Sun et al. *EPJ Quantum Technology* (2026) 13:35
https://doi.org/10.1140/epjqt/s40507-026-00481-3

EPJ.org

EPJ Quantum Technology

RESEARCH Open Access

Quantum related-key attacks on Feistel-like ciphers

Hong-Wei Sun[1*], Long Zhang[2], Dan-Dan Li[3], Zhen-Qiang Li[4], Rong-Xue Xu[1*] and Ke-Jia Zhang[1*]

*Correspondence:
sunhw@hlju.edu.cn;
2024025@hlju.edu.cn;
zhangkejia@hlju.edu.cn
[1]School of Computer and Big Data (School of Cybersecurity), Heilongjiang University, Harbin 150080, China
Full list of author information is available at the end of the article

Abstract

Quantum attacks employing superposition queries (Q2 model) have been demonstrated to compromise numerous classically secure block-cipher constructions. These attacks embed the target encryption structure into a period-finding problem, which dedicated quantum algorithms, including the Simon and BV algorithms, can solve. After restoring the secret state, attackers can launch an attack by recovering the key (the secret state contains key information) or distinguishing it from random permutations.

In this paper, we investigate the quantum security of Feistel-like ciphers in a related-key setting. For several constructions, we demonstrate how to define a periodic function that can be evaluated using the Simon algorithm with only $O(n)$ quantum queries to the encryption oracle, where n represents the block size. This enables secret-state recovery attacks on Type-1/2 generalized Feistel schemes, as well as key-recovery attacks on the Feistel-FK and Misty L-KF constructions. All these attacks achieve an exponential reduction in query complexity compared to the best known classical or quantum methods.

Keywords: Block cipher; Related-key attack; Quantum cryptanalysis; Simon algorithm; Circuit implementation

1 Introduction

Modern cryptography relies on computational complexity as its foundation. However, the advent of powerful quantum computers poses a significant threat. For instance, the Shor algorithm [1] can efficiently factor large numbers, rendering common public key cryptosystems like RSA [2] and ECC [3] vulnerable. Similarly, the Grover algorithm [4] accelerates key searches, effectively halving key lengths. Just as the introduction of electronic computers transformed classical cryptography into modern cryptography, quantum computers promise to revolutionize the field further. Therefore, studying the resilience of classical cryptosystems against quantum computing is crucial for designing quantum-resistant cryptosystems, which is paramount for the advancement of cryptography.

In addition, other quantum algorithms, including the Simon algorithm [5], BV algorithm [6], and Grover-meets-Simon algorithm [7], also pose a threat to the security of block cipher [8–14]. Following the concepts of pseudo-random function (PRF) and pseudo-

Springer

random permutation (PRP) security in quantum settings [15], these attacks are primarily categorized into two models based on their ability to restrict the adversary [16–18]: the standard security model (Q1 model) and the quantum security model (Q2 model).

Standard security: Suppose there's no efficient quantum algorithm capable of distinguishing a block cipher from a PRP (or PRF) with classical queries alone. In that case, the block cipher is considered standard secure (called the Q1 model).

Quantum security: Suppose there's no efficient quantum algorithm capable of distinguishing a block cipher from a PRP (or PRF) with quantum queries. In that case, the block cipher is considered quantum secure (called the Q2 model).

The distinction between the two models primarily lies in data collection methods. In the Q1 setting [17, 19, 20], the adversary gathers data through classical means (i.e., online classical query) and then processes it using quantum computing (i.e., offline quantum computation). In the Q2 setting [21, 22], the adversary has direct access to the quantum oracle $O_E : \Sigma_{x,y}\lambda_{x,y}|x\rangle|y\rangle \mapsto \Sigma_{x,y}\lambda_{x,y}|x\rangle|E(x) \oplus y\rangle$, where x and y are arbitrary n-bit strings. The Q1 attack aligns more closely with the primary threat model of post-quantum cryptography, known as the "harvest now, decrypt later" (HNDL) threat model. Although the Q2 attack lacks practical applications, it is still crucial for evaluating quantum security and designing related solutions. It can serve as a basis or motivation for creating improved Q1 attacks, or as an impossibility result that shows any valid security proof must consider adversaries capable of making classical queries to the scheme. In this paper, we examine the security of Feistel-like block cipher structures in the Q2 model. This model already represents a compelling adversarial setting, and adding the related-key assumption further increases the attacker's capabilities, resulting in a "worst-case" analysis. Although such access may be impractical, this perspective reveals algebraic properties of the structure that could remain hidden in classical or weaker models. This attack provides more than a quadratic speedup over the best known classical methods in some cases (e.g., Feistel constructions [13, 14] and certain MAC designs [23]), and can even break classically secure schemes.

The related-key attack [24, 25] is a potent cryptanalysis technique that exploits known ciphertext and corresponding plaintext to extract keys or other key-related information from the targeted encryption algorithm. With the formidable computing capabilities of quantum computers, cryptography researchers have begun employing quantum algorithms for related-key analysis, yielding numerous intriguing results. Roetteler and Steinwandt [26] initially introduced the concept of quantum related-key cryptanalysis. They highlighted that key information could be effectively extracted by accessing the quantum related-key oracle with a few plain-ciphertext pairs, requiring only $O(n)$ quantum queries, where n is the block size. Later, Hosoyamada and Aoki [27] extended this concept to a more generalized key-pair relationship, termed the "sliding relationship", demonstrating the effectiveness of the Simon algorithm in recovering the key of the 2-round iteration Even-Mansour structure with the same query complexity, $O(n)$. In 2021, Xie and Yang [28] introduced a novel quantum-related key attack based on the BV algorithm, requiring $O(n^2)$ quantum queries. Recently, Sun et al. [8] further expanded on this by showcasing that the BV algorithm can effectively target two types of iterative structures with independent subkeys: the iterative Even-Mansour structure and the i-round Feistel structure, both with complexity $O(n)$.

The quantum related-key attacks described above exploit the strong algebraic structure present in many popular block-cipher designs. This allows attackers to construct a periodic function based on the target encryption algorithm and employ quantum algorithms, such as the Simon algorithm, BV algorithm, etc., to recover this period. Once the secret state is restored, attackers can launch an attack by recovering the key (the secret state contains key information) or distinguishing it from random permutations. Compared to classical attacks, this type of Simon-based related-key attack provides exponential acceleration in query complexity. Recent works have demonstrated that certain block cipher structures, such as Even-Mansour construction, could be broken in the quantum related-key setting. However, the security of other block ciphers under this attack model remains uncertain.

Our contributions. In this paper, we delve into the application of quantum algorithms for breaking symmetric cryptographic primitives, focusing on the quantum security of Feistel-like cryptographic structures within related-key contexts.

1. For Type-1/2 generalized Feistel schemes (GFS) with independent subkeys, we present a secret-state recovery attack in a related-key setting, with a query complexity of $O(n)$. By gaining effective access to the related-key oracle of the target encryption algorithm, we extend the effective attack rounds to multiple rounds. Particularly, in cases where the round function is reversible, partial keys of the target structure can be recovered by accessing two related-key oracles.

   Using this method, we propose an efficient secret-state recovery attack against the 24-round CAST-256 algorithm using the Simon algorithm. We also employ the Grover-meets-Simon algorithm to achieve a more efficient key recovery attack, with complexity $2^{37(r-24)/2}$, where r represents the attack rounds. That is, our attack complexity is reduced by a factor $2^{129.5}$ compared to existing work [10].
2. For Feistel variants with independent subkeys: Feistel-FK construction and Misty L-KF construction, we present a key-recovery attack in a related-key setting, with a query complexity of $O(n)$. It achieves exponential speedup compared to some related studies [21, 29, 30].

Organization. The structure of the paper is as follows: Sect. 2 covers essential background information, including Feistel-like structures, quantum circuits, and quantum algorithms used in our attacks. Section 3 presents quantum secret-state recovery attacks on Type-1/2/3 generalized Feistel schemes. Section 4 proposes quantum key-recovery attacks on two Feistel variants: Feistel-FK construction and Misty L-KF construction. Section 5 summarizes the paper.

2 Preliminaries

We define F_2 as the prime field with elements 0 and 1, denoted $\{0,1\}$. The n-dimensional vector space over F_2 is represented as F_2^n, equivalent to $\{0,1\}^n$. The symbol "$\oplus$" signifies XOR (addition in F_2^n), while "$\cdot$" denotes the scalar product of bit-strings viewed as n-bit vectors.

2.1 Feistel-like schemes

Generalized Feistel schemes (GFS) extend the classic Feistel structure and are widely used in block cipher design. By dividing data into multiple sub-blocks and performing complex permutation and mixing operations on these sub-blocks in each encryption round, GFS

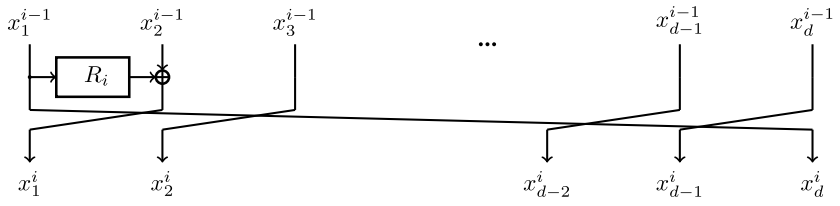


Figure 1 The *i*-th round of Type-1 GFS

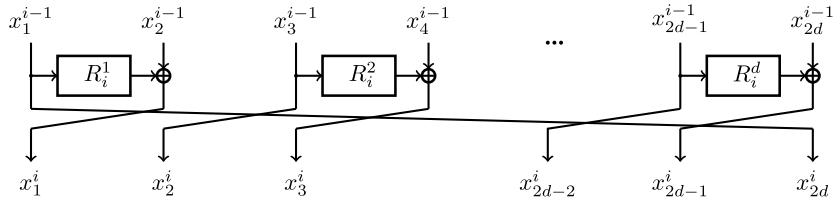


Figure 2 The *i*-th round of Type-2 GFS

enhances the security and flexibility of encryption algorithms. GFS holds a significant position in modern cryptography, providing a robust foundation for designing efficient and secure block cipher algorithms.

At CRYPTO 1989, Zheng et al. [31] introduced three general frameworks: Type-1, Type-2, and Type-3 generalized Feistel schemes. These frameworks are based on Feistel-like structures with more branches and different operations. They also demonstrated that, with independently and randomly selected round functions, a $(2d-1)$-round d-branch Type-1 GFS, a $(2d+1)$-round $2d$-branch Type-2 GFS, and a $(d+1)$-round d-branch Type-3 GFS are pseudorandom functions.

Type-1 GFS is illustrated in Fig. 1, with its round function size being 1/4 of the block length. The CAST-256 algorithm [32] is based on this structure. The i-th round Type-1 GFS can be represented as

$$x_1^i \leftarrow R_i(x_1^{i-1}) \oplus x_2^{i-1}, x_2^i \leftarrow x_3^{i-1}, x_3^i \leftarrow x_4^{i-1}, \ldots, x_d^i \leftarrow x_1^{i-1} \tag{1}$$

Type-2 GFS is depicted in Fig. 2, where two branches enter the round function simultaneously. These round functions may be identical or different. The CLEFIA [33] and RC6 [34] algorithms are designed based on this structure. The i-th round of Type-2 GFS can be represented as

$$x_1^i \leftarrow R_i^1(x_1^{i-1}) \oplus x_2^{i-1}, x_2^i \leftarrow x_3^{i-1}, \ldots, x_{2d}^i \leftarrow x_1^{i-1} \tag{2}$$

Type-3 GFS is shown in Fig. 3 and requires more sub-round functions. These round functions may be the same or different. The AEGIS algorithm [35] is based on an expansion of this structure. The i-th round of Type-3 GFS can be represented as

$$x_1^i \leftarrow R_i^1(x_1^{i-1}) \oplus x_2^{i-1}, x_2^i \leftarrow R_i^2(x_2^{i-1}) \oplus x_3^{i-1}, \ldots, x_d^i \leftarrow x_1^{i-1} \tag{3}$$

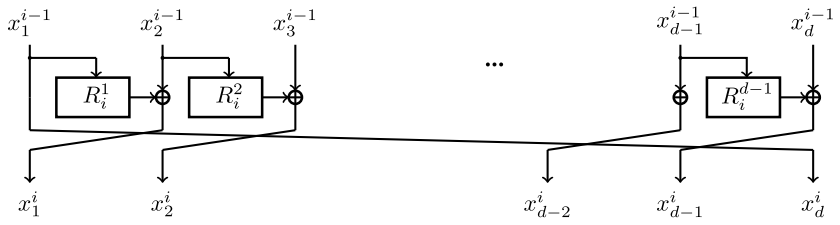
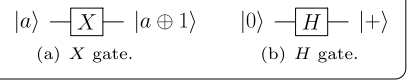
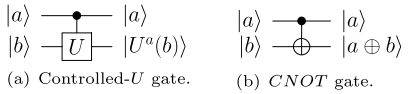


Figure 3 The *i*-th round of Type-3 GFS



(a) X gate. (b) H gate.

Figure 4 One-qubit gate



(a) Controlled-U gate. (b) $CNOT$ gate.

Figure 5 Two-qubit gate

For Type-1/2/3 GFS, the decryption process must strictly follow the reverse order of the encryption process. Decryption mainly relies on applying round functions and XOR operations in reverse.

2.2 Quantum gates

All quantum operations correspond to a unitary matrix, often represented by U. Any unitary matrix constitutes a valid quantum operation. After the quantum operation, the final state is obtained by multiplying the unitary matrix by the initial state vector: $|\psi_1\rangle = U|\psi_0\rangle$. Since unitary matrices are linear, quantum operations can be parallelized.

Among qubit operations, the most fundamental are called logic gates. These gates are categorized based on the number of qubits they act on one-bit gates, two-bit gates, three-bit gates, etc. Common one-bit gates include the X gate and the *Hadamard* gate (see Fig. 4):

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = |0\rangle\langle 1| + |1\rangle\langle 0| \tag{4}$$

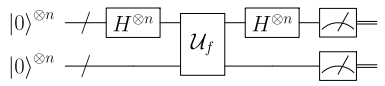
$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\langle 0| + \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\langle 1| \tag{5}$$

The most commonly used two-bit gate is the controlled-U gate (see Fig. 5), defined as

$$U_c = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes U \tag{6}$$

where I and U are one-bit gates, with I being the identity operation. In the controlled-U gate, the first qubit is the control bit, and the second is the target bit. The operation on the target bit depends on whether the control bit is $|0\rangle$ or $|1\rangle$. A common controlled-U gate is the $CNOT$ gate, which operates as follows: when the control bit is $|0\rangle$, the target bit remains unchanged; when the control bit is $|1\rangle$, the target bit undergoes the X gate

Figure 6 Simon algorithm



operation. The matrix representation of the *CNOT* gate is

$$U_{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{7}$$

By combining a few basic gates, we can approximate any unitary operation with arbitrary precision. The set $\{H, T, CNOT\}$ is theoretically sufficient to construct a universal gate set, such as the Toffoli gate. For practical fault tolerance and implementation, the sets $\{H, T, S, CNOT\}$ or $\{H, S, CNOT, Toffoli\}$ are generally used as universal gate sets in quantum computing. Any unitary operation can be decomposed into a combination of these basic gates. For instance, a unitary operation can be broken down into a product of two-level gates, which can then be further decomposed into combinations of *CNOT* and one-bit gates, and one-bit gates can be decomposed into combinations of *Hadamard* and T gates.

2.3 Quantum algorithms

Next, we review the Simon algorithm and the Grover-meets-Simon algorithm used in this paper. For a more detailed introduction, please refer to [5, 7, 14].

Simon algorithm. The Simon algorithm (see Fig. 6) finds the period of a periodic function using quantum computing steps. For this problem (called Simon's problem), the complexity of classical algorithms is exponential, while the Simon algorithm solves it in polynomial time. This demonstrates the significant speed advantage of quantum algorithms over classical ones for specific problems.

The Simon problem is as follows:

Simon problem [5]: Given a function $f : \{0,1\}^n \to \{0,1\}^n$, assume the function has a unique period $s \in \{0,1\}^n$. This means for any $(x, y) \in \{0,1\}^n$, we have

$$[f(x) = f(y)] \Leftrightarrow [x \oplus y \in \{0, s\}]$$

The objective is to find the period s given access to the function.

Notably, the above function f is a strict 2-to-1 function, where each function value corresponds to two different input values (i.e. Simon's promise condition). Traditionally, this problem is tackled using the classical collision search method, with a query complexity of $O(2^{n/2})$. However, the Simon algorithm introduces quantum superposition queries, reducing the query complexity to $O(n)$. In the subroutine of the Simon algorithm (Algorithm 1), the first execution yields a vector y_1 orthogonal to s, and the second yields another vector y_2 also orthogonal to s (many vectors y satisfy this condition). By repeating this step $O(n)$

Algorithm 1 Quantum Subroutine of Simon's Algorithm

Input: n; $U_f = |x\rangle|0\rangle \mapsto |x\rangle|f(x)\rangle$

Output: $y \perp s$

1:	Start in the state	$\triangleright\, \|0\rangle\|0\rangle$
2:	Apply a H transform	$\triangleright\, \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} \|x\rangle\|0\rangle$
3:	Query U_f	$\triangleright\, \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} \|x\rangle\|f(x)\rangle$
4:	Measure the output register $R2$	$\triangleright\, \frac{1}{\sqrt{2}}(\|z\rangle + \|z \oplus s\rangle)$
5:	Apply another H transform	$\triangleright\, \frac{1}{\sqrt{2}} \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} (-1)^{y\cdot z}(1 + (-1)^{y\cdot s})\|y\rangle$
6:	Measure the input register $R1$	$\triangleright\, y \perp s$

times, $(n-1)$ independent vectors y orthogonal to s can be obtained with high probability, leading to the unique non-zero solution s. Thus, for Simon's problem, the Simon algorithm efficiently finds the period s of the above 2-to-1 function in polynomial time.

In the cryptanalysis model, the accessible periodic function f does not always strictly satisfy Simon's promise (besides the period, the function f also has collisions). This means the equality of the function value does not simply correspond to the XOR result of inputs being equal to s. In this case, the problem condition ($f(x) = f(y)$ if and only if $x \oplus y \in \{0, s\}$) is relaxed to: for any input x, $f(x \oplus s) = f(x)$ is satisfied (usually considering the case where the collision ratio is small). To quantify how far the function is from satisfying Simon's promise, Kaplan et al. [14] introduce a parameter (the maximum collision ratio):

$$\varepsilon(f, s) = \max_{t\in\{0,1\}^n\setminus\{0,s\}} \frac{\{x : f(x) = f(x \oplus t)\}}{|x|} \tag{8}$$

where $\varepsilon(f, s)$ is closer to 1, the collision t is closer to the period; when $\varepsilon(f, s)$ is smaller, the impact of the collision on the period is reduced. Specifically, Kaplan has proved the following theorem.

Theorem 1 [14] *When $\varepsilon(f, s) \le p_0 < 1$, the Simon algorithm can still successfully recover the period s with a probability of at least $1 - (2(\frac{1+p_0}{2})^c)^n$ after cn quantum queries.*

In particular, when the parameter $c \ge 3/(1-p_0)$ is large enough, the probability of failure decreases exponentially with n. Theorem 1 shows that even in the presence of collisions, the period s can still be recovered after executing the Simon algorithm cn times. The recovered period s must be verified using several sets of inputs x and $x \oplus s$. If the verification fails, the Simon algorithm is re-executed until the correct period is found.

Grover-meets-Simon algorithm. Leander et al. [7] proposed the Grover-meets-Simon algorithm, which breaks FX construction with whitening keys in essentially the same complexity as Grover's algorithm breaks the underlying block cipher. This indicates that using whitening keys does not significantly enhance security in a quantum setting.

The Grover-meets-Simon problem is as follows.

Grover-meets-Simon problem (adapted from [5]): Given a function $f : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}^d$ and promise that there exists $u \in \{0,1\}^m$ such that the function $f(u, \cdot)$ has a unique period s_u (called hidden periodic function). The objective is to find a set of tuples $(u, s_u) \in U_s$, where $U_s := \{(u, s_u) : u \in \{0,1\}^m, s_u$ is the period of the function $f(u, \cdot)\}$.

Algorithm 2 Grover-meets-Simon Algorithm

Input: m, n, d; $U_f = |u\rangle|x\rangle|0\rangle \mapsto |u\rangle|x\rangle|f(u,x)\rangle$

Output: (u, s_u)

1: Start $|0\rangle$

2: Apply H $\triangleright |\psi\rangle = \frac{1}{\sqrt{2^m}} \sum_{x\in\{0,1\}^m} |x\rangle = \cos(\theta)|\psi_0\rangle + \sin(\theta)|\psi_1\rangle$

3: Apply Grover iteration $(2|\psi\rangle\langle\psi| - I) \cdot U_f$ $\triangleright \cos(3\theta)|\psi_0\rangle + \sin(3\theta)|\psi_1\rangle$

4: Note that U_f is a unitary operator that takes u as input, and tests whether $f(u,x)$ has a hidden period in superposition (see Algorithm 3 for details). $\triangleright |u\rangle|b\rangle \xrightarrow{U_f} |u\rangle|b \oplus r\rangle$

5: After $O(2^{m/2})$ Grover iterations, measure the index u $\triangleright u$

6: Apply Simon's algorithm on $f(u,x)$ $\triangleright s_u$

Algorithm 3 Test Procedure

Input: $|u\rangle|b\rangle$

Output: $|u\rangle|b \oplus r\rangle$

1: Start $\sum_{u\in\{0,1\}^m} |u\rangle|0\rangle|0\rangle|b\rangle$

2: Apply H $\triangleright \sum_{u\in\{0,1\}^m} |u\rangle \sum_{x\in\{0,1\}^n} |x_1\rangle \cdots |x_{cn}\rangle|0\rangle|b\rangle$

3: Apply U_f $\triangleright \sum_{u\in\{0,1\}^m} |u\rangle \sum_{x\in\{0,1\}^n} |x_1\rangle \cdots |x_{cn}\rangle|f(u,x_1)\rangle \cdots |f(u,x_{cn})\rangle|b\rangle$

4:

5: Apply H $\triangleright \sum_{u\in\{0,1\}^m} |u\rangle \sum_{x,y\in\{0,1\}^n} (-1)^{\langle x_1\cdot y_1\rangle}|y_1\rangle \cdots (-1)^{\langle x_{cn}\cdot y_{cn}\rangle}|y_{cn}\rangle|f(u,x_1)\rangle \cdots |f(u,x_{cn})\rangle|b\rangle$

6: Compute $d = \dim(Span(u_1, \ldots, u_{cn}))$: if $d \neq n-1$, set $r = 0$; otherwise $r = 1$.

7: Unompute $\triangleright |u\rangle|b \oplus r\rangle$

Table 1 Quantum secret-state recovery attack on Type-1/2 GFS

Goal	Construction	Attacked rounds	Condition	Complexity	Source
SR[1]	Type-1 GFS	$d^2 - d + 1$	-	$O(n)$	[36]
		$d^2 - 1$	-	$O(n)$	[10]
		∞	related key	$O(n)$	Ours
	Type-2 GFS	$2d + 1$	-	$O(n)$	[37]
		∞	related key	$O(n)$	Ours

[1] secret state recovery

Leander et al. incorporated the Simon algorithm into the iterative process of Grover's search. Specifically, Grover's algorithm is used as the outer loop with a query complexity of $O(2^{m/2})$, while Simon's algorithm is the inner loop with a complexity of $O(n)$. The specific process of the algorithm is shown in Algorithm 2.

3 Quantum secret-state recovery attack for Type-1/2 GFS

For the Type-1/2 GFS, we design a dedicated periodic function, allowing the period to be recovered using the Simon algorithm, resulting in a secret-state recovery attack with complexity $O(n)$. Additionally, this attack method can be applied to specific cases, such as a 24-round secret-state recovery attack on the CAST-256 algorithm, extending existing results by 7 rounds. Tables 1 and 2 summarize the secret-state recovery attacks on Type-1/2 GFS in the quantum related-key setting and compare them with existing quantum algorithms.

Table 2 Quantum attack on round-reduced CAST-256 algorithm*

Source	Setting	Distinguisher	Attacked rounds					
			$r = 25$	$r = 26$	$r = 27$	$r = 28$	$r = 29$	$r = 30$
[36]	-	14	$2^{203.5}$	2^{222}	$2^{240.5}$	2^{259}	$2^{277.5}$	2^{298}
[10]	-	17	2^{148}	$2^{166.5}$	2^{185}	$2^{203.5}$	2^{222}	$2^{240.5}$
Ours	related key	24	$2^{18.5}$	2^{37}	$2^{55.5}$	2^{74}	$2^{92.5}$	2^{111}

* Note that for CAST-256, the trivial bound is 2^{128} by Grover algorithm.

3.1 Attack strategy

This section explores related-key attacks [38, 39], where an attacker obtains ciphertexts encrypted with related keys and exploits their relationship for analysis. For instance, knowing a specific difference between two keys, an attacker can observe the corresponding difference in ciphertexts and use this information to launch an attack. Related-key attacks are a powerful cryptanalysis technique, posing a significant threat to cryptosystems sensitive to key relationships [40–42].

Specifically, for the target encryption algorithm E_k, the attacker can query the following two oracles:

- E: Given a plaintext $m \in \{0,1\}^n$ and a selected related-key pair, return the encryption result $E_{k'}(m)$
- D: Given a ciphertext $c \in \{0,1\}^n$ and a selected related-key pair, return the decryption result $D_{k'}(c)$

Here, k and k' represent different keys with a known relationship. The attack is successful if the adversary can recover the secret information by querying these two oracles. In our attack, only the encryption oracle is needed, that is, the adversary accesses the target encryption algorithm in superposition. Formally, our Simon-based attack is as follows:

1. Construct a periodic function based on the target encryption algorithm, in a related-key setting
2. Recover secret information using Simon algorithm

After recovering the secret state, we can conduct secret-state recovery or key-recovery attacks (period contains extractable key information). Notably, the adversary needs effective access to the function's quantum oracle in our attack. Therefore, we must provide quantum circuit implementations of periodic functions for different encryption algorithms.

3.2 Secret-state recovery attack in the related-key setting

Based on the above discussion, we propose an i-round secret-state recovery attack against Type-1/2 GFS with complexity $O(n)$.

Type-1 GFS [31]. The Type-1 GFS enhances the security and flexibility of cryptographic algorithms by introducing more branches and complex mixing operations. It significantly improves the traditional Feistel network and plays an important role in modern block-cipher design. As shown in Fig. 1, the algorithm divides the input state of dn bits into $d \geq 3$ branches, each containing a n-bit sub-block. For each round $j = 1, 2, \ldots, i$, the calculation process is as follows:

$$x_1^i \leftarrow R_i(x_1^{i-1}) \oplus x_2^{i-1}, x_2^i \leftarrow x_3^{i-1}, x_3^i \leftarrow x_4^{i-1}, \ldots, x_d^i \leftarrow x_1^{i-1} \tag{9}$$

To distinguish the i-round Type-1 GFS structure from random permutations, we construct the corresponding periodic function in a related-key setting. Given two related-key oracles, $E_k = E^R(x; k_1, k_2, \ldots, k_i)$ and $E_{k'} = E^R(x; k_2, k_3, \ldots, k_1)$, denoted as E_1 and E_2, we define the following function, with given constants $x_1^0, x_3^0, x_4^0, \ldots, x_d^0$:

$$\begin{aligned} f : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\ (b,x) &\mapsto \begin{cases} E_1(x_1^0, x, x_3^0, x_4^0, \ldots, x_d^0)_{l+1} & b = 0 \\ E_2(x, x_3^0, x_4^0, x_5^0, \ldots, x_1^0)_l & b = 1 \end{cases} \end{aligned} \tag{10}$$

where the related-key pair $k = (k_1, k_2, \ldots, k_i)$ and $k' = (k_2, k_3, \ldots, k_1)$ satisfy the relationship $k'_j = k_{j+1}$. The related-key oracle $E_1(\cdot)_{l+1}$ and $E_2(\cdot)_l$ represent the $(l+1)$-th and l-th branches of $E_1(\cdot)$ and $E_2(\cdot)$ respectively, and $l = 2, 3, \ldots, d$. In particular, this f satisfies $f(0,x) = f(1, x \oplus R_1(x_1^0))$:

$$\begin{aligned} f(0,x) &= E_1(x_1^0, x, x_3^0, x_4^0, \ldots, x_d^0)_{l+1} \\ &= E_2(x \oplus R_1(x_1^0), x_3^0, x_4^0, x_5^0, \ldots, x_1^0)_l \\ &= f(1, x \oplus R_1(x_1^0)) \end{aligned} \tag{11}$$

where the second equation results from $x_l^{i+1} = x_{l+1}^i$ (Eq. (9)). More precisely:

$$\begin{aligned} E_1(x_1^0, x, x_3^0, x_4^0, \ldots, x_d^0)_{l+1} &= E^{i+1}(x)_l \\ &= f_R^{k_1} \circ E_1(x_1^0, x, x_3^0, x_4^0, \ldots, x_d^0)_l \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_1}(x_1^0, x, x_3^0, x_4^0, \ldots, x_d^0)_l \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_2}(x \oplus R_1(x_1^0), x_3^0, x_4^0, x_5^0, \ldots, x_1^0)_l \\ &= E_2(x \oplus R_1(x_1^0), x_3^0, x_4^0, x_5^0, \ldots, x_1^0)_l \end{aligned} \tag{12}$$

where f_R denotes the round function. Then, based on an i-round Type-1 GFS, we construct a periodic function with $s = 1 \| R_1(x_1^0)$, where i can take any value. That is, the above relation holds for all i.

Estimation of $\varepsilon(f,s)$. In this case, the parameter $\varepsilon(f,s) = \max_{t \in \{0,1\}^n \setminus \{0,s\}} \frac{\{x : f(x) = f(x \oplus t)\}}{|x|}$ is bounded with overwhelming probability, assuming that E behaves as a random permutation. We show that $\varepsilon(f,s) < 1/2$ with overwhelming probability. Indeed, if $\varepsilon(f,s) > 1/2$, then there exists $(1,t) \notin s$ such that $Pr_x[f(0,x) = f(1, x \oplus t)] > 1/2$, i.e.,

$$Pr_x \left[\begin{array}{l} E_1(x_1^0, x \oplus t, x_3^0, x_4^0, \ldots, x_d^0)_{l+1} \\ \oplus E_2(x \oplus t, x_3^0, x_4^0, x_5^0, \ldots, x_1^0)_l = 0 \end{array} \right] > 1/2. \tag{13}$$

This condition implies a higher-order differential for $f(b,x)$ with probability greater than $1/2$, which occurs only with negligible probability when E is chosen at random [43]. Therefore, according to Theorem 1, the Simon algorithm can retrieve the period of this function using $O(n)$ queries to the target encryption algorithm. To make our attacks as straightforward as possible, we provide diagrams of circuits that compute the function f. These

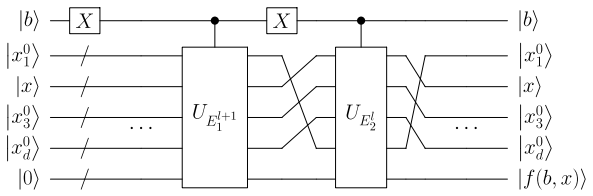


Figure 7 Periodic function for Type-1 GFS

circuits utilize a small number of basic building blocks, as shown in Fig. 7. This enables a successful secret-state recovery attack, where n denotes the block size.

Note that our subsequent attack implicitly assumes $\varepsilon(f,s) \leq 1/2$. This is a standard assumption in the related literature [7, 14] because if $\varepsilon(f,s) > 1/2$, there will be a classical attack. In other words, the probability of this happening in an ideal cryptographic structure is negligible [43].

Complexity of the attack: Our attack uses only $O(n)$ queries to the black box implementing the target encryption structure to find the period of the underlying periodic function via Simon's algorithm, thereby completing the secret-state recovery. Each run of Simon's algorithm uses $2n$ Hadamard gates, two X gates, and two quantum queries to U_E. As a result, the total cost is $O((2n + 2 + 2|U_E|_q)n)$ universal gates and $O(n)$ quantum queries, which is polynomial in complexity. Here, the parameter $|U_E|_q$ denotes the number of universal quantum gates required to implement the quantum circuit of the target encryption algorithm E. Its value depends on the design and concrete implementation of the specific cryptographic scheme.

Type-2 GFS [31]. The GFS is a cryptographic framework for designing block ciphers. The Type-2 GFS is a specific variation within this framework. A key feature of this structure is that each sub-block update depends on the two adjacent sub-blocks, enhancing both diffusion and confusion. As shown in Fig. 2, the input state is divided into four or more sub-blocks. The processing steps for each round are as follows:

$$x_1^i \leftarrow R_i^1(x_1^{i-1}) \oplus x_2^{i-1}, x_2^i \leftarrow x_3^{i-1}, \ldots, x_{2d}^i \leftarrow x_1^{i-1} \tag{14}$$

To distinguish the i-round Type-2 GFS structure from random permutations, we construct the corresponding periodic function in a related-key setting. Given two related-key oracles, $E_k = E^R(x; k_1, k_2, \ldots, k_i)$ and $E_{k'} = E^R(x; k_2, k_3, \ldots, k_1)$, denoted as E_1 and E_2, we define the following function, with a given random constant α:

$$f : \{0,1\} \times \{0,1\}^{dn} \to \{0,1\}^n$$
$$(b,x) \mapsto \begin{cases} E_1(\alpha, x_1, \alpha, x_2, \ldots, \alpha, x_d)_{l+1} & b = 0 \\ E_2(x_1, \alpha, x_2, \alpha, \ldots, x_d, \alpha)_l & b = 1 \end{cases} \tag{15}$$

where $x = (x_1, x_2, \ldots, x_d)$. Here, the related-key pair $k = (k_1, k_2, \ldots, k_i)$ and $k' = (k_2, k_3, \ldots, k_1)$ satisfy the relationship $k'_j = k_{j+1}$, where $k_j = (k_j^1, k_j^2, \ldots, k_j^d)$, $j = (1, 2, \ldots, i)$. The related-key oracle $E_1(\cdot)_{l+1}$ and $E_2(\cdot)_l$ represent the $(l+1)$-th and l-th branches of $E_1(\cdot)$ and

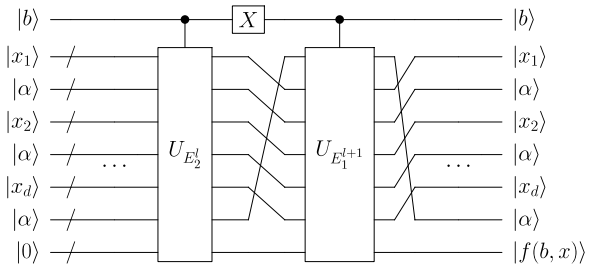


Figure 8 Periodic function for Type-2 GFS

$E_2(\cdot)$ respectively, and $l = 2,4,\ldots,2d$. In particular, this f satisfies $f(0,x) = f(1, x \oplus (R_1^1(\alpha)\|R_1^2(\alpha)\|\cdots\|R_1^d(\alpha))$:

$$\begin{aligned} f(0,x) &= E_1(\alpha, x_1, \alpha, x_2, \ldots, \alpha, x_d)_{l+1} \\ &= E_2(x_1 \oplus R_1^1(\alpha), \alpha, x_2 \oplus R_1^2(\alpha), \alpha, \ldots, x_d \oplus R_1^d(\alpha), \alpha)_l \\ &= f(1, x \oplus (R_1^1(\alpha)\|R_1^2(\alpha)\|\cdots\|R_1^d(\alpha)) \end{aligned} \tag{16}$$

where the second equation results from $x_l^{i+1} = x_{l+1}^i$, $l = 2,4,\ldots,2d$ (Eq. (14)). More precisely:

$$\begin{aligned} &E_1(\alpha, x_1, \alpha, x_2, \ldots, \alpha, x_d)_{l+1} \\ &= E^{i+1}(x)_l \\ &= f_R^{k_1} \circ E_1(\alpha, x_1, \alpha, x_2, \ldots, \alpha, x_d)_l \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_1}(\alpha, x_1, \alpha, x_2, \ldots, \alpha, x_d)_l \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_2}(x_1 \oplus R_1^1(\alpha), \alpha, x_2 \oplus R_1^2(\alpha), \alpha, \ldots, x_d \oplus R_1^d(\alpha), \alpha)_l \\ &= E_2(x_1 \oplus R_1^1(\alpha), \alpha, x_2 \oplus R_1^2(\alpha), \alpha, \ldots, x_d \oplus R_1^d(\alpha), \alpha)_l \end{aligned} \tag{17}$$

where f_R denotes the round function. That is, this function f is a periodic function with $s = 1\|R_1^1(\alpha)\|R_1^2(\alpha)\|\cdots\|R_1^d(\alpha)$. Therefore, according to Theorem 1, the Simon algorithm can retrieve the period of this function using $O(n)$ queries to the target encryption algorithm (Fig. 8). This enables a successful secret-state recovery attack, where n denotes the block size.

Note that we can recover the corresponding round key in our secret-state recovery attacks on Type-1/2 GFS if the round function R is reversible. For instance, if $R_1(x_1^0) = P(x_1^0 \oplus k_1)$, we can obtain the key k_1, where x_1^0 is a random constant.

Related work. Previous Q2-model attacks on Type-1/2 GFS primarily relied on the Simon algorithm. Dong et al. [37] first showed that a quantum distinguishing attack can target a $(2d-1)$-round Type-1 GFS and a $(d+1)$-round Type-2 GFS ($d \geq 3$). Ni et al. [36] then advanced this direction within the quantum chosen-ciphertext (qCCA) model by constructing a quantum distinguisher that distinguishes a (d^2-d+1)-round Type-1 GFS in polynomial time. Sun et al. [10] further increased the attackable rounds for Type-1 GFS

to $(d^2 - 1)$. Overall, these Simon-based quantum attacks offer an exponential advantage in query complexity over classical approaches.

Although the Q2 model alone provides a strong adversarial setting, our work introduces additional related-key assumptions that further enhance the attacker's capabilities, resulting in a "worst-case" security perspective. Under this improved model, the number of rounds vulnerable to effective attacks can be extended arbitrarily, significantly expanding the theoretical boundaries of this research area.

Application to Round-Reduced CAST-256 [44]. CAST-256 is a symmetric block encryption algorithm based on the Type-1 GFS. Although CAST-256 did not become the AES standard, it is still used in specific applications such as secure communication protocols and encryption libraries. Its design and analysis have significant implications for cryptographic research and the development of other encryption algorithms.

The CAST-256 algorithm divides a 128-bit message into four 32-bit branches, supporting key lengths of 128, 192, or 256 bits. The algorithm comprises 48 rounds of encryption: 24 forward and 24 reverse rounds of Type-1 GFS. Each round function uses a 37-bit key. Notably, the proposed attack is general and does not require any additional encryption information of CAST-256. Next, we discuss the secret-state recovery attack against CAST-256.

Based on the 24-round CAST-256 algorithm, we construct the corresponding periodic function in a related-key setting. Given two related-key oracles, $E_k = E^R(x; k_1, k_2, \ldots, k_{24})$ and $E_{k'} = E^R(x; k_2, k_3, \ldots, k_1)$, denoted as E_1 and E_2, we define the following function, with given constants x_1^0, x_3^0 and x_4^0:

$$f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$
$$(b,x) \mapsto \begin{cases} E_1(x_1^0, x, x_3^0, x_4^0)_{l+1} & b = 0 \\ E_2(x, x_3^0, x_4^0, x_1^0)_l & b = 1 \end{cases} \tag{18}$$

where the related-key pair $k = (k_1, k_2, \ldots, k_{24})$ and $k' = (k_2, k_3, \ldots, k_1)$ satisfy the relationship $k'_j = k_{j+1}$ (Fig. 9). The related-key oracle $E_1(\cdot)_{l+1}$ and $E_2(\cdot)_l$ represent the $(l+1)$-th and l-th branches of $E_1(\cdot)$ and $E_2(\cdot)$ respectively, and $l = 2, 3, 4$. In particular, this f satisfies $f(0,x) = f(1, x \oplus R_1(x_1^0))$:

$$\begin{aligned} f(0,x) &= E_1(x_1^0, x, x_3^0, x_4^0)_{l+1} \\ &= E_2(x \oplus R_1(x_1^0), x_3^0, x_4^0, x_1^0)_l \\ &= f(1, x \oplus R_1(x_1^0)) \end{aligned} \tag{19}$$

That is, this function f is a periodic function with $s = 1 \| R_1(x_1^0)$. Therefore, according to Theorem 1, the Simon algorithm can retrieve the period of this function using $O(n)$ queries to the target encryption algorithm. This enables a successful secret-state recovery attack against the 24-round CAST-256 algorithm, extending previous results by 7 rounds (see Table 2).

Based on the 24-round quantum distinguisher of the CAST-256 algorithm, adding $(r - 24)$ rounds before or after the distinguisher enables a key recovery attack on the r

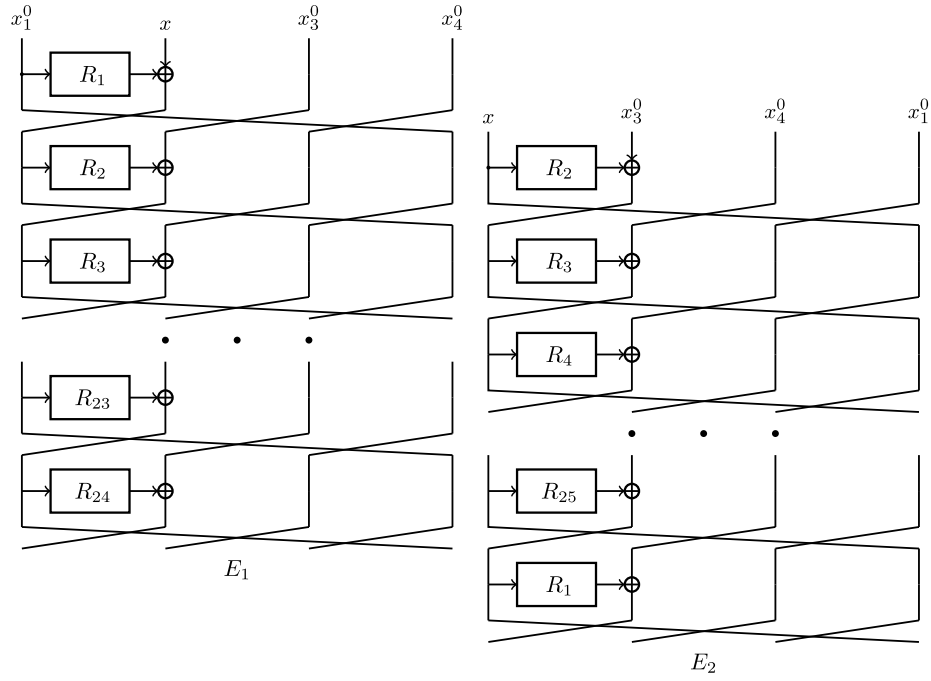


Figure 9 Quantum related-key attack on CAST-256 algorithm

($r > 24$)-round encryption algorithm using the Grover-meets-Simon algorithm. Specifically, the Grover algorithm searches for the round keys of the first or last ($r - 24$) rounds. In contrast, the Simon algorithm verifies if the remaining 24 rounds are periodic, confirming the correctness of the guessed key. Combining the distinguisher with the Grover algorithm, our quantum key-recovery attack offers significant advantages, with complexity $2^{37(r-24)/2}$. For instance, in a 30-round key recovery attack, the query complexity is reduced from 2^{128} using only the Grover algorithm to 2^{111}, given a 37-bit round key length. Also, our attack reduces query complexity by $2^{129.5}$ times compared to existing methods (see Table 2). In particular, in the case of $r \leq 24$, our key-recovery attack degenerates to a distinguishing attack with complexity $O(n)$. Note that for round-reduced CAST-256, the trivial bound using the Grover algorithm is 2^{128}. Based on this, we can attack the 30-round CAST-256 algorithm (256-bit key version) in 2^{111} time, which improves upon the best previous attack of 23 rounds [10].

Note that the attack on CAST-256 assumes shifted round keys, which do not correspond to its actual key schedule. Therefore, the results in Table 2 should be interpreted as theoretical distinguishing attacks in an idealized related-key model rather than as a practical compromise of the deployed CAST-256 cipher.

4 Quantum key-recovery attack for Feistel-FK and misty L-KF constructions

This section introduces new quantum key-recovery attacks on Feistel-FK and Misty L-KF constructions using $O(n)$ superposition queries. These attacks exponentially accelerate the query complexity compared with some previous methods [21, 29, 30]. Table 3 provides a comparison of attack complexities.

Table 3 Quantum key-recovery attack on Feistel-like ciphers

Goal	Construction	Attacked rounds	Condition	Complexity	Source
KR[1]	Feistel-FK	5	-	$O(n)$	[21]
		∞	related key	$O(n)$	Ours
	Misty L-KF	6	-	$O(2^{(3n+2)/4})$	[30]
		∞	related key	$O(n)$	Ours
	Type-3 GFS	r	-	$O(2^{(d-1)(r-d-1)k_j/2})$	[45]
		∞	related key	$O(2^{(d-2)k_j/2})$	Ours

[1] key recovery

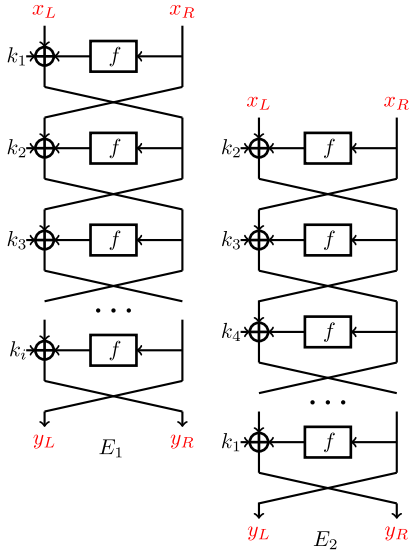


Figure 10 Quantum related-key attack on i-round Feistel-FK construction

4.1 Key recovery attack for Feistel-FK construction

The Feistel cipher structure, or Luby-Rackoff cipher, converts random functions into pseudo-random permutations [46]. Given i round functions $f_i : F_2^n \to F_2^n$ and input (L_0, R_0), the output (L_i, R_i) is produced through the same calculation process. For each round $j = 1, 2, \ldots, i$, the calculation process is as follows

$$\begin{cases} L_j = R_{j-1} \\ R_j = L_{j-1} \oplus f_j(R_{j-1}) \end{cases} \tag{20}$$

Replacing the key-controlled random permutation with a public permutation f and a round key k_j XORed with the output of f yields the Feistel-FK construction, as shown in Fig. 10:

$$\begin{cases} L_j = R_{j-1} \\ R_j = L_{j-1} \oplus f(R_{j-1}) \oplus k_j \end{cases} \tag{21}$$

Based on the i-round Feistel-FK construction, we construct the corresponding periodic function in a related-key setting. Given two related-key oracles, $E_k = E^f(x; k_1, k_2, \ldots, k_i)$ and $E_{k'} = E^f(x; k_2, k_3, \ldots, k_1)$, denoted as E_1 and E_2, respectively, we define the following

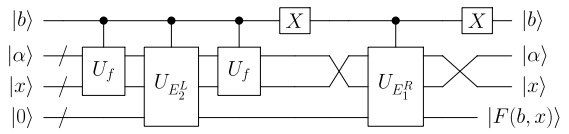


Figure 11 Periodic function for Feistel-FK

function with a given constant α:

$$F : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$
$$(b,x) \mapsto \begin{cases} E_1(x,\alpha)_R & b = 0 \\ E_2(\alpha, x \oplus f(\alpha))_L & b = 1 \end{cases} \tag{22}$$

The related-key pair $k = (k_1, k_2, \ldots, k_i)$ and $k' = (k_2, k_3, \ldots, k_1)$ satisfy the relationship $k'_j = k_{j+1}$ (Fig. 10). The related-key oracles $E_1(\cdot)_R$ and $E_2(\cdot)_L$ represent the right and left branches of $E_1(\cdot)$ and $E_2(\cdot)$, respectively. Moreover, this F satisfies $F(0,x) = F(1, x \oplus k_1)$:

$$\begin{aligned} F(0,x) &= E_1(x,\alpha)_R \\ &= E_2(\alpha, x \oplus f(\alpha) \oplus k_1)_L \\ &= F(1, x \oplus k_1) \end{aligned} \tag{23}$$

where the second equation results from $L_{j+1} = R_j$ (Eq. (21)). More precisely:

$$\begin{aligned} E_1(x,\alpha)_R &= E^{i+1}(x)_L \\ &= f_R^{k_1} \circ E_1(x,\alpha)_L \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_1}(x,\alpha)_l \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_2}(\alpha, x \oplus f(\alpha) \oplus k_1)_L \\ &= E_2(\alpha, x \oplus f(\alpha) \oplus k_1)_L \end{aligned} \tag{24}$$

where f_R denotes the round function. The function F is periodic with $s = 1 \| k_1$. According to Theorem 1, the Simon algorithm can retrieve the period using $O(n)$ queries to the target encryption algorithm (Fig. 11). This allows a successful key-recovery attack, where n denotes the block size.

The partial key recovery attack on the Feistel-FK construction can be extended to a complete key recovery attack. Specifically, given i oracles $E_1, E_2, \ldots, E_i$ (similar in form to above E_1 and E_2), we use a random constant α to define the following function:

$$F_j : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$
$$(b,x) \mapsto \begin{cases} E_j(x,\alpha)_R & b = 0 \\ E_{j+1}(\alpha, x \oplus f(\alpha))_L & b = 1 \end{cases} \tag{25}$$

where $1 \le j \le i - 1$. From the above, we see that executing the Simon algorithm on the function F_j allows key k_i to be recovered after $O(n)$ superposition queries. Thus, the full key $k_1, k_1, \ldots, k_i$ can be recovered.

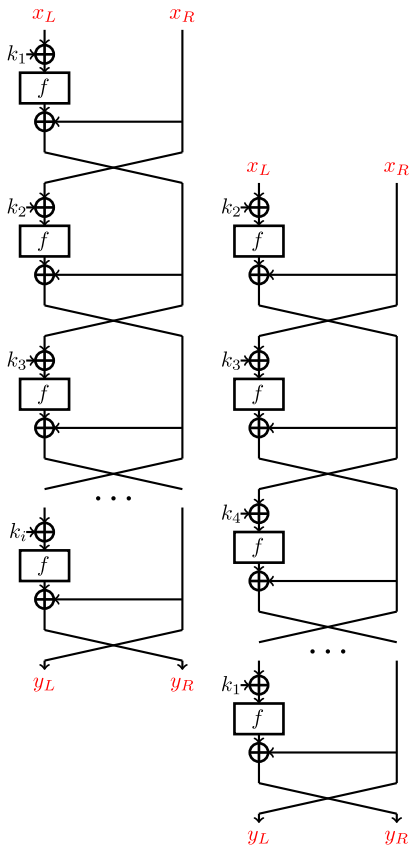


Figure 12 Quantum related-key attack on *i*-round Misty L-KF construction

4.2 Key recovery attack for misty L-KF construction

MISTY (Mitsubishi Improved Security Technology) [47] is a symmetric block encryption algorithm that Matsui designed based on the Feistel network. The MISTY L-F construction is a left version of MISTY, where the round function is denoted as

$$\begin{cases} L_i = R_{i-1} \\ R_i = f_i(L_{i-1}) \oplus k_i \end{cases} \tag{26}$$

Replacing the key-controlled random permutation with a public permutation f and a round key k_i XORed with the input of f yields the Misty L-KF construction, as shown in Fig. 12:

$$\begin{cases} L_i = R_{i-1} \\ R_i = R_{i-1} \oplus f(L_{i-1} \oplus k_i) \end{cases} \tag{27}$$

For the i-round Misty L-KF construction, we give the periodic function in a related-key setting. Using related-key oracles $E_k = E^f(x; k_1, k_2, \ldots, k_i)$ and $E_{k'} = E^f(x; k_2, k_3, \ldots, k_1)$, denoted as E_1 and E_2, we define the function with a constant α:

$$\begin{aligned} &F : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n \\ &\qquad (b,x) \mapsto \begin{cases} E_1(x,\alpha)_R & b = 0 \\ E_2(\alpha, \alpha \oplus f(x))_L & b = 1 \end{cases} \end{aligned} \tag{28}$$

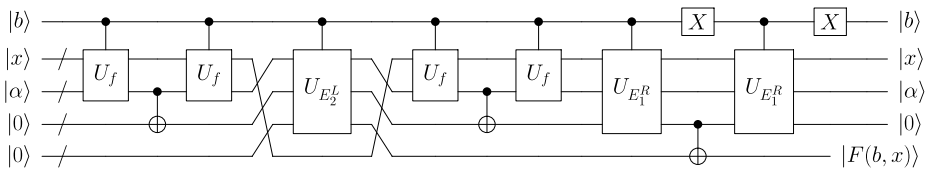


Figure 13 Periodic function for Misty L-KF

The related-key pair $k = (k_1, k_2, \ldots, k_i)$ and $k' = (k_2, k_3, \ldots, k_1)$ satisfy $k'_j = k_{j+1}$ (Fig. 12). The related-key oracles $E_1(\cdot)_R$ and $E_2(\cdot)_L$ represent the right and left branches of $E_1(\cdot)$ and $E_2(\cdot)$. Additionally, F satisfies $F(0,x) = F(1, x \oplus k_1)$:

$$\begin{aligned} F(0,x) &= E_1(x,\alpha)_R \\ &= E_2(\alpha, f(x \oplus k_1) \oplus \alpha)_L \\ &= F(1, x \oplus k_1) \end{aligned} \tag{29}$$

where the second equation results from $L_{i+1} = R_i$ (Eq. (27)). More precisely:

$$\begin{aligned} E_1(x,\alpha)_R &= E^{i+1}(x,\alpha)_L \\ &= f_R^{k_1} \circ E_1(x,\alpha)_L \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_1}(x,\alpha)_L \\ &= f_R^{k_1} \circ f_R^{k_i} \circ \cdots \circ f_R^{k_2}(\alpha, f(x \oplus k_1) \oplus \alpha)_L \\ &= E_2(\alpha, f(x \oplus k_1) \oplus \alpha)_L \end{aligned} \tag{30}$$

where f_R denotes the round function. The function F has a period defined by $s = 1 \| k_1$. Using Theorem 1, the Simon algorithm can determine this period with $O(n)$ queries to the target encryption algorithm (Fig. 13). This facilitates a successful key-recovery attack, where n denotes the block size. Specifically, given i oracles $E_1, E_2, \ldots, E_i$, the full key $k_1, k_1, \ldots, k_i$ can be recovered.

5 Conclusion

In this paper, we give secret state recovery and key recovery attacks for a series of Feistel-like constructions in the quantum-related-key model, requiring only $O(n)$ quantum queries. Unlike the Type-1 and Type-2 generalized Feistel structures (GFS), the Type-3 GFS offers higher security and complexity through more intricate cross-mixing and permutation operations. This makes it better suited for encryption scenarios with higher security requirements. Consequently, we cannot construct a period function, making the above secret-state recovery attack unsuitable for Type-3 GFS. However, we introduce a hidden period function, allowing the period to be recovered using the Grover-meets-Simon algorithm, resulting in a new key recovery attack with complexity $2^{(d-2)k_l/2}$, where d represents the number of branches, and k_l represents the length of the round key. We describe this in Appendix. A further question is whether Type-3 GFS has a polynomial-time secret state recovery attack, which we leave as an open problem.

As a final remark, our attack relies on an idealized related-key assumption rather than the practical key schedules used in real-world scenarios. Therefore, it does not threaten practical Feistel-like schemes. This raises two open questions: (1) how to adapt such attacks to specific schemes like CAST-256 that use actual key schedules, and (2) whether Feistel-like structures still exhibit exploitable related-key periodicity under more realistic related-key differentials, such as those generated by real key scheduling algorithms. Addressing these questions is vital for bridging the gap between theoretical analysis and practical cryptanalysis.

Appendix: Secret-state recovery attack for Type-3 GFS

The Type-3 GFS enhances data diffusion and obfuscation effects during encryption by introducing cross-mixing and permutation operations between its branches. The algorithm divides the input state of dn bits into $d \geq 3$ branches, each containing a n-bit sub-block. In each round $j = 1, 2, \ldots, i$, the dn-bit input undergoes encryption using $(d-1)$ round functions controlled by keys, implemented as follows:

$$x_1^j \leftarrow R_i^1(x_1^{j-1}) \oplus x_2^{j-1}, x_2^j \leftarrow R_j^2(x_2^{j-1}) \oplus x_3^{j-1}, \ldots, x_d^j \leftarrow x_1^{j-1} \tag{A.1}$$

We construct a hidden periodic function in a related-key setting to distinguish the i-round Type-3 GFS construction from random permutations. We define this function using two related-key oracles, $E_k = E^R(x; k_1, k_2, \ldots, k_i)$ and $E_{k'} = E^R(x; k_2, k_3, \ldots, k_1)$, denoted as E_1 and E_2, where $k_j = (k_j^1, k_j^2, \ldots, k_j^{d-1})$, $1 \leq j \leq i$. The function is specified with given constants $x_1^0, x_2^0, \ldots, x_{d-1}^0$:

$$\begin{aligned} f : \{0,1\}^{(d-2)n} \times \{0,1\} \times \{0,1\}^n &\rightarrow \{0,1\}^n \\ (k_u, b, x) &\mapsto \begin{cases} E_1(x_1^0, x_2^0, \ldots, x_{d-1}^0, x)_1 & b = 0 \\ E_2(x_2^0 \oplus R_1^{u_1}(x_1^0), x_3^0 \oplus R_1^{u_2}(x_2^0), \ldots, x)_d & b = 1 \end{cases} \end{aligned} \tag{A.2}$$

where the related-key pair $k = (k_1, k_2, \ldots, k_i)$ and $k' = (k_2, k_3, \ldots, k_1)$ satisfy the relationship $k'_j = k_{j+1}$ ($k_j \in \{0,1\}^{k_l}$) and $k_u = (k_u^1, k_u^1, \ldots, k_u^{d-2})$. The related-key oracle $E_1(\cdot)_1$ and $E_2(\cdot)_d$ represent the 1-th and d-th branches of $E_1(\cdot)$ and $E_2(\cdot)$, respectively. In particular, this f satisfies $f(k_1, 0, x) = f(k_1, 1, x \oplus R_1^{d-1}(x_{d-1}^0))$:

$$\begin{aligned} f(k_1, 0, x) &= E_1(x_1^0, x_2^0, \ldots, x_{d-1}^0, x)_1 \\ &= E_2(x_2^0 \oplus R_1^1(x_1^0), x_3^0 \oplus R_1^2(x_2^0), \ldots, x \oplus R_1^{d-1}(x_{d-1}^0))_d \\ &= f(k_1, 1, x \oplus R_1^{d-1}(x_{d-1}^0)) \end{aligned} \tag{A.3}$$

That is, this function f is a hidden periodic function with period $s = 1 \| R_1^{d-1}(x_{d-1}^0)$. The attacker first guesses $k_u \in \{0,1\}^{(d-2)k_l}$ (the Grover part). Only if the guess is correct, i.e., $k_u = k_1$, does the attacker obtain a periodic function, which can then be detected with the Simon algorithm. Consequently, the Grover-meets-Simon algorithm recovers the period k_1 with $O(2^{(d-2)k_l/2})$ queries to the Type-3 GFS, enabling a successful key-recovery attack.

Abbreviations

PRF, Pseudo-Random Function; PRP, Pseudo-Random Permutation; HNDL, Harvest Now, Decrypt Later; GFS, Generalized Feistel Schemes.

Author contributions
Hong-Wei Sun and Rong-Xue Xu wrote the main manuscript text, Ke-Jia Zhang, Long Zhang, Dan-Dan Li and Zhen-Qiang Li prepared all the algorithms. All authors reviewed the manuscript.

Funding information
This work is supported by the National Natural Science Foundation of China (Grant Nos. 62471070 and 62271234), the Fundamental Research Funds for Heilongjiang Universities under Grant No. 2024-KYYWF-0137, Open Foundation of State Key Laboratory of Public Big Data (Guizhou University) under Grant No. PBD2022-16, Double First-Class Project for Collaborative Innovation Achievements in Disciplines Construction in Heilongjiang Province under Grant Nos. LJGXCG2022-054 and LJGXCG2023-028, Open Foundation of State key Laboratory of Networking and Switching Technology (Beijing University of Posts and Telecommunications) (SKLNST-2024-1-04).

Data Availability
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
The Author confirms: that the work described has not been published before; that it is not under consideration for publication elsewhere; that its publication has been approved by all co-authors.

Competing interests
The authors declare no competing interests.

Author details
[1]School of Computer and Big Data (School of Cybersecurity), Heilongjiang University, Harbin 150080, China. [2]School of Mathematical Science, Heilongjiang University, Harbin 150080, China. [3]School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing 100876, China. [4]State Key Laboratory of Cryptology, Beijing 100878, China.

Received: 8 September 2025 Accepted: 11 February 2026 Published online: 17 February 2026

References

1. Shor PW. Algorithms for quantum computation: discrete logarithms and factoring. In: 35th annual symposium on foundations of computer science. IEEE Computer Society; 1994. p. 124–34.
2. Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Commun ACM. 1978;21(2):120–6.
3. Koblitz N. Elliptic curve cryptosystems. Math Comput. 1987;48(177):203–9.
4. Grover LK. A fast quantum mechanical algorithm for database search. In: Miller GL, editor. Proceedings of the twenty-eighth annual ACM symposium on the theory of computing, Philadelphia, Pennsylvania, USA, May 22–24, 1996. New York: ACM; 1996. p. 212–9.
5. Simon DR. On the power of quantum computation. SIAM J Comput. 1997;26(5):1474–83.
6. Bernstein E, Vazirani UV. Quantum complexity theory. SIAM J Comput. 1997;26(5):1411–73.
7. Leander G, May A. Grover meets Simon - quantumly attacking the FX-construction. In: Advances in cryptology - ASIACRYPT. 2017. p. 161–78.
8. Sun HW, Wei CY, Cai BB, et al. Improved BV-based quantum attack on block ciphers. Quantum Inf Process. 2023;22:9. https://doi.org/10.1007/s11128-022-03752-x.
9. Sun HW, Cai BB, Qin SJ, et al. Quantum attacks on beyond-birthday-bound MACs. Phys A, Stat Mech Appl. 2023;625:129047.
10. Sun HW, Cai BB, Qin SJ, et al. Quantum attacks on type-1 generalized Feistel schemes. Adv Quantum Technol. 2023;6(10):2300155.
11. Li Z, Cai B, Sun H, et al. Novel quantum circuit implementation of advanced encryption standard with low costs. Sci China, Phys Mech Astron. 2022;65:290311.
12. Kuwakado H, Morii M. Quantum distinguisher between the 3-round Feistel cipher and the random permutation. In: 2010 IEEE international symposium on information theory proceedings (ISIT), June 2010. 2010. p. 2682–5.
13. Kuwakado H, Morii M. Security on the quantum-type even-mansour cipher. In: ISITA. IEEE; 2012. p. 312–6.
14. Kaplan M, Leurent G, Leverrier A, et al. Breaking symmetric cryptosystems using quantum period finding. In: CRYPTO 2016, Part II. 2016. p. 207–37.
15. Zhandry M. How to construct quantum random functions. In: 2012 IEEE 53rd annual symposium on foundations of computer science. IEEE; 2012. p. 679–87.
16. Gagliardoni T. Quantum security of cryptographic primitives. 2017. arXiv preprint arXiv:1705.02417.
17. Hosoyamada A, Sasaki Y. Cryptanalysis against symmetric-key schemes with online classical queries and offline quantum computations. In: Cryptographers' track at the RSA conference. Cham: Springer; 2018. p. 198–218.
18. Kaplan M, Leurent G, Leverrier A, et al. Quantum differential and linear cryptanalysis. IACR Trans Symmetric Cryptol. 2016;2016(1):71–94.
19. Hosoyamada A, Sasaki Y. Quantum Demiric-Selçuk meet-in-the-middle attacks: applications to 6-round generic Feistel constructions. In: Security and cryptography for networks: 11th International Conference, SCN 2018, Amalfi, Italy, September 5–7, 2018, Proceedings 11. Berlin: Springer; 2018. p. 386–403.

20. Bonnetain X, Hosoyamada A, Naya-Plasencia M, et al. Quantum attacks without superposition queries: the offline Simon's algorithm. In: International conference on the theory and application of cryptology and information security. Cham: Springer; 2019. p. 552–83.
21. Canale F, Leander G, Stennes L. Simon's algorithm and symmetric crypto: generalizations and automatized applications In: Annual international cryptology conference. Cham: Springer Nature Switzerland; 2022. p. 779–808.
22. Bonnetain X, Schrottenloher A. Single-query quantum hidden shift attacks. Cryptology ePrint archive. 2023.
23. Sun HW, Gao F, Xu RX, et al. Quantum Key-Recovery Attacks on Permutation-Based Pseudorandom Functions. IEEE Internet Things J. 2025.
24. Bellare M, Kohno T. A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In: Biham E, editor. Advances in cryptology - EUROCRYPT 2003. EUROCRYPT 2003. Lecture notes in computer science. vol. 2656. Berlin: Springer; 2003.
25. Albrecht MR, Farshim P, Paterson KG, Watson GJ. On cipher-dependent related-key attacks in the ideal-cipher model. In: Joux A, editor. Fast software encryption. FSE 2011. Lecture notes in computer science. vol. 6733. Berlin: Springer; 2011.
26. Roetteler M, Steinwandt R. A note on quantum related-key attacks. Inf Process Lett. 2015;115(1):40–4.
27. Hosoyamada A, Aoki K. On quantum related-key attacks on iterated Even-Mansour ciphers. IEICE Trans Fundam Electron Commun Comput Sci. 2019;102(1):27–34.
28. Xie H, Yang L. A quantum related-key attack based on the Bernstein-Vazirani algorithm. Quantum Inf Process. 2020;19(8):240.
29. Ito G, Hosoyamada A, Matsumoto R, et al. Quantum chosen-ciphertext attacks against Feistel ciphers. In: Topics in cryptology-CT-RSA 2019: the cryptographers' track at the RSA conference 2019, San Francisco, CA, USA, March 4–8, 2019, Proceedings. Berlin: Springer; 2019. p. 391–411.
30. Zou J, Zou HK, Dong XY, Wu WL, Luo YY. New Key Recovery Attack Based on Periodic Property. J. Softw. (In Chinese).
31. Zheng Y, Matsumoto T, Imai H. On the construction of block ciphers provably secure and not relying on any unproved hypotheses. In: Brassard G, editor. Advances in cryptology - CRYPTO' 89 proceedings, CRYPTO 1989. Lecture notes in computer science. vol. 435. New York: Springer; 1990.
32. Carlisle A, Gilchrist J. The CAST-256 Encryption Algorithm. RFC 2612. June; 1999.
33. Shirai T, Shibutani K, Akishita T, Moriai S, Iwata T. The 128-bit blockcipher CLEFIA (extended abstract). In: Biryukov A, editor. Fast software encryption. FSE 2007. Lecture notes in computer science. vol. 4593. Berlin: Springer; 2007. https://doi.org/10.1007/978-3-540-74619-5_12.
34. Rivest RL, Robshaw MJB, Sidney R, et al. The RC6TM block cipher. First advanced encryption standard (AES) conference. 1998;16.
35. Wu H, Preneel B. AEGIS: a fast authenticated encryption algorithm. In: Lange T, Lauter K, Lisoněk P, editors. Selected areas in cryptography - SAC 2013. SAC 2013. Lecture notes in computer science. vol. 8282. Berlin: Springer; 2014. https://doi.org/10.1007/978-3-662-43414-7_10.
36. Ni B, Ito G, Dong X, et al. Quantum attacks against type-1 generalized Feistel ciphers and applications to CAST-256. In: International conference on cryptology in India. Cham: Springer; 2019. p. 433–55.
37. Dong X, Li Z, Wang X. Quantum cryptanalysis on some generalized Feistel schemes. Sci China Inf Sci. 2019;62(2):22501.
38. Biham E. New types of cryptanalytic attacks using related keys. J Cryptol. 1994;7:229–46.
39. Knudsen LR. Cryptanalysis of LOKI 91. International Workshop on the Theory and Application of Cryptographic Techniques. 1992: 196–208.
40. Dunkelman O, Keller N, Shamir A. A practical-time related-key attack on the KASUMI cryptosystem used in GSM and 3G telephony. In: Advances in cryptology-CRYPTO 2010: 30th annual cryptology conference, Santa Barbara, CA, USA, August 15–19, 2010. Proceedings 30. 2010. p. 393–410.
41. Ferguson N, Kelsey J, Lucks S, et al. Improved cryptanalysis of Rijndael. In: Fast software encryption: 7th international workshop, FSE 2000 New York, NY, USA, April 10–12, 2000. Proceedings 7. 2001. p. 213–30.
42. Mantin I. A practical attack on the fixed RC4 in the WEP mode. In: Advances in cryptology-ASIACRYPT 2005: 11th international conference on the theory and application of cryptology and information security, Chennai, India, December 4–8, 2005. Proceedings 11. 2005. p. 395–411.
43. Daemen J, Rijmen V. Probability distributions of correlation and differentials in block ciphers. J Math Cryptol. 2007;1(3):221–42.
44. Adams C, Gilchrist J. The CAST-256 encryption algorithm[R]. 1999.
45. Zhang Z, Wu W, Sui H, et al. Quantum attacks on type-3 generalized Feistel scheme and unbalanced Feistel scheme with expanding functions. Chin J Electron. 2023;32(2):209–16.
46. Feistel H, Notz WA, Smith JL. Some cryptographic techniques for machine-to-machine data communications. Proc IEEE. 1975;63(11):1545–54.
47. Matsui M. New structure of block ciphers with provable security against differential and linear cryptanalysis. In: International workshop on fast software encryption. Berlin: Springer; 1996. p. 205–18.

Publisher's note