TSINGHUA SCIENCE AND TECHNOLOGY
ISSNll1007-0214 07/10 pp585–595
DOI: 10.26599/TST.2018.9010130
Volume 24, Number 5, October 2019

Key-Recovery Attacks on LED-Like Block Ciphers

Linhong Xu, Jiansheng Guo*, Jingyi Cui, and Mingming Li

Abstract: Asymmetric cryptographic schemes, represented by RSA, have been shown to be insecure under quantum computing conditions. Correspondingly, there is a need to study whether the symmetric cryptosystem can still guarantee high security with the advent of quantum computers. In this paper, based on the basic principles of classical slide attacks and Simon's algorithm, we take LED-like lightweight block ciphers as research objects to present a security analysis under both classical and quantum attacks, fully considering the influence on the security of the ciphers of adding the round constants. By analyzing the information leakage of round constants, we can introduce the differential of the round constants to propose a classical slide attack on full-round LED-64 with a probability of 1. The analysis result shows that LED-64 is unable to resist this kind of classical slide attack, but that attack method is not applicable to LED-128. As for quantum attacks, by improving on existing quantum attack methods we demonstrate a quantum single-key slide attack on LED-64 and a quantum related-key attack on LED-128, and indicators of the two attack algorithms are analyzed in detail. The attack results show that adding round constants does not completely improve the security of the ciphers, and quantum attacks can provide an exponential speed-up over the same attacks in the classical model. It further illustrates that the block cipher that is proved to be safe under classical settings is not necessarily secure under quantum conditions.

Key words: key-recovery attack; cryptanalysis; post-quantum cryptography; lightweight block cipher; LED

1 Introduction

With the continuous development of quantum computing, its application in the field of cryptography has gradually become a research hotspot in academia and industry. Cryptography has also entered the era of post-quantum cryptography, one feature of which is that the influence of quantum computers on the security of existing cryptographic algorithms is now of great concern.

Currently, much research is focusing on asymmetric cryptographic schemes. The most famous discovery is that RSA[1] can be broken in polynomial time under quantum computing conditions by way of Shor's algorithm[2]. For symmetric cryptographic schemes, it is worth investigating whether security under quantum computing conditions is consistent with that under classical settings. The same algorithms that can achieve exponential increase in speed in quantum computers can also be applied to symmetric cryptography. For example, Grover's pioneering result[3] can reduce the time complexity for exhaustive key attack on an n-bit key block cipher from $O(2^n)$ to $O(2^{\frac{n}{2}})$. Simon[4] demonstrated an algorithm to calculate the period of a given function in polynomial time.

Based on existing quantum algorithms, a series of quantum cryptanalysis methods have been proposed. In 2016, Kaplan et al.[5] gave a quantum slide attack method to the iterative Even-Mansour (E-M) ciphers[6] using the same round keys as Simon's algorithm. Leurent et al.[7] improved Grover's algorithm to provide general methods for quantum differential

• Linhong Xu, Jiansheng Guo, Jingyi Cui, and Mingming Li are with the Information Science and Technology Institute, Zhengzhou 450001, China. E-mail: xlh_right@126.com; tsg_31@126.com; xd_cjy@126.com; 18203622214@163.com.
∗ To whom correspondence should be addressed.

Manuscript received: 2018-10-16; accepted: 2018-11-10

and linear analysis. The attacks in Refs. [5, 7] are based on the promise that an adversary can use the quantum superposition state to query the encryption oracle and perform quantum computation operations, this is the Q_2 model assumption defined in Ref. [7]. Correspondingly, if the adversary performs only classical operations during the data collection phase and performs quantum operations in the key recovery phase, this is denoted as the Q_1 model.

Similar quantum cryptanalysis methods based on the Q_2 model also appear in subsequent work. Kuwakado and Morii[8, 9] proved that the 3-round Feistel and E-M structure are insecure with superposition queries. In 2015, Roetteler and Steinwandt[10] presented a quantum related-key attack based on Simon's algorithm. In 2017, Hosoyamada and Aoki[11] built on the work in Ref. [5], proposed an improved polynomial-time quantum related-key attack. In Ref. [11], the authors targeted iterative E-M structural ciphers using different round keys and gave a specific example of a key-recovery attack on the 2-round E-M structural cipher. Leurent and May[12] combined Grover's algorithm with Simon's algorithm to demonstrate a quantum attack on the block ciphers constructed by the FX structure.

The classical slide attack[13], such as the side-channel attack for Ref. [14], is a very effective method of cryptanalysis. This can be seen as a variant of the related-key attack and the method is applicable in both single-key and related-key models. In general, this attack requires a block cipher and has the following features:

(1) The same round function, or several rounds of the round function form a period.

(2) A simple key schedule, such as using the same master key for round keys.

In summary, the cipher has self-similarity. In order to resist classical slide attack, cryptologist destroy this self-similarity by adding round constants to the ciphers. However, different ways of adding these round constants also influence the ability to resist such an attack.

Our contributions. The main objective of this paper is to show the key-recovery attack on LED-like[15] lightweight block ciphers under classical and quantum computing settings. Section 2.2 gives the specific properties of LED-like block ciphers.

(1) Under the classical setting, we improve the original slide attack. By analyzing the information leakage of round constants, we can introduce the differential of the round constants to propose a classical slide attack on full-round LED-64 with a probability of 1. But this kind of classical slide attack is not applicable to LED-128.

(2) Under the quantum setting, by improving the existing attack methods, we show a quantum single-key slide attack on LED-64 and a quantum related-key attack on LED-128. The given quantum attacks are based on the Q_2 model. We then analyze the success probability and complexity indicators of the attacks in detail.

The attack results show that, for LED-like block ciphers, an irrational way of adding round constants does not necessarily improve the security of the ciphers and the ciphers that are proven to be safe under classical settings are not necessarily secure under quantum attack conditions.

Organization. The paper is organized as follows. First, Section 1 mainly introduces the research background and significance of this article. Section 2 provides background knowledge for the research, including the description of the quantum gate circuit, the introduction of the LED-like block ciphers, and the classic slide attack method. Section 3 describes the basic principles of Simon's algorithm and the two quantum attack methods. Sections 4 and 5 then take LED-64 and LED-128 as target ciphers and give corresponding attack algorithms and analyze of various indexes of these attack algorithms. Finally, Section 6 concludes the paper and points out some possible new research directions.

2 Preliminaries

2.1 Symbol description

E_K: A full-round block cipher.

$K = (k_1, k_2, k_3, \ldots, k_r)$: Round keys of an r-round block cipher.

P_i: The i-th round function.

rc: Initial round constant of LED.

$|\psi\rangle$: Quantum state.

m: Plaintext.

c: Ciphertext.

2.2 Basic quantum gates and circuits

The quantum gates that will be used later are briefly introduced in this section.

For $b \in \{0,1\}$, $x, y \in \{0,1\}^n$, Fig. 1 shows gate $H^{\otimes n}$, gate *CNOT*, and gate *CCNOT*. They are

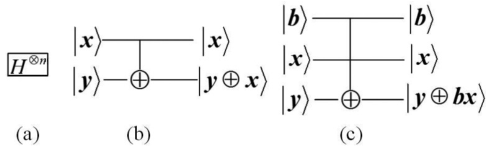


Fig. 1 (a) gate $H^{\otimes n}$, (b) gate *CNOT*, and (c) gate *CCNOT*.

$$H^{\otimes n}|x\rangle \mapsto \frac{1}{\sqrt{2^n}}\sum(-1)^{x\cdot y}|y\rangle,$$
$$CNOT:|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus x\rangle,$$
$$CCNOT:|b\rangle\,|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus bx\rangle.$$

For the public random permutation P and the function f, we call the quantum gates of P and quantum oracle f, P: $|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus P(x)\rangle$ and f: $|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus f(x)\rangle$ (see Fig. 2). Figure 3 shows a concrete representation of the quantum gate of controlled P, CP:$|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus bP(x)\rangle$, and the quantum circuit of controlled function f, Cf:$|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus bf(x)\rangle$.

2.3 LED-like block ciphers

In 1997, Even and Mansour[6] proposed a simple structure for constructing a cryptographic algorithm using pseudo-random permutation. It was defined as an E-M structure, and the corresponding security proof was given. Specifically, for a pseudo-random permutation P with n-bit size, and the keys k_1 and k_2, we can construct a cryptographic algorithm $E_{k_1,k_2}(x) = P(x \oplus k_1) \oplus k_2$. This algorithm is considered to be secure under attacks with time complexity lower than exhaustive search. Many existing block ciphers are constructed based on this structure, such as traditional block cipher-AES[16], lightweight block ciphers PRINCE[17], and LED.

LED is a 64-bit lightweight block cipher proposed by Guo et al.[15] in CHES-2011. The two main variants of the cipher are LED-64 and LED-128, which support the key size 64 and 128, respectively. The corresponding numbers of rounds of cipher are 32 and 48. In the round function part, two key-size LED algorithms use the same round operation. Each round consists of four transformations in the sequence of AddConstants, SubCells, ShiftRows, and MixColumns. The construction of LED-64 is a generalized E-M structure with one key k_1 and 8 steps. Each step includes four rounds. Slightly different from LED-64, LED-128 includes 12 steps and alternate uses the master keys k_1 and k_2 as the round key. For more details of LED, see Ref. [15].

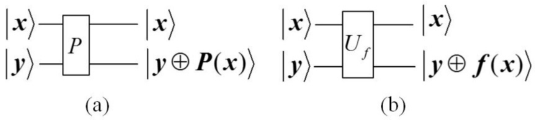


Fig. 2 (a) Quantum gate *P* and (b) quantum oracle of *f*.

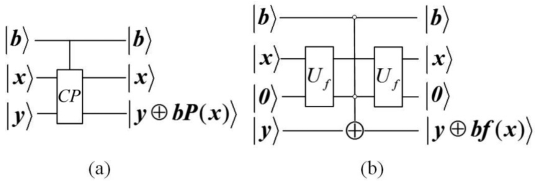


Fig. 3 (a) *CP* and (b) *Cf*.

This paper mainly studies the LED-like lightweight block ciphers with iterative E-M structure. Some properties of this type block ciphers are as follows.

(1) The design of the ciphers can be regarded as the iterative transformation based on the basic E-M structure.

(2) The ciphers have a simple key schedule. For example, LED uses the master key directly as the round key.

(3) The round function uses different round constants in a cipher, and each round constant is affected by the previous round. If the previous round constants are changed, the round constants in the subsequent round will also change.

(4) The ciphers can use a single round operation as a round function or, similar to LED, can use multiple round operations as a round function. In accordance with the aboved property (3), since the round constants are inserted in the round functions of the ciphers, it is obvious that each round function is different.

2.4 Classical slide attack

The classic slide attack is not limited by the number of rounds of the ciphers, and it can perform security analysis on all-round cryptographic algorithms.

E is an n-bit block cipher with r rounds, $E = P_r \circ P_{r-1} \circ \cdots \circ P_1$. Each P_i is the same round function, and round keys are generated by the key schedule. The idea of the original slide attack is mainly focused on the slide element, which means that one encryption process slides over another to ensure that the two encryption processes are identical except for the difference in encryption order. At this point, the adversary needs to find two sets of plaintext-ciphertext pairs (m, c) and (m_1, c_1) satisfying the relationship of $m_1 = P_1(m \oplus k)$ and $c_1 = P_{r+1}(c) \oplus k$. Based on this, the correct key can be recovered, and the plaintext-ciphertext pair which satisfies the corresponding relationship is called a slide pair. According to the principle of birthday attack, $2^{\frac{n}{2}}$ plaintext-ciphertext pairs are needed to find the slide pair in general, then the correct key can be recovered. Figure 4 shows the original slide attack.

In this paper, we give an improved classical slide attack method by introducing the differential of the round constants, and then applying it to perform a key-recovery attack on LED-like ciphers. For specific examples, see Sections 4.1 and 5.1.

Here, we compare the ability of block ciphers to resist classical slide attacks using different methods of adding round constants and different key schedules.

(1) LED-64-like ciphers can resist the original slide attack, but they are unable to resist the improved slide attack presented below in Section 4.1.

(2) LED-128-like ciphers have the same method of adding round constants as LED-64-like ciphers. They also use the master keys directly as the round keys, but the round keys form a loop every few rounds. This kind of cipher can resist the original slide attack, and Section 5.1 below proves that it can also resist the improved classical slide attack under the related-key conditions.

(3) For ciphers in which round constants are added in the same way as LED-like ciphers (i.e., the round keys used are derived from the master keys through key schedule), each round key is not the same but there is a certain link between them. Due to the correlation between the round keys, an adversary can use the improved slide attack based on the related-key model to find the slide pair and filter out the correct key. That is, such ciphers are generally unable to resist the improved classical related-key slide attack.

(4) Ciphers using a fixed random number as the round constants with no link between each round constant, can resist the original slide attack and the improved slide attack presented in this paper, regardless of the key schedule of the ciphers.

3 Basic Principle of Quantum Attack

Grassl et al.[18] gave the exact number of qubits and basic logic gates needed to attack AES by Grover's algorithm[3], and provided a basic method for constructing a quantum circuit of a cryptographic algorithm. In the present paper, the assumption behind the quantum attacks is that the adversary can perform a quantum query to the encryption circuits and perform quantum computation (Q_2 model). Under the method provided in Ref. [18], the encryption circuits can be constructed, so that the attack assumption can be implemented under the conditions of quantum computing. In addition, when analyzing the complexity of the attack algorithm under the Q_2 model, it is reasonable to only consider the complexity required for the quantum query and classical computations, without considering the computational complexity required to construct a quantum circuit. The basic principles of the Simon's algorithm[4], the quantum slide attack, and quantum related-key attack are introduced below.

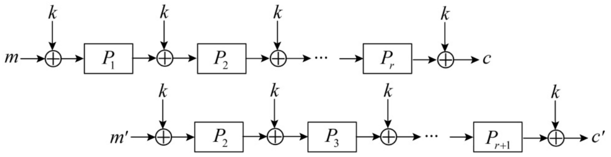


Fig. 4 Original slide attack.

3.1 Simon's algorithm

Problem 1[4] Assume that f is a function, f: $\{0,1\}^n \rightarrow \{0,1\}^n$. For $\forall x \in \{0,1\}^n$ and some $\boldsymbol{s} \in \{0,1\}^n$, that satisfy $f\ (\ x \oplus \boldsymbol{s}) = f(x)$, how to find $\boldsymbol{s}$?

The computational complexity required for the optimal algorithm to solve the above problem under classical settings is $O(2^{\frac{n}{2}})$. Simon[4] proposed an exponential speed-up quantum algorithm that requires only $O(n)$ quantum circuit queries to find the period $\boldsymbol{s}$.

The quantum part of Simon's algorithm mainly serves to execute the following subroutine, where the quantum query to the classical function f is formalized in the standard way by a unitary transform $U_f|\boldsymbol{x}\rangle\,|\boldsymbol{y}\rangle = |\boldsymbol{x}\rangle\,|\boldsymbol{y} \oplus f(\boldsymbol{x})\rangle$. The main steps are as follows.

First construct the quantum circuit Q (see Fig. 5), which contains the quantum oracle of the function f, U_f. Select the first register as the data register A, the second register as the target register B. Then, measuring the quantum state of register B in $|\boldsymbol{\psi}_1\rangle$, and the quantum state of register A collapse to $|\boldsymbol{\psi}_2\rangle$. Applying another Hadamard transform leads $|\boldsymbol{\psi}_2\rangle$ to the state

$$|\boldsymbol{\psi}_3\rangle = \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2^n}}\sum(-1)^{y\cdot z}(1+(-1)^{y\cdot s})|\boldsymbol{y}\rangle.$$

Measuring $|\boldsymbol{\psi}_3\rangle$ will result in vectors $\boldsymbol{y} \in \{0,1\}^n$. Note that for $\boldsymbol{y} \in \{0,1\}^n$ with $\boldsymbol{y} \cdot \boldsymbol{s} = y_1 \cdot s_1 \oplus y_2 \cdot s_2 \oplus$

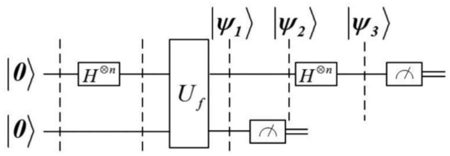


Fig. 5 Quantum circuit of Simon's algorithm.

$\cdots \oplus y_n \cdot s_n = 1$, there is destructive interference and the amplitudes of those strings vanish. Therefore, the distribution of output vectors $\boldsymbol{y}$ is consistent with the uniform distribution on set $\{\, \boldsymbol{y} \in \{0,1\}^n | \boldsymbol{y} \cdot \boldsymbol{s} = 0\}$. Repeating this quantum procedure $O(n)$ times can obtain the orthogonal space of $\boldsymbol{s}$ with high probability (Lemma 1), which can be efficiently solved classically to obtain the string $\boldsymbol{s}$. More details are described in Ref. [4].

Lemma 1[4] Assume there is a periodic function f with period $\boldsymbol{s}$, $\exists p_0$, $0 < p_0 < 1$ that satisfies

$$\varepsilon(f;\boldsymbol{s}) = \max_{t \notin \{0,s\}} \Pr_x[f(x) = f(x \oplus \boldsymbol{t})] \leqslant p_0.$$

By repeating this quantum procedure cn times, $\boldsymbol{s}$ can be obtained with a probability at least $1 - (2(\frac{1+p_0}{2})^c)^n$.

3.2 Quantum slide attack

In 2016, Kaplan et al.[5] proposed a quantum slide attack algorithm for recovering the keys of block ciphers in polynomial time by Simon's algorithm. They then applied the quantum slide attack to iterative E-M structural ciphers using the same round keys and round functions. In this paper, we improve the attack method and give a quantum slide attack on LED-64-like ciphers using same round keys and different round functions. Note that, the different round functions in this paper are specific to their use in the round function, with the rest of the operations remaining the same.

Assume that E_1 is an r-round block cipher. Its block size and key size are both n-bit. Every round uses the same round key k, $k \in \{0,1\}^n$. The i-th round function is defined as P_i, $i \in \{1, 2, \ldots, r\}$. For each P_i, except for the values of the round constants used in AddConstants, the rest of the operations are all the same. It is clear that each P_i can be seen as an n-bit random permutation. The block cipher can be expressed as

$$C = E_1(X) = (P_{k_r} \circ P_{k_{r-1}} \circ \cdots \circ P_{k_2} \circ P_{k_1})(X) \oplus k,$$

among it, $P_{k_r} = P_r(x \oplus k)$. Choosing such two block ciphers E_1 and E_2. $C = E_1(X)$ and $C' = E_2(X) = (P_{k_r'} \circ P_{k_{r-1}'} \circ \cdots \circ P_{k_2'} \circ P_{k_1'})(X) \oplus k$, $P_{k_r'} = P_r'(x \oplus k)$. Among these, in order to satisfy the conditions of slide attack, we introduce a differential in the initial round constant, leading to $P_i' = P_{i+1}$, $(1 \leqslant i \leqslant r)$, for the round function P_i' in E_2 and the round function P_i in E_1. Lemma 2 introduces a class of periodic functions.

Lemma 2[5] Assume that there are two block ciphers E_1 and E_2 as described above, P_1 and $P_r + 1$ are the first and the $(r+1)$-th round function of the cipher E_1, respectively. We define the following function g,

$$g : \{0,1\}^{n+1} \to \{0,1\}^n,$$

$$g(b||x) = \begin{cases} P_{r+1}(E_1(x)) \oplus x, b = 0, \\ E_2(P_1(x)) \oplus x, b = 1. \end{cases}$$

For all $x \in \{0,1\}^n$ and $b \in \{0,1\}$, g is a periodic function with $\boldsymbol{s} = 1||k$.

In order to apply Lemma 1 to obtain $\boldsymbol{s}$, we bound $\varepsilon(g, 1||k)$,

$$\varepsilon(g, 1||k) = \max_{(\boldsymbol{\tau}||\boldsymbol{t}) \notin \{(0||0),(1||k)\}} \Pr_x[g(b||x) = g\,(b \oplus \boldsymbol{\tau}||x \oplus \boldsymbol{t})],$$

assuming that both $P_{r+1} \circ E_1$ and $E_2 \circ P_1$ are indistinguishable from random permutations.

Lemma 3[5] For the function defined in Lemma 2, it satisfies

$$\varepsilon(g, 1||k) = \max_{(\boldsymbol{\tau}||\boldsymbol{t}) \notin \{(0||0),(1||k)\}} \Pr_x[g(b||x) = g\,(b \oplus \boldsymbol{\tau}||x \oplus \boldsymbol{t})] \leqslant \frac{1}{2}.$$

According to Lemmas 2 and 3, the function satisfies the promises of Simon's problem with $\boldsymbol{s} = 1||k$, so that the key k of E_1 can be recovered with $O(cn)$ complexity. The quantum circuit U_g is shown in Fig. 6, which modifies Fig. 7 in Ref. [11] to make the representation more accurate. The attack application of LED-64 is given in Section 4.2.

3.3 Quantum related-key attack

In 2017, based on the work of Kaplan et al.[5], Hosoyamada and Aoki[11] presented a quantum related-key cryptanalysis technique for a class of ciphers constructed by iterative E-M structure using different

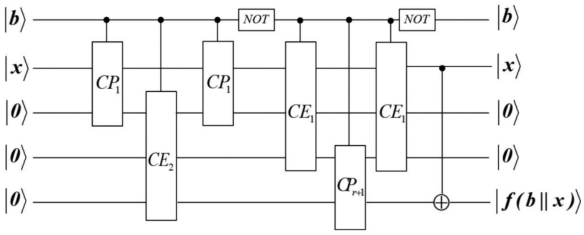


Fig. 6 Quantum circuit U_g.

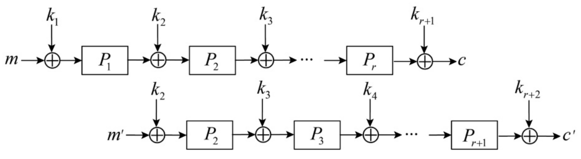


Fig. 7 Related-key slide attack.

round keys and the same round function. Inspired by Ref. [11], this paper proposes a quantum related-key attack algorithm to LED-like ciphers using different round keys and different round functions. In Section 5, the attack application of LED-128 is given.

Defining an r-round block cipher E_K, with block size n-bit, key size $2n$-bit, and the round function P_i, $i \in \{1, 2, \ldots, r\}$. $K = (k_1, k_2, k_3, \ldots, k_r, k_{r+1})$ represents the round keys generated by the key-schedule and k_{r+1} denotes the whitening key. Assuming an adversary can query such two quantum oracles E_K and $E'_{K'}$,

$$C = E_K(X) = (P_{k_r} \circ P_{k_{r-1}} \circ \cdots \circ P_{k_2} \circ P_{k_1})(X) \oplus k_{r+1},$$

$$C' = E'_{K'}(X) = (P_{k'_r} \circ P_{k'_{r-1}} \circ \cdots \circ P_{k'_1})(X') \oplus k'_{r+1}.$$

Among these, $P_{k_r} = P_r(x \oplus k_r)$, $P'_{k_r} = P'_r(x \oplus k'_r)$. K and K' represent two different keys. $K = (k_1, k_2, \ldots, k_r, k_{r+1})$ and $K' = (k'_1, k'_2, \ldots, k'_r, k'_{r+1})$, K' satisfies $k'_j = k_{j+1}$ $(1 \leqslant j \leqslant r+1)$. The round function in $E'_{K'}$ and E_K satisfies $P'_i = P_{i+1}$, $1 \leqslant i \leqslant r+1$.

In Ref. [11], the authors first extended the problem solved by Simon's algorithm and gave a method to find the period of periodic functions up to constant addition. Based on this method of Ref. [11], we introduce a new application of iterative E-M structure ciphers using different round keys and different round functions in Section 5.2.

Problem 2[11] Defining a function ϕ, ϕ: $\{0,1\}^n \rightarrow \{0,1\}^n$, vector $\boldsymbol{s}$, and $\boldsymbol{\gamma} \in \{0,1\}^n$, for $\forall x \in \{0,1\}^n$, that satisfies $\phi\,(\,x \oplus \boldsymbol{s}\,) = \phi(x) \oplus \boldsymbol{\gamma}$, how to find $\boldsymbol{s}$ and $\boldsymbol{\gamma}$?

We consider the differential of ϕ to solve this problem. Defining the differential of ϕ,

$$\boldsymbol{\Delta}_u \phi(x) = \phi(x) \oplus \phi\,(\,x \oplus \boldsymbol{u}\,)\,, \boldsymbol{u} \in \{0,1\}^n.$$

Then for $\forall \boldsymbol{w} \in \text{span}(\boldsymbol{s}, \boldsymbol{u}) = i\boldsymbol{s} \oplus j\boldsymbol{u}$ $(i, j \in \{0,1\})$, and $\forall x \in \{0,1\}^n$, $\boldsymbol{\Delta}_u \phi\,(\,x \oplus \boldsymbol{w}\,) = \boldsymbol{\Delta}_u \phi(x)$.

An error in Ref. [11] needs to be pointed out here. For a fixed $\boldsymbol{u}$, Hosoyamada and Aoki[11] thought that the function $\boldsymbol{\Delta}_u \phi$ was a double-periodic function with the period of $\boldsymbol{s}$ and $\boldsymbol{u}$. But we actually find that $\boldsymbol{\Delta}_u \phi(x)$ is a multi-periodic function with periods $\{\boldsymbol{s}, \boldsymbol{u}, \boldsymbol{s} \oplus \boldsymbol{u}\}$. Let $\boldsymbol{\Delta}_u \phi(x) = \varphi$, for $\forall \boldsymbol{w} \in \text{span}(\boldsymbol{s}, \boldsymbol{u})$ and $\forall x \in \{0,1\}^n$, that satisfies $\varphi\,(x \oplus \boldsymbol{w}) = \varphi(x)$. In other words, φ is a multi-periodic function with three periods $\{\boldsymbol{s}, \boldsymbol{u}, \boldsymbol{s} \oplus \boldsymbol{u}\}$. We can prove the following Lemma similar to Lemmas 2 and 3. The quantum circuit of $\boldsymbol{\Delta}_u \phi(x)$ is shown in Fig. 8.

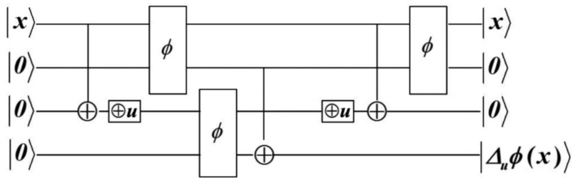


Fig. 8 Quantum circuit of $\boldsymbol{\Delta}_u \boldsymbol{\phi}(x)$.

Lemma 4[11] The function definition is similar to Lemma 2,

$$g : \{0,1\}^{n+1} \rightarrow \{0,1\}^n,$$

$$g(b||x) = \begin{cases} P_{r+1}(E_1(x)) \oplus x, b = 0, \\ E_2(P_1(x)) \oplus x, b = 1. \end{cases}$$

For $\forall \boldsymbol{u}_0 \in \{0,1\}^n \backslash \{0^n\}$ and $\boldsymbol{u} = (\,0||\boldsymbol{u}_0)$, defining the differential function of g,

$$\varphi(x) = \boldsymbol{\Delta}_u g(x) = g(x) \oplus g(x \oplus \boldsymbol{u}).$$

For $\forall x \in \{0,1\}^n$ and $b \in \{0,1\}$, φ is a function that has three periods $\boldsymbol{w}$, $\boldsymbol{w} \in$ {span ($\boldsymbol{s}$ = $(1||k_1)$,$\boldsymbol{u}$)\0}, Here, E_1 and E_2 correspond to E_K and $E'_{K'}$, respectively.

In order to apply Simon's algorithm to obtain the period $\boldsymbol{s}$, we also bound

$$\varepsilon(\boldsymbol{\Delta}_u g; \{\text{span}(\boldsymbol{s}, \boldsymbol{u}) \backslash 0\}) = \max_{t \notin \text{span}(s,u)} \Pr_x[\boldsymbol{\Delta}_u g(x) = \boldsymbol{\Delta}_u g(x \oplus \boldsymbol{t})].$$

The same assumption is made here as in Ref. [7], that both $P_{r+1} \circ E_1$ and $E_2 \circ P_1$ are indistinguishable from random permutations.

Lemma 5[11] For the function $\boldsymbol{\Delta}_u g(x) = \varphi$ defined in Lemma 4, it satisfies

$$\varepsilon(\boldsymbol{\Delta}_u g; \boldsymbol{s}, \boldsymbol{u}) = \max_{t \notin \text{span}(s,u)} [\boldsymbol{\Delta}_u g(x) = \boldsymbol{\Delta}_u g(x \oplus \boldsymbol{t})] < \frac{1}{2}.$$

Lemma 6[11] For the functions g and $\boldsymbol{\Delta}_u g(x) = \varphi$ given by Lemma 4, we assume that there exists a positive number $p_0 < 1$, vector $\boldsymbol{u}_0 \in \{0,1\}^n$, and $\boldsymbol{u} = (\,0||\boldsymbol{u}_0\,)$, such that $\varepsilon\,(\,\varphi; \{\text{span}\,(\boldsymbol{s},\boldsymbol{u})\,\backslash 0\}\,) \leqslant p_0$. Then we can obtain $\boldsymbol{s}$ with probability at least $1 - (2((1+p_0)/2)^c)^n$ by querying the subroutine of Simon's algorithm cn times.

The detailed proof process of Lemmas 4 and 5 can be found in Ref. [11].

In what follows, Sections 4 and 5 take LED-64 and LED-128 as examples and give the results of the security analysis of LED-like block ciphers under classical and quantum attack conditions.

4 Slide Attacks on LED-64

4.1 Classical slide attack on LED-64

In this section, based on the idea of original slide attack, we introduce a differential in the initial round constant of LED-64 to ensure the self-similarity of the cipher, giving an improved slide attack on LED-64. The analysis results show that LED-64-like block ciphers are unable to resist the improved slide attack. The following describes the classic-improved slide attack algorithm for LED-64 and analyzes the indexes of this attack algorithm. Note that, since LED-64 is consisted of 8 steps and each step includes 4 rounds, we consider every 4 round operations as a round function $P_i (1 \leqslant i \leqslant r)$, $r = 8$.

4.1.1 Attack Algorithm 1

Step 1 Generate plaintext-ciphertext pairs, by randomly selecting plaintexts and encrypting them to obtain corresponding ciphertexts under the known-plaintext attack setting. For the initial round constant $rc = (rc_6, rc_5, rc_4, rc_3, rc_2, rc_1)$, select its difference $\Delta rc = (0, 0, 1, 1, 1, 1)$. Then, randomly select 2^{32} plaintexts m', encrypt them to obtain corresponding ciphertexts c' under the condition of changing the round constants, thereby generating 2^{64} plaintext pairs (m, m') and the corresponding ciphertext pairs (c, c').

Step 2 Find the slide pair. A slide pair needs to satisfy the following equation:

$$P_1(m \oplus k_1) = m',\ P_9(c) \oplus k_1 = c'.$$

According to the above equation, a key k_1 can be obtained from a plaintext pair (m, m'), and a key k'_1 can be obtained from the ciphertext pair (c, c'), corresponding to the plaintext pair. If $k'_1 = k_1$, then the exact plaintext-ciphertext pair is the desired slide pair, and the key obtained is the correct key.

4.1.2 Complexity analysis of attack Algorithm 1

Theorem 1 As in the case of the classical slide attack on LED-64, the required data complexity is 2^{33}, time complexity is 2^{62}, and the success probability is up to 1.

Proof In terms of data complexity, since the probability of searching for a slide pair is 2^{-64}, 2×2^{32} known-plaintexts are required, and 2^{64} plaintext pairs (m, c) and (m', c') are generated to obtain a slide pair and the correct key.

Time complexity mainly involves two parts of the algorithm. One is the encryption of 2^{33} plaintexts in Step 1. The other is that all plaintext-ciphertext pairs need to perform two steps of encryption. In other words, it needs $2 \times 2^{64}/8 = 2^{62}$ full-round LED-64 encryptions.

In summary, for attack Algorithm 1, the required data complexity is 2^{33} known-plaintexts, time complexity is 2^{62} full-round LED-64 encryptions, and the success probability is 1. ■

This attack shows that the proposition in Ref. [13] that LED-64 can resist a slide attack is incorrect.

4.2 Quantum slide attack on LED-64

Based on the quantum slide attack method given in Section 3.2, we introduce a differential in the round constant, that is, the original LED-64 is represented by E_1, while E_2 represents the altered LED-64 with a change in the initial round constant. Here, $n = 64$, $r = 8$. The function g_1 is defined as follows:

$$g_1 : \{0,1\}^{n+1} \rightarrow \{0,1\}^n,$$

$$g(b||x) = \begin{cases} P_{r+1}(E_1(x)) \oplus x, & b = 0, \\ E_2(P_1(x)) \oplus x, & b = 1. \end{cases}$$

In accordance with Lemma 2, the period $\boldsymbol{s}$ of the function g is $(1||k_1)$. Based on this function, and combined with Lemma 3, attack Algorithm 2 using Simon's algorithm to recover the key k_1 is given below.

4.2.1 Attack Algorithm 2

Step 1 Construct the quantum circuit U_{g_1} suitable for Simon's algorithm as shown in Fig. 6. Among this, $CE_i (i = 1, 2)$ and $CP_j (j = 1, 9)$ are constructed from oracle E_1 and E_2, and from gate $CP_j (j = 1, 9)$ as shown in Fig. 3, respectively.

Step 2 Choose the set $\boldsymbol{L}$ to store the vector $\boldsymbol{y}$. Initially assign $\boldsymbol{L} = 0$. Choose $4n+1$ qubit states, stored in the registers A, B, C, D, and F, in order from top to bottom, respectively. Note that register A only stores 1-bit control information, while each of the remaining registers is an n-bit register. Apply Hadamard transform $H^{\otimes n}$ to the register B to attain an equal superposition state $|\boldsymbol{\phi}_1\rangle$. Repeat the following loop (Steps 3.1 and 3.2) at most $c(n + 1)$ times.

Step 3.1 Make a quantum query to the function to map the state $|\boldsymbol{\phi}_1\rangle$ to $|\boldsymbol{\phi}_2\rangle$, stored in register F. Measure the register F, the register B collapses to the state

$$|\boldsymbol{\phi}_3\rangle = \frac{1}{\sqrt{2}}(\,|z\rangle|\,\mathbf{0}\rangle + |z \oplus s\rangle|\mathbf{0}\rangle).$$

Then, apply $H^{\otimes n+1}$ to the register B to attain

$$|\boldsymbol{\phi}_4\rangle = \frac{1}{\sqrt{2^{n+2}}}\sum(-1)^{y \cdot z}(1 + (-1)^{y \cdot s})|\boldsymbol{y}\rangle.$$

Measure register B at this time to get the random vector $\boldsymbol{y}$ that satisfies $\boldsymbol{y} \cdot \boldsymbol{s} = 0$.

Step 3.2 Use a classical algorithm[19] to determine whether $\boldsymbol{y}$ is linearly independent of the vectors in $\boldsymbol{L}$. If it is independent, define it $\boldsymbol{y}_i$ (i represents the number of elements already stored in $\boldsymbol{L}$, counting from 0), and add $\boldsymbol{y}_i$ to $\boldsymbol{L}$. If $i < n - 1$, return to Step 3.1 and continue to the next loop. If $i = n - 1$, this means we have n linearly-independent vectors in $\boldsymbol{L}$ and we are to break the loop and shift operation to Step 5. On the other hand, if it is dependent, discard it and return to Step 3.1 and continue the next loop.

Step 4 The attack fails if the above loop ends naturally after $c(n+1)$ times.

Step 5 Reaching this step indicates that the attack succeeds. Add the $(n+1)$-th vector $\boldsymbol{y}_n$, which is linearly-independent of the elements of $\boldsymbol{L}$ and not orthogonal to $\boldsymbol{s}$. Such that this constructs a system of $(n+1)$ independent equations satisfying

$$\boldsymbol{y}_i \cdot \boldsymbol{s} = \begin{cases} 0, & i = 0, 1, \ldots, n-1, \\ 1, & i = n. \end{cases}$$

Thus, use the improved Gaussian elimination method[19] to solve the system for $\boldsymbol{s} = (1||k_1)$ and then output the key k_1.

4.2.2 Complexity analysis of attack Algorithm 2

In this paper, the quantum attack algorithms are based on Q_2 model. Therefore, the required complexity of constructing the quantum circuits is not considered in the complexity analysis process. Based on Ref. [4] and Lemma 1, we set $c = 3$ in Step 2, thus operating the loop (Steps 3.1 and 3.2) $3(n+1)$ times. The success probability of attack Algorithm 2 is $1 - (2((1+\frac{1}{2})/2)^3)^n \approx 99.9\%$. Below, we specifically analyze the complexity required for a successful attack.

Referring to the definition in Ref. [11], we describe the assumptions for time complexity of quantum query operations. We treat an n-bit operation or an $n+1$ operation as a unit operation. Following these assumptions, we regard querying the following gates as a unit time:

(1) n-bit and $(n+1)$-bit Hadamard transformation $H^{\otimes n}$, $H^{\otimes n+1}$;

(2) XOR operation on two n-bit strings;

(3) Quantum gate CP_j; and

(4) Encryption oracle E_i,

$$E_i : |\boldsymbol{x}\rangle|\boldsymbol{y}\rangle \mapsto |\boldsymbol{x}\rangle|\boldsymbol{y} \oplus E_i(x)\rangle.$$

This definition is clearly reasonable under the assumption of the Q_2 model. Compared to it, the complexity required for the XOR operation on two 1-bit states is negligible.

Theorem 2 As in the case of the quantum slide attack on LED-64, the probability of success is about 99.9%, the required space complexity is 2^9, time complexity is 2^{12} quantum query and 2^{26} classical computation.

Proof According to attack Algorithm 2 and the quantum circuit of Fig. 6, the attack requires $4n+1$ qubits in total. That is, the space complexity is approximately 2^9. The time complexity of this attack is mainly composed of the quantum query operation complexity and the classical computational complexity.

In regards to the quantum query, inside a loop, Step 3.1 needs to query the Hadamard transformation twice, CP_j three times, CE_i three times, and the XOR operation on two n-bit strings once, over Fig. 6. Among these, a complete CE_i consists of 3 unit operations (see Fig. 3). This sums to 15 unit operations being performed for each iteration of Step 3.1. Attack Algorithm 2 repeats the loop at most about $c(n+1)$ times, therefore the total required time complexity of quantum query operations is $3 \times (64+1) \times 15 \approx 2^{12}$.

The time complexity of the classical computation is mainly determined by Steps 3.2 and 5. Based on the improved Gaussian elimination method in Ref. [19], for each iteration of Step 3.2, we not only need to judge the linear dependence of the vector $\boldsymbol{y}$ and the elements in $\boldsymbol{L}$, but also ensure that the matrix $\boldsymbol{l}$ generated by the set has the simplest form, where $\boldsymbol{l} = [\boldsymbol{y}_0, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{n-1}]^{\mathrm{T}}$. In a loop, the classical computational complexity required for Step 3.2 is about $(n+1)^3$.

In Step 5, according to the $n \times (n+1)$-dimensional matrix $\boldsymbol{l}$, we add the $(n+1)$-th vector $\boldsymbol{y}_n$, which is linearly independent on the elements of $\boldsymbol{L}$ and not orthogonal to $\boldsymbol{s}$, then construct a system of $n+1$ independent equations. Solving the system for $\boldsymbol{s} = (1||k)$, the required classical computational complexity is about $(n+1)^2$.

In total, the time complexity required for the classical computation is $c(n+1) \times (n+1)^3 + (n+1)^2 \approx 2^{26}$.

In summary, for attack Algorithm 2, the probability of success is about 99.9%, the required space complexity is 2^9, and time complexity is 2^{12} quantum query and 2^{26} classical computation. ■

5 Key-Recovery Attacks on LED-128

5.1 Classical slide attack on LED-128

For LED-128, we first analyze its security with the improved classical slide attack method proposed in Section 4.1 under a related-key model. We choose the related-key $k'_j = k_{j+1}$ $(1 \leqslant j \leqslant r)$. That is, for the plaintext pairs (m, m') and the corresponding ciphertext pairs (c, c'), the slide pair needs to satisfy $P_1(m \oplus k_1) = m'$ and $P_{13}(c) \oplus k_2 = c'$. Since k_1 and k_2 are independent 64-bit keys, the probability of recovering the correct key is 2^{-128}. This means the required time complexity of this attack is equal to brute force. In other words, the classical related-key slide attack cannot effectively recover the keys of LED-128. However, quantum related-key attack can evaluate the cipher in polynomial time.

5.2 Quantum related-key attack on LED-128

For LED-128 we introduce a differential in the round constant and choose the related-key to construct E_1 and E_2 by the quantum related-key method described in Section 3.3. In E_1, we use the key K, and the related-key K' is used in E_2, where

$$K = (k_1, k_2, k_1, k_2, k_1, k_2, k_1, k_2, k_1, k_2, k_1, k_2, k_1),$$
$$K' = (k_2, k_1, k_2, k_1, k_2, k_1, k_2, k_1, k_2, k_1, k_2, k_1, k_2).$$

E_1 therefore represents the original LED-128, and E_2 represents the changed LED-128 with altered initial round constants and using the related-key. Define the following function g_2,

$$g_2 : \{0,1\}^{n+1} \rightarrow \{0,1\}^n,$$
$$g_2(b||x) = \begin{cases} P_{13}(E_1(x)) \oplus x, & b = 0, \\ E_2(P_1(x)) \oplus x, & b = 1. \end{cases}$$

For the equation

$$g_2(0||x) = g_2((0||x) \oplus (1||k_1)) \oplus (k_1 \oplus k_2),$$

we know that the period of g_2 is $\boldsymbol{s} = (1||k_1)$, and the constant is $k_1 \oplus k_2$. According to Lemma 4, if

$$\boldsymbol{u}_0 \in \{0,1\}^n \backslash \{0^n\},\ \boldsymbol{u} = (0||\boldsymbol{u}_0)$$

is chosen, the period of $\boldsymbol{\Delta}_u g_2(x) = g_2(x) \oplus g_2(x \oplus \boldsymbol{u})$ is $\boldsymbol{w}, \boldsymbol{w} \in \{\text{span}[(1||k_1), \boldsymbol{u}] \backslash 0\}$. The specific attack algorithm for solving the keys k_1 and k_2 is given by Simon's algorithm and Lemma 6.

5.2.1 Attack Algorithm 3

Step 1 Arbitrarily choose $\boldsymbol{u}_0 \in \{0,1\}^n \backslash \{0^n\}$, and let $\boldsymbol{u} = (0||\boldsymbol{u}_0)$. Construct the circuit for $\boldsymbol{\Delta}_u g_2$ based on the function g_2.

Step 2 Choose the set $\boldsymbol{L}$ to store the vector $\boldsymbol{y}$, initially assign $\boldsymbol{L} = 0$. Choose $8n + 2$ qubit states, stored in the registers A_1, B_1, C_1, D_1, F_1, A_2, B_2, C_2, D_2, and F_2, in order from top to bottom. Note that registers A_1 and A_2 store only 1-bit control information, while each of the remaining registers is an n-bit register. Apply Hadamard transform $H^{\otimes n}$ to the register B_1 to attain an equal superposition state $|\boldsymbol{\phi}'_1\rangle$. Repeat the following loop (Steps 3.1 and 3.2) at most $c(n + 1)$ times.

Step 3.1 Make a quantum query to the quantum circuit $\boldsymbol{\Delta}_u g_2$ to map the state $|\boldsymbol{\phi}'_1\rangle$ to $|\boldsymbol{\phi}'_2\rangle$, stored in register F_2. Measure register F_2, register B_1 collapses to the state $|\boldsymbol{\phi}'_3\rangle$. Then, apply $H^{\otimes n+1}$ to register B_1 to attain

$$|\boldsymbol{\phi}'_4\rangle = \frac{1}{\sqrt{2^{n+2}}} \sum (-1)^{y \cdot z} (1 + (-1)^{y \cdot s} + (-1)^{y \cdot u} + (-1)^{y \cdot (u \oplus s)}) |\boldsymbol{y}\rangle.$$

Measure register B_1 at this time to get the random vector $\boldsymbol{y}$, that satisfies

$$\boldsymbol{y} \cdot \boldsymbol{w} = 0,\ \boldsymbol{w} \in \{\text{span}[\,\boldsymbol{s}, \boldsymbol{u}\,] \backslash 0\}.$$

Step 3.2 The procedure is little different from Step 3.2 of attack Algorithm 2. Using a classical algorithm[19] to determine whether $\boldsymbol{y}$ is linearly independent of the vector in $\boldsymbol{L}$. If it is independent, define it $\boldsymbol{y}_i$ (i represents the number of elements already stored in $\boldsymbol{L}$, counting from 0), and add $\boldsymbol{y}_i$ to $\boldsymbol{L}$. If $i < n - 2$, return to Step 3.1 and continue to the next loop. If $i = n - 2$, this means we have $n - 1$ linearly-independent vectors in $\boldsymbol{L}$, we break the loop and shift operation to Step 5. On the other hand, if it is dependent, discarded it and return to Step 3.1 and continue the next loop. Note that we only construct $\boldsymbol{L}$ with $n - 1$ linearly-independent vectors because we need to solve the multiple non-zero solutions of the system in Step 5.

Step 4 The attack fails if the above loop ends naturally after $c(n + 1)$ times.

Step 5 Reaching this step indicates that the attack succeeds. Add the n-th vector $\boldsymbol{y}_{n-1}$, which is linearly-independent of the elements of $\boldsymbol{L}$ and not orthogonal to $\boldsymbol{w}$. such that this constructs a system of n independent equations satisfying

$$\boldsymbol{y}_i \cdot w = \begin{cases} 0, & i = 0, 1, \ldots, n - 2, \\ 1, & i = n - 1. \end{cases}$$

Thus, use the improved Gaussian elimination method[19] to solve the system for $\boldsymbol{w}$, then find $\boldsymbol{V} = \text{span}(\boldsymbol{u}, \boldsymbol{s})$, obtain k_1, and calculate k_2. Output the keys k_1 and k_2.

5.2.2 Complexity analysis of attack Algorithm 3

The same as Section 4.2.2, the required complexity of constructing the quantum circuits is not considered in the complexity analysis process. Based on Refs. [4, 11] and Lemma 6, we set $c = 3$ in Step 2, thus repeating the loop (Steps 3.1 and 3.2) $3(n + 1)$ times. The success probability of attack Algorithm 3 is $1 - (2((1 + \frac{1}{2})/2)^3)^n \approx 99.9\%$. The required complexity is mainly divided into two parts: space complexity and time complexity.

Theorem 3 As in the case of the quantum related-key attack on LED-128, the probability of success is about 99.9%, the required space complexity is 2^{10}, and time complexity is 2^{14} quantum query and 2^{26} classical computation.

Proof According to the above attack procedure, the attack requires $8n + 2$ qubits in total. That is, the space complexity is not more than 2^{10}. The time complexity is mainly composed of the quantum query operation complexity and the classical computational complexity.

In regards to the quantum query, Step 3.1 performs the Hadamard transformation twice, the unit quantum gate operation 27 times and the unit XOR operation 17 times per iteration (see Fig. 9). The full attack algorithm repeats the loop at most about $c(n+1)$ times. Therefore, the time complexity of quantum query operation is $3 \times (64 + 1) \times 46 \approx 2^{14}$.

The time complexity of the classical computation is mainly determined by Steps 3.2 and 5. For Step 3.2, the classical computational complexity required is same as the attack Algorithm 2 which is about $(n + 1)^3$ in a loop. In Step 5, we find the vector space $\boldsymbol{V} = \text{span}\,(\boldsymbol{u}, \boldsymbol{s})$ and then calculate k_1 and k_2. For this, the required classical computational complexity is not more than $(n + 1)^3$.

In total, the time complexity required for the classical computation is $c(n + 1)^4 + (n + 1)^3 \approx 2^{26}$.

In summary, for attack Algorithm 3, the success probability is about 99.9%, the required space complexity is 2^{10}, time complexity is 2^{14} quantum query and 2^{26} classical computation. ■

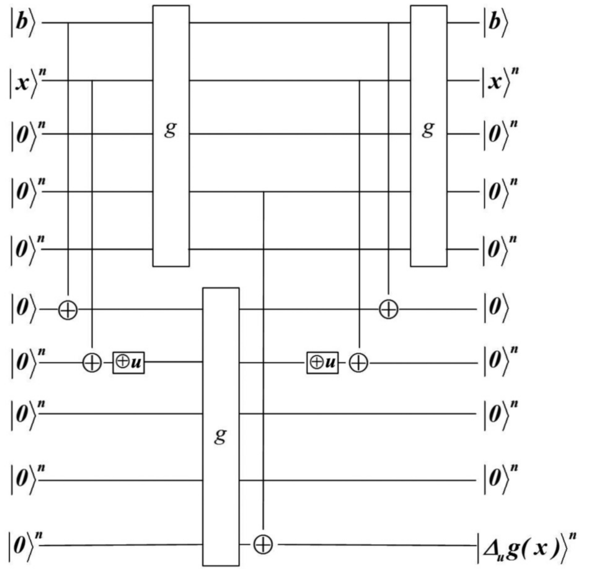


Fig. 9 Quantum circuit of $\Delta_u g(x)$.

6 Conclusion

In this paper, through the study of the properties of LED-like block ciphers, we use the improved classical slide attack and quantum attack methods to perform key-recovery attacks on LED-like block ciphers. Under the classical settings, the adversary can use the attack Algorithm 1 given in this paper to recover the master key of LED-64 with the success probability 1 and the complexity below brute-force. However, this attack method is not applicable to LED-128. Under the conditions of quantum computers, the adversary can give quantum key-recovery attacks on LED-64 and LED-128 in polynomial time and the success probability is both 99.9%. For the quantum attack on LED-64, the required space complexity is 2^9, time complexity is 2^{12} quantum query and 2^{26} classical computation. For the quantum attack on LED-128, the required space complexity is 2^{10}, time complexity is 2^{14} quantum query and 2^{26} classical computation. The above attacks show that the method of adding round constants has a certain influence on the safety of a cipher, and symmetric cryptographic algorithms that are proved to be secure under classical settings are not necessarily secure under quantum computing conditions.

However, there are certain flaws in the study presented in this paper, which point to the areas of focus for future research. Two such openings are:

(1) If a cipher uses a fixed round constant in each round and there is no correlation between the round constants, resulting in the round functions being different in each round of the cipher, the idea of slide attack is not then applicable. The question thus arises of how to perform a quantum key-recovery attack.

(2) In addition, there is a need to consider the effect of the whitening keys on the security of the ciphers under a quantum attack. The analysis of the FX structure block cipher proposed by Leurent et al.[12], combining Grover's algorithm and Simon's algorithm, provides a research path. Learning from this idea, combined with a variety of quantum computing algorithms, it is worth studying whether it is possible to design an effective quantum key-recovery attack method for Feistel, ARX structural block ciphers.

Acknowledgment

This work was supported by the Foundation of Science and Technology on Information Assurance Laboratory (No. KJ-17-003).

References

[1] R. L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM,* vol. 21, no 2, pp.120–126, 1978.

[2] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete loga-rithms on a quantum computer, *SIAM Review,* vol. 41, no 2, pp. 303–332, 1999.

[3] L. K. Grover, A fast quantum mechanical algorithm for database search, arXiv preprint quant-ph/9605043, 1996.

[4] D. R. Simon, On the power of quantum computation, *SIAM Journal of Computing,* vol. 26, no 5, pp. 1474–1483, 1997.

[5] M. Kaplan, G. Leurent, A. Leverrier, and M. N. Plasencia, Breaking symmetric cryptosystems using quantum period finding, in *Proceedings of Annual International Cryptology Conference (CRYPTO 2016),* Santa Barbara, CA, USA, 2016, pp. 207–237.

[6] S. Even and Y. Mansour, A construction of a cipher from a single pseudorandom permutation, *Journal of Cryptology,* vol. 10, no 2, pp.151–161, 1997.

[7] G. Leurent, M. Kaplan, A. Leverrier, and M. N. Plasencia, Quantum differential and linear cryptanalysis, arXiv preprint arXiv:1510.05836, 2015.

[8] H. Kuwakado and M. Morii, Quantum distinguisher between the 3-round Feistel cipher and the random permutation, in *Proceedings of 2010 IEEE International Symposium on Information Theory,* Austin, TX, USA, 2010, pp. 2682–2685.

[9] H. Kuwakado and M. Morii, Security on the quantum-type Even-Mansour cipher, in *Proceedings of 2012 International Symposium on Information Theory and its Applications,* Honolulu, HI, USA, 2012, pp. 312–316.

[10] M. Roetteler and R. Steinwandt, A note on quantum related-key attacks, *Information Processing Letters,* vol. 115, no 1, pp. 40–44, 2015.

[11] A. Hosoyamada and K. Aoki, On quantum related-key attacks on iterated even-mansour ciphers, *IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Sciences,* vol. 102, no. 1, pp. 27–34, 2019.

[12] G. Leurent and A. May, Grover meets Simon-quantumly attacking the FX-construction, in *Proceedings of International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2017),* Hong Kong, China, 2017, pp. 161–178.

[13] A. Biryukov and D. Wanger, Slide attacks, in *Proceedings of International Workshop on Fast Software Encryption (FSE-1999)*, Rome, Italy, 1999, pp. 245–259.

[14] M. Tang, M. X. Luo, J. F. Zhou, Z. Yang, Z. P. Guo, F. Yan, and L. Liu, Side-channel attacks in a real scenario, *Tsinghua Science and Technology,* vol. 23, no 5, pp. 586–598, 2018.

[15] J. Guo, T. Peyrin, A. Poschmann, and M. Robshaw, The LED block cipher, in *Proceedings of 2011 International Workshop on Cryptographic Hardware and Embedded Systems (CHES 2011),* Nara, Japan, 2011, pp. 326–341.

[16] J. Daemen and V. Rijmen, Advanced encryption standard, Springer Science & Business Media, 2013.

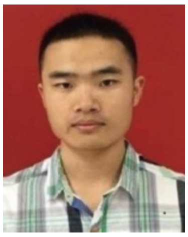
[17] J. Borghoff, A. Canteaut, T. Güneysu, E. B. Kavun, M. Knezevic, L. R. Knudsen, and P. Rombouts, PRINCE—A low-latency block cipher for pervasive computing applications, in *Proceedings of International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2012),* Beijing, China, 2012, pp. 208–225.

[18] M. Grassl, B. Langenberg, M. Roetteler, and R. Steinwandt, Applying Grover's algorithm to AES: Quantum resource estimates, in *Proceedings of Post-Quantum Cryptography (PQCrypto 2016),* Fukuoka, Japan, 2016, pp. 29–43.

[19] M. Loceff, A course in quantum computing, http://creativecommons.org/licenses/by-nc-nd/4.0/, 2018.

Linhong Xu is a postgraduate at the Information Science and Technology Institute, Zhengzhou, China. He received the bachelor degree in cryptography from the Information Science and Technology Institute, Zhengzhou, China, in 2016. His main research interests include cryptography and information security.

Jiansheng Guo is currently a professor at the Information Science and Technology Institute, Zhengzhou, China. He received the PhD degree in cryptography from the Information Science and Technology Institute, Zhengzhou, China, in 2004. His main research interests include cryptography, quantum information, and security.



Jingyi Cui is a PhD candidate at the Information Science and Technology Institute, Zhengzhou, China. He received the master degree in cryptography from the Information Science and Technology Institute, Zhengzhou, China, in 2017. His main research interests include cryptography and information security.

Mingming Li is a postgraduate at the Information Science and Technology Institute, Zhengzhou, China. He received the bachelor degree from Xinjiang University in 2016. His main research interests include cryptography and information security.