14L$$
$$= E_k(E_k(x \oplus E_k(\alpha_0 \oplus 15L) \oplus 2 \cdot 3^4 L) \oplus c_{m||m} \oplus L) \oplus 14L$$
$$= f(0, x).$$

When we have got the period of function V (which is also the period of T and f in Eqs. (8) and (9)) with variable associated data and constant message, we can repeat the attack process in Case 2 of Sect. 3.2 to get another period with different message and no associated data. Finally, the tag of $\alpha_0||A[2]$ and $M[1]||M[2]$ is as same as $\alpha_1||A[2] \oplus E_k(\alpha_0 \oplus 15L) \oplus E_k(\alpha_1 \oplus 15L)$ and $M[2] \oplus 5L||M[1] \oplus 5L$'s.

Case 2: $|A[a]|\,\%n \neq 0$ and $a = d = 2$, i.e., $A = A[1]||A[2]||10^*$ and $M = M[1]||M[2]$. Similar to Case 1 of Sect. 3.3, we set the value of $M[1]||M[2]$ to an arbitrary constant $m||m$. Then,

$$T = E_k(E_k(E_k(A[1] \oplus 3^3L) \oplus A[2]||10^* \oplus 2 \cdot 3^5L) \oplus c_{m||m} \oplus L) \oplus 14L. \quad (10)$$

And we define the following function:

$$f : \{0,1\} \times \{0,1\}^n \rightarrow \{0,1\}^n$$
$$b, x \rightarrow Tag\_COPA(\alpha_b||x, m||m) =$$
$$E_k(E_k(E_k(\alpha_b \oplus 3^3L) \oplus x \oplus 2 \cdot 3^5L) \oplus c_{m||m} \oplus L) \oplus 14L. \quad (11)$$

Then, we perform Simon's algorithm with this function f (as same as the circuit in Fig. 6). Its period is as same as the one in Case 1 of Sect. 3.2, i.e., $s = 1||E_k(\alpha_0 \oplus 15L) \oplus E_k(\alpha_1 \oplus 15L)$. Finally, the tag of $\alpha_0||A[2]||10^*$ and $M[1]||M[2]$ is equal to $\alpha_1||A[2]||10^* \oplus E_k(\alpha_0 \oplus 15L) \oplus E_k(\alpha_1 \oplus 15L)$ and $M[2] \oplus 5L||M[1] \oplus 5L$'s.

Case 3: When $a, d > 2$, i.e., $A = A[1]||A[2]||\cdots||A[a]$ or $A[1]||A[2]||\cdots||A[a]||10^*$, and $M = M[1]||M[2]||\cdots||M[d]$. Then,

$$T = E_k(E_k(\Sigma \oplus 2^{d-1}3^2L) \oplus S \oplus E_k(E_k(A[1] \oplus 3^3L) \oplus A[2] \oplus \cdots \oplus A[a]$$
$$\oplus 2 \cdot 3^4L) \oplus L) \oplus 14L,$$
$$or,$$
$$= E_k(E_k(\Sigma \oplus 2^{d-1}3^2L) \oplus S \oplus E_k(E_k(A[1] \oplus 3^3L) \oplus A[2] \oplus \cdots \oplus A[a]||10^*$$
$$\oplus 2 \cdot 3^5L) \oplus L) \oplus 14L. \quad (12)$$

Similar to the Case 2 in Sect. 3.2, we only need to consider the case of $a = 3$ to find the period of the function V, so $V(A[1]||A[2]||A[3]||\cdots||A[a]) = V'(A[1] \oplus s||A[2] \oplus s||A[3]||\cdots||A[a])$.

We firstly need to calculate the period of T with variable associated data and constant message like Case 1 in Sect. 3.2, and define the following function:

$$f : \{0,1\}^n \rightarrow \{0,1\}^n$$
$$x \rightarrow Tag\_COPA(x||x \oplus \sigma||A[3] \, or \, x||x \oplus \sigma||A[3]||10^*, m||m||m)$$
$$= E_k(c_{m||m||m} \oplus E_k(E_k(x \oplus 3^3L) \oplus E_k(x \oplus \sigma \oplus 2 \cdot 3^3L) \oplus A[3]$$
$$\oplus 2^2 \cdot 3^4L) \oplus L) \oplus 14L$$
$$or,$$
$$= E_k(c_{m||m||m} \oplus E_k(E_k(x \oplus 3^3L) \oplus E_k(x \oplus \sigma \oplus 2 \cdot 3^3L) \oplus$$
$$A[3]||10^* \oplus 2^2 \cdot 3^5L) \oplus L) \oplus 14L. \quad (13)$$

where $c_{m||m||m} = E_k(\Sigma \oplus 2^{d-1}3^2L) \oplus S$ and $\sigma = A[1] \oplus A[2]$. The circuit of quantum forgery attack is shown in Fig. 7. We can find that its period is $s = \sigma \oplus 17L$.
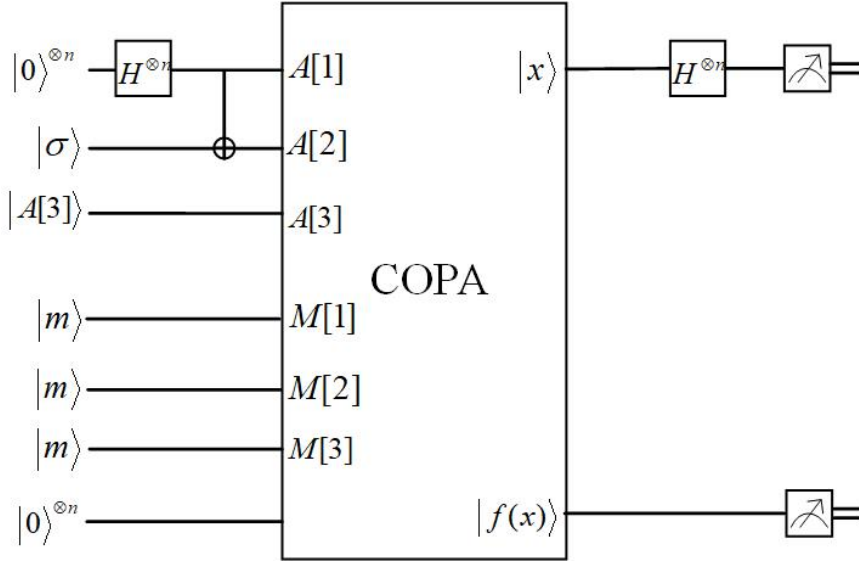
Fig. 7: The circuit of quantum forgery attack on COPA with associated data $(a, b > 2)$.

Finally, the tag of $A[1]||A[2]|| \cdots ||A[a]$ *or* $A[a]|10^*$ and $M[1]||M[2]|| \cdots ||M[d]$ is identical to $A[2] \oplus 17L||A[1] \oplus 17L|| \cdots ||A[a]$ *or* $A[a]|10^*$ and $M[2] \oplus 5L||M[2] \oplus 5L|| \cdots ||M[d]$'s.

From the above quantum forgery attacks on COPA, we found that a quantum superposition query on encrypted Oracle can get all the tags generated by the input plaintext, and then get the orthogonal value of the hidden period through Simon's algorithm. In order to completely compute the hidden period, it is necessary to repeat Simon's algorithm $O(n)$ times. And each time Simon's algorithm is executed, the quantum superposition query needs to be performed again. Therefore, the number of queries for the entire quantum forgery attack is the number of repeated executions of Simon's algorithm. And the success probability of quantum forgery attacks is equivalent to the probability of successfully finding the hidden period through Simon's algorithm.

4 Quantum Forgery Attacks on AES-COPA by Simon's Algorithm

Compared with COPA, AES-COPA has more Nonce (a public message number N) as input. And it would participate in the process of processing associated data as part of the associated data, i.e., $X[1]||X[2]|| \cdots ||X[x] = A[1]||A[2]|| \cdots ||A[a]||N$. The rule of processing $X[1]||X[2]|| \cdots ||X[x]$ is as same as the process of processing associated data in COPA. Therefore, the quantum forgery attacks on COPA can also be applied on AES-COPA. Moreover, AES-COPA can accept "fractional" messages M, i.e., the length $|M|$ is not
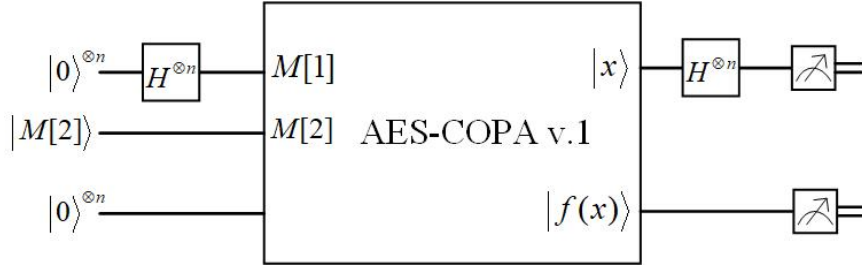
Fig. 8: The circuit of quantum forgery attack on AES-COPA v.1 ($d = 2$, $|M[2]|\,\%n \neq 0$).

necessarily a positive multiple of the block size n. Therefore, we focus on how to implement quantum forgery attack on AES-COPA (v.1 and v.2) with fractional messages in different cases.

4.1 Quantum Forgery Attacks on AES-COPA v.1

Firstly, assume that we have $d \geqslant 2$ and $1 \leqslant |M[d]| \leqslant n - 1$. Since $XLS_d()$ is invertible, $XLS_d(M[d]T') \neq XLS_d(M[d]^*T'^*)$ for any $M[d] \neq M[d]^*$ and $T' \neq T'^*$. Therefore, we only to find the period of T' for different messages $M[1]||M[2]||\cdots||M[d-1]$.

For the sake of simplicity, we temporarily set the associated data and Nonce as fixed constants. So, V is a fixed constant, too. And we consider two cases: 1) $d = 2$, 2) $d > 2$.

Case 1: When $d = 2$, $|M[2]|\,\%n \neq 0$. Then,

$$T' = E_k(E_k(M[1] \oplus 5L) \oplus S') \oplus 7L \tag{14}$$

where $S' = E_k(M[1] \oplus 3L) \oplus V \oplus L$. So,

$$T' = E_k(E_k(M[1] \oplus 5L) \oplus E_k(M[1] \oplus 3L) \oplus L \oplus V) \oplus 7L \tag{15}$$

We can define the following function:

$$f : \{0,1\}^n \to \{0,1\}^n$$
$$x \to Tag\_AES - COPA\_v.1(x) = XLS_d(E_k(E_k(x \oplus 5L) \oplus E_k(x \oplus 3L)$$
$$\oplus L \oplus V) \oplus 7L). \tag{16}$$

Through this function, the circuit of our attack on this case is similar to Case 1 in Sect. 3.2 (as shown in Fig. 8). The period of this function is $s = 6L$. Finally, the tag of $M[1]||M[2]$ is as same as $M[1] \oplus 6L||M[2]$ with same associated data and Nonce. If you want to find two different associated data and Nonce, you can refer to Sect. 3.3.
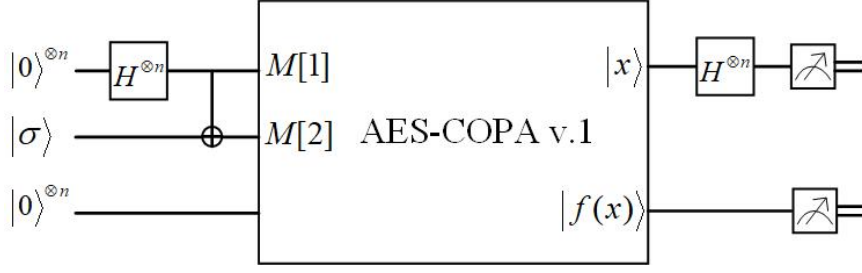
Fig. 9: The circuit of quantum forgery attack on AES-COPA v.1 ($d > 2$, $|M[2]| \% n \neq 0$).

Case 2: When $d > 2$, $|M[d]| \% n \neq 0$. Then,

$$T' = E_k(E_k(M[1] \oplus M[2] \oplus \cdots \oplus M[d-1] \oplus 2^{d-2}3^2 L) \oplus S') \oplus 2^{d-2}7L, \quad (17)$$

where

$$S' = E_k(M[d-1] \oplus 2^{d-2}3L) \oplus \cdots \oplus E_k(M[2] \oplus 6L) \oplus E_k(M[1] \oplus 3L) \oplus L \oplus V. \quad (18)$$

We can see that this case is similar to Case 2 in Sect. 3.2. The function can be defined as below:

$$f : \{0,1\}^n \rightarrow \{0,1\}^n$$
$$x \rightarrow Tag\_AES - COPA\_v.1(x||x \oplus \sigma) =$$
$$XLS_d(E_k(E_k(\sigma \oplus 10L) \oplus E_k(x \oplus \sigma \oplus 6L) \oplus E_k(x \oplus 3L) \oplus L) \oplus 14L)$$
$$(19)$$

where $M[1] \oplus M[2] = \sigma$. Finally, we can apply Simon algorithm on this function f to get period $s = \sigma \oplus 5L$ (as shown in Fig. 9). The tag of $M[1]||M[2]||\cdots||M[d]$ is valid for $M[2] \oplus 5L||M[1] \oplus 5L||\cdots||M[d]$.

In addition to the above two cases, there is a special case, i.e., $d = 1$ and $|M[1]| \% n \neq 0$. Due to space limitations, we do not introduce the encryption process of AES-COPA v.1 in this case (the entire process can be referred to Ref. [5]). And the positions of bits in tag need to be moved in the encryption process, which is not suitable for using the Simon algorithm to implement quantum forgery attacks. Therefore, we do not discuss this case.

4.2 Quantum Forgery Attacks on AES-COPA v.2

From the entire encryption process of AES-COPA v.2, we can find that the encryption of AES-COPA v.2 for fractional messages is a little different from COPA and AES-COPA v.1. To implement forgery attack, we consider three cases: 1)$d = 1$, $|M[1]| \% n \neq 0$; 2)$d = 2$, $|M[2]| \% n \neq 0$; 3) $d > 2$, $|M[d]| \% n \neq 0$. Assume that the associated data and Nonce are fixed constants.

Case 1: When $d = 1$, $|M[1]| \% n \neq 0$, then,

$$T = E_k(E_k(M[1]||10^* \oplus 3^2 7L) \oplus E_k(M[1]||10^* \oplus 3L) \oplus V \oplus L) \oplus 14L \quad (20)$$

The function can be defined:

$$\begin{aligned} f : \{0,1\}^n &\to \{0,1\}^n \\ x &\to Tag\_AES - COPA\_v.2(x) = E_k(E_k(x \oplus 27L) \oplus E_k(x \oplus 3L) \\ &\oplus L \oplus V) \oplus 14L, \end{aligned}$$

$$(21)$$

where $27 = 3^2 7 = (x+1)^2(x^2+x+1) mod(x^{128}+x^7+x^2+x+1) = x^4+x^3+x+1$.

The circuit of our attack in this case is similar to Fig .4. Only the cryptographic oracle is the Eq. 21 of AES-COPA v.2. The result period is $s = 24L$. So, the tag of $M[1]$ is as same as $\lfloor M[1]||10^* \oplus 24L \rfloor_{10^*}$, where $\lfloor A \rfloor_{10^*}$ indicates that the bit string "10^*" at the end of bit string A should be removed. For example, if the last bit in bit string A is "1", then the last bit of A should be removed; if the last two bits in bit string A are "10", then the last two bits of A should be removed; and so on.

Case 2: When $d = 2$, $|M[2]| \% n \neq 0$, then,

$$\begin{aligned} T =& E_k(E_k(M[1] \oplus M[2]||10^* \oplus 54L) \oplus E_k(M[1] \\ & \oplus 3L) \oplus E_k(M[2]||10^* \oplus 18L) \oplus V \oplus L) \oplus 28L \end{aligned} \quad (22)$$

where $54 = 2 \cdot 3^2 \cdot 7$ and $18 = 2 \cdot 3 \cdot 7$.

The function f is defined as below:

$$\begin{aligned} f : \{0,1\}^n &\to \{0,1\}^n \\ x &\to Tag\_AES - COPA\_v.2(x||x \oplus \sigma) = E_k(E_k(\sigma \oplus 54L) \quad (23) \\ &\oplus E_k(x \oplus 3L) \oplus E_k(x \oplus \sigma \oplus 18L) \oplus V \oplus L) \oplus 28L, \end{aligned}$$

where $\sigma = M[1] \oplus M[2]||10^*$. Then, we apply Simon's algorithm on this function f, whose circuit is similar to Fig. 5, and get the period $s = \sigma \oplus 17L$. So, the tag of $M[1]||M[2]$ is equal to $\lfloor M[2]||10^* \oplus 17L||M[1] \oplus 17L \rfloor_{10^*}$.

Case 3: When $d > 2$, $|M[d]| \% n \neq 0$. Since our attack only requires the first two message blocks $M[1]||M[2]$, whether the last message block is full does not affect implementation of forgery attacks. The process of our quantum forgery attack is as same as Case 2 in Sect. 4.2.

Similar to the quantum forgery attacks on COPA, under the Q2 model, the number of quantum superposition queries is the number of times that Simon's algorithm is executed, and the success rate of our attack is also the success rate of the hidden period calculated.
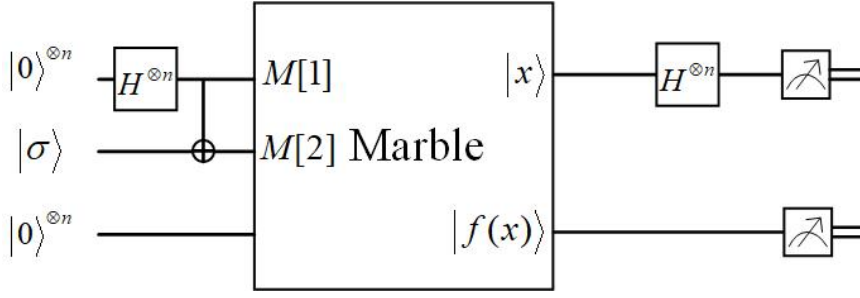
Fig. 10: The circuit of quantum forgery attack on Marble.

5 Quantum Forgery Attacks on Marble by Simon's Algorithm

By observing the entire overview of Marble v1.2, we find that the attack strategy in Sect. 3 can not apply on Marble directly. But inspired by Lu's attack strategy [12], we can use Simon algorithm to recover the value of L firstly, and then calculate the tag of new message by L to achieve the purpose of forgery attack.

To recover the value of L, we can apply Simon algorithm on the process of calculating S_1 (S_1 in Marble v1.2):

$$
\begin{aligned}
S_1 =& Const_1 \oplus E_1(Const_0) \oplus E_1(A[1] \oplus 5L) \oplus E_1(A[2] \oplus 10L) \oplus \cdots \\
& \oplus E_1(A[a] \oplus 2^{a-1}3^3 L) \oplus E_1(M[1] \oplus 2L) \oplus E_1(M[2] \oplus 4L) \oplus \cdots \quad (24) \\
& \oplus E_1(M[d] \oplus 2^d L) \oplus E_1(\Sigma \oplus \tau \oplus 2^d \cdot 7L),
\end{aligned}
$$

For easy computing, we do not consider the associated data and choose an arbitrary constant σ, and then define the following function:

$$
\begin{aligned}
f : \{0,1\}^n &\to \{0,1\}^n \\
x &\to S_1\_Marble(A=0, M=x||x \oplus \sigma) = \\
& Const_1 \oplus E_1(Const_0) \oplus E_1(x \oplus 2L) \oplus E_1(x \oplus \sigma \oplus 4L) \oplus E_1(\sigma \oplus 14L)
\end{aligned}
$$

$$(25)$$

where $Const_0$ and $Const_1$ are constants. Finally, we can use Simon algorithm to recover the function f's period $s = \sigma \oplus 6L$ (as shown in Fig. 10). The value of L can be obtained by s.

When we get the value of L, we can query the Marble v1.2 encryption oracle with $d+1$ block message $\tilde{M} = M[1]||M[2]||\cdots||M[d]|| \oplus_{i=1}^d M_i \oplus 2^{d+1}L \oplus 2^d \cdot 7L$, and obtain its ciphertext $\tilde{C} = (C[1], C[2], \cdots, C[d], \tilde{C}[d+1])$. Then, the ciphertext for $M = M[1]||M[2]||\cdots||M[d]$ is $C = (C[1], C[2], \cdots, C[d])$, and the tag for M is $\tilde{C}[d+1] \oplus 2^d \cdot 3 \cdot L \oplus 2^{d-1} \cdot 3 \cdot 7 \cdot L$. Since the associated data is not considered, the ciphertext and tag are also applicable to the Marble v1.1. The specific process of generating forged tag can refer to Ref. [12].

On the other hand, if the associated data is considered, the forgery attack can refer to Fuhr *et al.*'s attack strategy [14]. For any associated data A and

message M, the adversary computes the masked value B of a chunk of 8 identical blocks of associated data after A and queries the encryption oracle on $(A||B, M)$. The answer (C, T) to that query is also valid ciphertext and tag for (A, M). The attack for Marble v1.0 is also applicable to other versions of Marble.

Different from the quantum forgery attacks on COPA and AES-COPA, the quantum forgery attacks here are mainly to obtain the secret parameter L through quantum superposition query and Simon's algorithm. But the attack process is similar. So the number of queries and the success rate are the same.

6 Efficiency Comparison

In this section, we will compare our quantum forgery attack with other forgery attacks (Ref. [11, 12, 14]) in terms of attack efficiency, which mainly consists of two aspects: the number of queries and the success probability of forgery attacks. The comparison result is shown in Table 3. In Ref. [12, 14], they use birthday attack to find the collisions of the tag, which can recover the secret parameter L. Then, they can compute the forgery tag by querying the oracle once for specified message. We can see that the number of queries mainly concentrate in recovering the secret parameter L. So, the number of queries is close to the birthday-bound $2^{n/2}$. And Ref. [11] uses pigeonhole principle on the case of processing fractional messages to implement forgery attack, which only needs $2^{n/3}$ queries. Their success probability is a trade-off with the number of queries.

In our quantum forgery attack, we rely on the Simon's algorithm to query collisions. Therefore, the number of queries in our attack is roughly equivalent to the number of repeating Simon's algorithm. And the success probability of our attack is equal to the success probability of Simon's algorithm. Due to the characteristics of the queries in superposition, the number of queries can be reduced from exponential time to polynomial time. Our attack only needs cn queries, where c is a constant. Besides, Shi *et al.* [28] proved that Simon's algorithm returns s with cn queries, with probability at least $1 - 2^n \times (0.6454)^{cn}$. If we choose $c = 4$ and $n = 128$, the success probability is very close to 1, which is much greater than these classic forgery attack.

7 Discussion and Conclusion

In this paper, we have presented quantum forgery attacks on COPA, AES-COPA and Marble authenticated encryption algorithms. Due to quantum superposition query and Simon's algorithm, our quantum forgery attack on COPA, AES-COPA and Marble are more efficient than classic forgery attacks, i.e., it only needs cn queries and its success probability is very close to 1.

However, the premise for our quantum forgery attack to be effective is in the quantum setting. If the attacker only queries classically, we may not reduce the

Table 3: Comparison between classic forgery attack and quantum forgery attack

		Ref. [12]	Ref. [11]	Ref. [14]	Our attack
COPA	Queries	$2^\sigma + 2^\varphi \left(1 \leqslant \sigma, \varphi < \frac{n}{2}\right)$	$2^{n/3}$	#	cn
	Success Prob.	$1 - e^{-2^{\sigma+\varphi-n}}$	25%	#	$1 - 2^n \times (0.6454)^{cn}$
AES-COPA (v.1/2)	Queries	2^{63}	#	#	cn
	Success Prob.	6%	#	#	$1 - 2^n \times (0.6454)^{cn}$
Marble (v1.0/1/2)	Queries	2^{65}	#	2^{65}	cn
	Success Prob.	32%	#	32%	$1 - 2^n \times (0.6454)^{cn}$

queries to only $O(n)$ times. But we can use Grover algorithm offline to find collisions, which can reduce the queries from $O(2^{n/2})$ to $O(2^{n/3})$. So, using quantum algorithms offline to improve the efficiency of breaking symmetric cipher will be one of our future research directions.

References

1. Lu, J.: On the Security of the LAC Authenticated Encryption Algorithm. in Proc. of Australasian Conference on Information Security and Privacy, ACISP 2016. 395-408 (2016)
2. CAESAR-Competition for Authenticated Encryption: Security, Applicability, and Robustness. http://competitions.cr.yp.to/caesar.html
3. Boer, G.J., McFarlane, N.A.: The AES atmospheric general circulation model. GARP Publ. Ser. **22**, 409-460 (1979)
4. Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Tischhauser, E., Yasuda, K.: Parallelizable and Authenticated Online Ciphers. in Proc. of Advances in Cryptology-ASIACRYPT 2013. 424-443 (2013)
5. Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Tischhauser, E., Yasuda, K.: AES-COPA v1. Submission to the CAESAR competition, March 2014. http://competitions.cr.yp.to/round1/aescopav1.pdf
6. Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Tischhauser, E., Yasuda, K.: AES-COPA v2. Submission to the CAESAR competition. 2015. http://competitions.cr.yp.to/round1/aescopav2.pdf
7. National Institute of Standards and Technology (NIST). Advanced Encryption Standard (AES), FIPS-197, 2001.
8. Guo, J.: Marble Specification Version 1.0. Submission to the CAESAR competition, 15 March 2014. http://competitions.cr.yp.to/round1/marblev10.pdf
9. Guo, J.: Marble Specification Version 1.1. Submission to the CAESAR competition, 26 March 2014. http://competitions.cr.yp.to/round1/marblev11.pdf

10. Guo, J.: Marble Specification Version 1.2. Submission to the CAESAR competition, 16 January 2015. https://groups.google.com/forum/#!topic/crypto-competitions/FoJITsVbBdM
11. Nandi, M.: Revisiting Security Claims of XLS and COPA. IACR Cryptology ePrint Archive. 444 (2015)
12. Lu, J.: Almost Universal Forgery Attacks on the COPA and Marble Authenticated Encryption Algorithms. in Proc. of the 2017 ACM on Asia Conference on Computer and Communications Security. 789-799 (2017)
13. Dunkelman, O., Keller, N., Shamir, A.: Almost universal forgery attacks on AES-based MAC's. Designs, Codes and Cryptography. 76(3), 431-449 (2015)
14. Fuhr, T., Leurent, G., Suder, V.: Collision Attacks Against CAESAR Candidates. in Proc. of Advances in Cryptology-ASIACRYPT 2015. 510-532 (2015)
15. Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. in Proc. of 35th Annual Symposium on Foundations of Computer Science. 124-134 (1997)
16. Liu, W-J., Gao, P-P., Yu, W-B., Qu, Z-G., Yang, C-N.: Quantum Relief algorithm. Quantum Information Processing, 17(10), 280 (2018)
17. Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., Lloyd, S.: Quantum machine learning. Nature. 549, 195-202 (2017)
18. Liu, W., Chen, J., Wang, Y., Gao, P., Lei, Z.: Quantum-Based Feature Selection for Multiclassification Problem in Complex Systems with Edge Computing. Complexity. 2020, 8216874 (2020)
19. Gao, Y-L., Chen, X-B., Xu, G., Yuan K-G., Liu W., Yang Y-X.: A novel quantum blockchain scheme base on quantum entanglement and DPoS. Quantum Information Processing. 19, 420 (2020)
20. Banerjee, S., Mukherjee, A., Panigrahi, P. K.: Quantum blockchain using weighted hypergraph states. Physical Review Research. 2(1), 013322 (2020)
21. Simon, D.R.: On the Power of Quantum Computation. SIAM Journal on Computing. 26(5), 1474-1483 (1997)
22. Grover, L. K.: Quantum Computers Can Search Arbitrarily Large Databases by a Single Query. Physical Review Letters. 79(23), 4709-4712 (1997)
23. Bernstein, E., Vazirani, U.: Quantum Complexity Theory. SIAM Journal on Computing. 26(5), 1411-1473 (1997)
24. Kuwakado, H., Morii, M.: Quantum distinguisher between the 3-round Feistel cipher and the random permutation. in Proc.of the 2010 IEEE International Symposium on Information Theory. 13-18 (2010)
25. Kuwakado, H., Morii, M.: Security on the quantum-type Even-Mansour cipher. in Proc.of the 2012 International Symposium on Information Theory and its Applications. 28-31 (2012)
26. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Breaking Symmetric Cryptosystems Using Quantum Period Finding. in Proc.of the Advances in Cryptology-CRYPTO 2016. 207-237(2016)
27. Shi, T., Jin, C., Guan, J.: Collision attacks against AEZ-PRF for authenticated encryption AEZ. China Communications. 15(2), 46-53 (2018)
28. Shi, T.R., Jin, C.H., Hu, B., Guan, J., Cui, J.Y., Wang, S.P.: Complete analysis of Simon's quantum algorithm with additional collisions. Quantum Information Processing. 18(11), 334 (2019)
29. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Quantum Differential and Linear Cryptanalysis. arXiv:1510.05836 (2015)
30. Chailloux, A., Naya-Plasencia, M., Schrottenloher, A.: An Efficient Quantum Collision Search Algorithm and Implications on Symmetric Cryptography. in Proc. of Advances in Cryptology-ASIACRYPT 2017. 211-240 (2017)
31. Leander, G. May, A.: Grover Meets Simon - Quantumly Attacking the FX-construction. in Proc.of Advances in Cryptology-ASIACRYPT 2017. 161-178 (2017)
32. Xie, H., Yang, L.: Using Bernstein-Vazirani algorithm to attack block ciphers. Designs, Codes and Cryptography. 87(5), 1161-1182 (2019)
33. Hosoyamada, A., Sasaki, Y., Xagawa, K.: Quantum Multicollision-Finding Algorithm. in Proc. of Advances in Cryptology-ASIACRYPT 2017. 179-210 (2017)
34. Hosoyamada, A., Sasaki, Y., Tani, S., Xagawa, K.: Improved Quantum Multicollision-Finding Algorithm. in Proc. of Post-Quantum Cryptography 2019. 350-367 (2019)

35. Bonnetain, X., Naya-Plasencia, M., Schrottenloher, A.: On Quantum Slide Attacks. in
    Proc. of Selected Areas in Cryptography-SAC 2019. 492-519 (2019)
36. Hosoyamada, A., Sasaki, Y.: Quantum Demiric-Selcuk Meet-in-the-Middle Attacks: Ap-
    plications to 6-Round Generic Feistel Constructions. in Proc. of Security and Cryptogra-
    phy for Networks 2018. 386-403 (2018)