Conditional Constant Function Problem and Its Quantum Solutions: Attacking Feistel Ciphers

Zhen-Qiang Li[1], Shu-Qin Fan[1⋆], Fei Gao[2], Yong-Lin Hao[1], Xi-Chao Hu[1], Lin-Chun Wan[3], Hong-Wei Sun[4], and Qi Su[1]

[1] State Key Laboratory of Cryptology, Beijing 100878, China, lizhenqiang92@163.com, fansq@sklc.org

[2] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China,

[3] School of Computer and Information Science, Southwest University, Chongqing 400715, China

[4] School of Computer and Big Data (School of Cybersecurity), Heilongjiang University, Harbin 150080, China

Abstract. In this paper, we define the conditional constant function problem (CCFP) and, for a special case of CCFP, we propose a quantum algorithm for solving it efficiently. Such an algorithm enables us to make new evaluations to the quantum security of Feistel block cipher in the case where a quantum adversary only has the ability to make online queries in a classical manner, which is relatively realistic. Specifically, we significantly improved the chosen-plaintext key recovery attacks on two Feistel block cipher variants known as Feistel-KF and Feistel-FK. For Feistel-KF, we construct a 3-round distinguisher based on the special case of CCFP and propose key recovery attacks mounting to $r > 3$ rounds. For Feistel-FK, our CCFP based distinguisher covers 4 rounds and the key recovery attacks are applicable for $r > 4$ rounds. Utilizing our CCFP solving algorithm, we are able to reduce the classical memory complexity of our key recovery attacks from the previous exponential $O(2^{cn})$ to $O(1)$. The query complexity of our key recovery attacks on Feistel-KF is also significantly reduced from $O(2^{cn})$ to $O(1)$ where c's are constants. Our key recovery results enjoy the current optimal complexities. They also indicate that quantum algorithms solving CCFP could be more promising than those solving the period finding problem.

Keywords: Feistel cipher · chosen-plaintext · classical queries · Grover's algorithm · constant function

1 Introduction

Compared with classical computing, quantum computing exhibits significant speed-up in handling certain problems, such as linear systems [10,35,20], dimensionality reduction [23,27,40], and so on [32,19,41,21]. In particular, it seriously threatens the security of classical cryptographic schemes. In the field of asymmetric cryptography, Shor's algorithm [30] can solve factorization and discrete logarithms in polynomial time, which will completely break the currently used public-key systems, such as RSA and ECC. In the field of symmetric cryptography, Grover's algorithm [9] offers a quadratic speedup on an exhaustive search attack. This gives rise to the common assertion that symmetric-key schemes only retain roughly half of their classical bits of security. When aiming for post-quantum security, doubling the key size is necessary.

Nevertheless, exhaustive search for key recovery is just one of many potential attack methods. The post-quantum security of symmetric schemes requires more nuanced consideration. The report on quantum computing by the National Academy of Sciences [28] indicates that some currently unknown smart quantum attacks could be more efficient than Grover's algorithm. For instance, by utilizing Simon's algorithm [31], the Even-Mansour cipher [17] and three-round Feistel cipher [16] were shown to be broke in polynomial time.

Quantum attacks can be primarily classified into two models [44], namely Q1 model and Q2 model, under different abilities of the attacker. In the Q1 model, attackers can perform any offline quantum computation, while they are only permitted to make classical queries to an encryption oracle. In the Q2 model, other than offline quantum computations, attackers can make quantum superposition queries to an encryption oracle. Currently, some polynomial-time Q2 attacks (e.g. Even-Mansour cipher [17]) have been proposed, while the assumption of attacks in the Q2 model (i.e., requiring the ability of quantum superposition queries) is rather strong such that the practical impact of these attacks remains blurry. This motivates us to investigate the Q1 attack, since it is relatively realistic.

Feistel block ciphers are one of the most important and extensively researched cryptographic schemes. The n-bit internal state of a Feistel block cipher is updated by iterative calls of an F function which takes half of the internal state and a $n/2$-bit round-key as inputs and produces a new $n/2$-bit state for updating another half of the internal state. Under the assumption of round-key independence, the key length of r-round Feistel block ciphers is $nr/2$ bits. Based on the different definitions of F, the Feistel block cipher can be further partitioned into 3 categories, namely Feistel-F, Feistel-KF and Feistel-FK. The security of the Feistel block ciphers against quantum attacks has been studied in various settings for both Q1 and Q2 models.

As a generic exhaustive search, Grover's algorithm [9] can recover the $nr/2$-bit key of Feistel block ciphers with $O(2^{nr/4})$ times, $O(1)$ classical queries and $O(n)$ qubits. When $O(n2^p)$ qubits are available, the parallel Grover's algorithm [8] can further reduce the time complexity to $O(2^{nr/4-p/2})$. There are also dedicated quantum attacks utilizing the structural features of Feistel block ciphers.

For the Q2 model, Kuwakado and Morii [16] proposed a quantum distinguisher on 3-round Feistel-F constructions. Their idea is to construct a periodic function based on the construction of Feistel-F which can then be distinguished from the random function, using Simon's algorithm [31]: the quantum period finding algorithm with polynomial complexities. Such an idea was employed by

Leander and May for designing the Grover-meet-Simon algorithm [18], which is the basis of almost all quantum attacks on Feistel block ciphers afterwards. Based on the 3-round distinguisher, in 2018, Dong et al. [7] recovered the $nr/2$-bit key of r-round ($r > 3$) Feistel-F block ciphers with time $O(n^3 2^{(r-3)n/4})$ and $O(n2^{(r-3)n/4})$ quantum queries. Based on Zhandry's quantum collision algorithm [43], in 2023, Chartouny et al. [5] put forward quantum distinguishing attacks on the 5-round Feistel-F construction, with a time complexity and query complexity of $O(2^{n/3})$. In 2024, Chartouny et al. [4] showed a quantum distinguishing attack on the 6-round Feistel-F construction with $O(2^{4n/5})$ time and quantum queries, by utilizing the subset finding quantum algorithm developed by Childs and Eisenberg [6]. For Feistel-KF, Hosoyamada and Sasaki [11] proposed a 3-round quantum distinguisher and recovered the $nr/2$-bit key with $O(n^2 2^{(r-3)n/4})$ quantum queries and $O(n^3 2^{(r-3)n/4})$ time. As to Feistel-FK, Ito et al. [12] found a 5-round quantum distinguisher, based on which they put forward a key recovery attack on r-round ($r > 5$) block ciphers with time $O(n^3 2^{(r-5)n/4})$ and $O(n2^{(r-5)n/4})$ quantum queries.

As to the Q1 model, Hosoyamada and Sasaki [11] presented a quantum algorithm based on the parallelized Grover search [8] to solve a variant of claw finding problem, which reduced the the number of classical queries for attacking 6-round Feistel-KF constructions from $O(2^{3n/4})$ (in classical attacks) to $O(2^{n/2})$. In 2023, Xu et al. [36] further generated new key-recovery attacks on r-round ($r \geq 7$) Feistel-KF constructions, requiring $O(2^{2n/3+(r-7)n/4})$ time and $O(2^{2n/3})$ classical queries. Liu et al. [22] proposed a quantum algorithm to solve the claw problem of finding multiple equations (a variant of claw finding problem) based on quantum walk [45], reducing the time complexity and the classical memory complexity required for attacking 6-round Feistel-FK constructions from $O(2^{n/2})$ of the classical attack [37] to $O(2^{n/3})$. In 2024, Yu et al. [39] demonstrated that Dong et al.'s Q2 attack [7] on Feistel-F constructions can be transformed into the Q1 attack by increasing the amount of classical memory.

Contributions. In this paper, we generalize the key search problem in [18] as a formally defined *conditional constant function problem* and give a quantum algorithm for solving a special case of it. The algorithm is based on the parallel Grover's algorithm [8] but with fewer queries, enabling us to improve the key recovery attack on Feistel-KF and Feistel-FK block ciphers in the quantum chosen-plaintext (CPA) setting under the Q1 model. We show the complexities of our attacks along with those of previous results in Table 1. For Feistel-KF, we give a 3-round distinguisher requiring $O(1)$ classical queries instead of $O(n)$ quantum ones. Combining the distinguisher with our new algorithm, we propose a new Q1 attack on r-round ($r > 3$) Feistel-KF. In comparison with previous results, our method reduces the number of queries from exponential to constant. Note that such a reduction is applicable for both Q1 and Q2 models. [5] For Feistel-FK, we give a 4-round distinguisher with $O(1)$ classical queries. Based on

[5] As any Q1 attack can be readily transformed to a Q2 attack by regarding quantum oracles as classical oracles, we are able to construct an attack in the Q2 model that has the same complexity as our Q1 attack.

such a distinguisher, we present key recovery attacks on r-round ($r > 4$) Feistel-FK. As can be seen from Table 1, our attacks on both Feistel-KF and Feistel-FK have significant lower memory complexities and require fewer queries.

Table 1. Summary of key recovery attacks on Feistel block ciphers in the CPA setting. D: number of queries; T: time complexity; Q: number of qubits; M: amount of classical memory. r: number of rounds.

Structures	Setting	Rounds	D	T	Q	M	DTQM
Feistel-F	Q2[7]	> 3	$2^{(r-3)n/4}$	$2^{(r-3)n/4}$	n^2	1	$n^2 2^{(r-3)n/2}$
	Q1[39]	> 3	$2^{n/2}$	$2^{(r-3)n/4}$	n^2	$2^{n/2}$	$n^2 2^{(r+1)n/4}$
Feistel-KF	Classical[11]	6	$2^{3n/4}$	$2^{3n/4}$	/	$2^{n/2}$	$\geq 2^{2n}$
	Q2[11]	> 3	$2^{(r-3)n/4}$	$2^{(r-3)n/4}$	n^2	1	$n^2 2^{(r-3)n/2}$
	Q1[11]	6	$2^{n/2}$	$2^n/Q$	$Q \leq 2^{n/2}$	$2^{n/2}$	2^{2n}
	Q1[36]	> 6	$2^{2n/3}$	$2^{2n/3+(r-7)n/4}$	$2^{5n/6}$	$2^{5n/6}$	$2^{(5+r)n/4}$
	Q1(Sect. 4)	> 3	$\mathbf{1}$	$\mathbf{2^{(r-3)n/4}/\sqrt{Q}}$	$\mathbf{Q \leq 2^{(r-3)n/2}}$	$\mathbf{1}$	$\mathbf{2^{(r-3)n/4}\sqrt{Q}}$
	Q1(Sect. 4)	6	$\mathbf{1}$	$\mathbf{2^{3n/4}/\sqrt{Q}}$	$\mathbf{Q \leq 2^{3n/2}}$	$\mathbf{1}$	$\mathbf{2^{3n/4}\sqrt{Q}}$
Feistel-FK	Classical[37]	6	1	$2^{n/2}$	/	$2^{n/2}$	$\geq 2^n$
	Q2[12]	> 5	$2^{(r-5)n/4}$	$2^{(r-5)n/4}$	n^2	1	$n^2 2^{(r-5)n/2}$
	Q1[22]	6	1	$2^{n/3}$	$2^{n/3}$	$2^{n/3}$	2^n
	Q1(Sect. 5)	> 4	$\mathbf{1}$	$\mathbf{2^{(r-4)n/4}/\sqrt{Q}}$	$\mathbf{Q \leq 2^{(r-4)n/2}}$	$\mathbf{1}$	$\mathbf{2^{(r-4)n/4}\sqrt{Q}}$
	Q1(Sect. 5)	6	$\mathbf{1}$	$\mathbf{2^{n/2}/\sqrt{Q}}$	$\mathbf{Q \leq 2^n}$	$\mathbf{1}$	$\mathbf{2^{n/2}\sqrt{Q}}$

2 Preliminaries

2.1 Notation

For a positive integer n, let $\{0,1\}^n$ denote the set of all n-bit strings. Let $Perm(n)$ denote the set of all permutations on $\{0,1\}^n$, and let $Func(n)$ denote the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$. For bit strings a and b, $a\|b$ indicates their concatenation. For given vectors a and b with same dimension, their inner product is denoted $a \cdot b$. Let "$\oplus$" denote the XOR.

2.2 Pseudo-random Permutation

This paper considers that the attacker makes chosen-plaintext attacks. That is, the attacker queries with plaintexts and then gets corresponding ciphertexts. The attacker analyzes these plaintext-ciphertext pairs to recover the correct key. PRP-CPA and qPRP-CPA denote the pseudo-random permutation (PRP) security and the quantum pseudo-random permutation (PRP) security under chosen-plaintext attacks. The formal definition is given as follows.

Definition 1. *(PRP-CPA/qPRP-CPA [24]) Let $\mathcal{E}_k : K \times X \to X$ be a family of permutations indexed by the elements in K, and $g : X \to X$ be a permutation* *in $Perm(X)$. Let $\mathcal{A}$ be an attacker. The PRP-CPA/qPRP-CPA advantage of $\mathcal{A}$ is defined as:*

$$Adv_{\mathcal{E}}^{PRP-CPA/qPRP-CPA}(\mathcal{A}) = |Pr_{k\in K}(\mathcal{A}^{\mathcal{E}_k(*)} \Rightarrow 1) - Pr_{g\in PermX}(\mathcal{A}^{g(*)} \Rightarrow 1)|, \tag{1}$$

where $*$ is replaced by $\cdot$ (classical) or $\odot$ (quantum).

$\mathcal{A}^{f(\cdot)} \Rightarrow 1$ (resp. $\mathcal{A}^{f(\odot)} \Rightarrow 1$) represents an algorithm that makes classical queries (resp. quantum queries) to oracle $\mathcal{E}_k$ and outputs 1. When $1 - Adv(\mathcal{A})$ is a sufficiently small value, the Definition 1 ensures that $\mathcal{E}_k$ can be distinguished from a random permutation.

2.3 Basics of Quantum Computation

We suppose that the reader knows some basics of quantum computation, such as the definitions of qubit, quantum gates (including Hadamard gate H, Toffoli gate, CNOT gate, multi-control CNOT gate, and so on), quantum states and the ket notation $|\cdot\rangle$. For a more extensive presentation, please refer to [26].

All quantum computations are unitary operators of the Hilbert space and are reversible. Generally, any classical computation can be implemented as a quantum circuit as long as one employs a sufficient number of ancilla qubits. A quantum circuit represents that a sequence of universal quantum gates (e.g., CNOT gate, NOT gate and Toffoli gate) are applied to a set of qubits. The invertibility of quantum gates guarantees that the quantum circuit is reversible. The initial state of ancilla qubits is $|0\rangle$ and they are recovered to $|0\rangle$ after performing quantum computations. Certain quantum computations are performed on the input state $|x\rangle$. Thereafter, an uncomputation process is implemented by executing the same operations in reverse to restore the initial state of the ancilla qubits. The uncomputation of a unitary operation U corresponds to using its adjoint (i.e., conjugate transpose) operator $U^\dagger$.

In cryptanalysis, effective access to an oracle is essential. A single value will be obtained if a classical oracle (for example, a cipher with an unknown key) are queried once. A quantum oracle for a function f can be represented as a unitary operator $O_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$, which can be queried in superposition. That is, $O_f \sum_x |x\rangle|y\rangle = \sum_x |x\rangle|y \oplus f(x)\rangle$.

2.4 Grover's Algorithm

Grover's algorithm [9], or the Grover search, can find a marked element from an unstructured data set with a quadratic speedup, compared with the classical algorithm, as stated in Algorithm 1. In particular, it solves the following Grover's problem.

Problem 1. **(Grover's problem)** Let $f : \{0,1\}^n \to \{0,1\}$ is a Boolean function such that there exist $x_0 \in \{0,1\}^n$ such that $f(x_0) = 1$ $(|\{x_0 \in \{0,1\}^n | f(x_0) = 1\}| = M)$. Find a x_0.

Algorithm 1: Grover's algorithm [9]

```
Input: Oracle $O_f : |x\rangle|y\rangle \to |x\rangle|y \oplus f(x)\rangle$, where $f : \{0,1\}^n \to \{0,1\}$ is a
       Boolean function and $y \in \{0,1\}$
Output: $x_0$
Initialize a quantum state: $|0\rangle^{\otimes n}|1\rangle$.
Apply $n+1$-fold Hadamard gate $H^{\otimes n+1}$ to attain
    $$H^{\otimes n+1}|0\rangle^{\otimes n}|1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle|-\rangle = |\varphi\rangle|-\rangle.$$
Perform Grover iteration operation $R \approx \frac{\pi}{4}\sqrt{2^n/M}$ times to give the goal state:
    $$[((2|\varphi\rangle\langle\varphi| - I) \otimes I_2)O_f]^R|\varphi\rangle|-\rangle.$$
Measure to return a $x_0$ with a high probability.
```

In Algorithm 1, $|-\rangle = (|0\rangle - |1\rangle)/\sqrt{2}$ and O_f is applied to mark the solution indices due to $O_f|x\rangle|-\rangle = (-1)^{f(x)}|x\rangle|-\rangle$, where $x \in \{0,1\}^n$. Assume the time required to evaluate f once is $O(1)$. then Grover's algorithm can solve Problem 1 in time $O(2^{n/2}/\sqrt{M})$, using $O(n)$ qubits.

When $O(n2^p)$ qubits are available, the Grover's algorithm can be parallelized. Kim et al. [15] summarized two methods to parallelize Grover's algorithm as inner and outer parallelization. Zalka [42] concluded that these two parallelization versions require $O(2^{(n-p)/2}/\sqrt{M})$ time and $O(n2^p)$ qubits. Jaques et al [13] pointed out that inner parallelization is more applicable to the scenario of key search in terms of success probability and input size. In particular, inner parallelization divides the entire search space into 2^p disjoint subsets, and assigns each subset to a parallel machine. Since the size of search space per machine is 2^{n-p}, the time solving Problem 1 can be reduced to $O(2^{(n-p)/2}/\sqrt{M})$. This paper chooses inner parallelization as well.

Moreover, Grover's algorithm has been generalized into the quantum amplitude amplification (QAA) technique [3], as described in Theorem 1.

Theorem 1. *(QAA [3]) Suppose $\mathcal{A}$ is any quantum algorithm on q qubits that does not perform measurement. Let $\mathcal{B} : \{0,1\}^q \to \{0,1\}$ be a function that categorizes the outcomes of $\mathcal{A}$ as either good state or bad state. Let $p > 0$ be the initial success probability that the measurement of $\mathcal{A}|0\rangle$ is good. Set $t = \lceil \frac{\pi}{4\theta} \rceil$, where θ is defined using $sin^2\theta = p$ and $0 < \theta < \frac{\pi}{2}$. Besides, define the unitary operator $Q = \mathcal{A}\mathcal{S}_0\mathcal{A}^{-1}\mathcal{S}_{\mathcal{B}}$, where the unitary operator $\mathcal{S}_{\mathcal{B}}$ changes the sign of the good state, i.e., $\mathcal{S}_{\mathcal{B}}|x\rangle = (-1)^{\mathcal{B}}|x\rangle$, while $\mathcal{S}_0 = 2|0\rangle\langle 0| - I$ changes the sign of the amplitude exclusively when it isn't the zero state $|0\rangle$. Eventually, after carrying out the computation of $Q^t\mathcal{A}|0\rangle$, the measurement yields a good state with probability at least $max\{1-p, p\}$.*

2.5 The Feistel Structure and Its Variants

For a r-round Feistel block cipher, we let n be its block size and denote $x \in \{0,1\}^n$ as its internal state: such an internal state can be divided into two $n/2$-bit halves as $x = (x_L, x_R)$. The input of the Feistel block cipher consists of the n-bit initial value of the internal state, denoted as x^0, and r $n/2$-bit independent round keys $k_1, \ldots, k_r \in \{0,1\}^{n/2}$. Starting from x^0, the Feistel block cipher computes the new state x^i from x^{i-1} $(i = 1, \ldots, r)$ as follows:

$$\begin{cases} x_R^i = x_L^{i-1} \\ x_L^i = F_i(x_L^{i-1}, k_i) \oplus x_R^{i-1} \end{cases} \tag{2}$$

where F_i is the function mapping elements in $\{0,1\}^{n/2} \times \{0,1\}^{n/2}$ to $\{0,1\}^{n/2}$. The round function in Eq. (2) is referred as the Feistel-F construction [12] and is illustrated as Fig. 1.(a). The F_i in Eq. (2) can be further simplified from a key dependent function to a public permutation over $\{0,1\}^{n/2}$ while the round keys are injected through the simplest XOR operations. Such a simplicity results in two Feistel variants known as Feistel-KF and Feistel-FK whose round functions are defined as Eq. (3) and Eq. (4), illustrated in Fig. 1.(b) and Fig. 1.(c) respectively.

$$\begin{cases} x_R^i = x_L^{i-1} \\ x_L^i = F_i(x_L^{i-1} \oplus k_i) \oplus x_R^{i-1} \end{cases} \tag{3}$$

$$\begin{cases} x_R^i = x_L^{i-1} \\ x_L^i = F_i(x_L^{i-1}) \oplus k_i \oplus x_R^{i-1} \end{cases} \tag{4}$$

Both Feistel-KF and Feistel-FK are applied in the design of standard block ciphers. Typical Feistel-KF block ciphers are DES [33] and Camellia [1] while lightweight primitives such as Piccolo [29] and Simeck [38] adopt Feistel-FK.

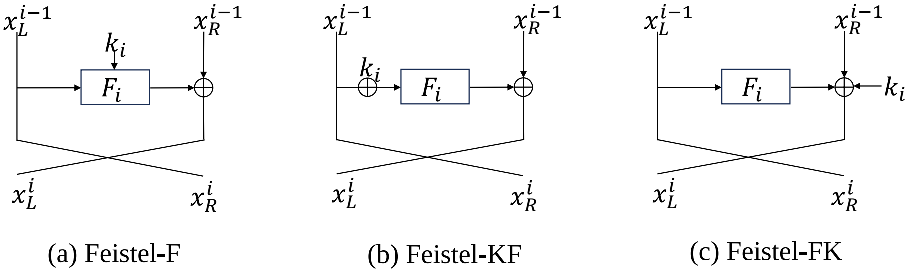

Fig. 1. Three Feistel constructions.

3 The Conditional Constant Function Problem and Its Quantum Solutions

We give the formal definition of the conditional constant function problem as Problem 2, which is referred to as CCFP hereafter for short.

Problem 2. **(Conditional constant function problem, CCFP)** Let $F:\{0,1\}^k\times\{0,1\}^n\rightarrow\{0,1\}^n$ be a function satisfying that there is a unique secret value $i_0\in\{0,1\}^k$ such that $F(i_0,x)\equiv C$ holds for all $x\in\{0,1\}^n$, where $C\in\{0,1\}^n$ is an unknown constant. Find i_0.

3.1 Previous Algorithm

Leander and May [18] considered the special case of CCFP where $F(i,x)$ is defined as Eq. (5)

$$F(i,x)=f(i,x)\oplus g(x), \tag{5}$$

where $f(i,x)$ is predefined as $f:\{0,1\}^k\times\{0,1\}^n\rightarrow\{0,1\}^n$ ($f(i,x)$ is a random permutation in $Perm(n)$ for any given $i\in\{0,1\}^k$) and $g(x)=f(i_0,x)\oplus C$. They solved such a Eq. (5) defined CCFP with the Grover search. We briefly review their method here as the context of our new quantum algorithm.

First, a test is defined to check whether $f(i,x)\oplus g(x)$ is constant for deciding if $i=i_0$. Specifically, make oracle queries to $g(x)$ with $\ell+1$ random $x_j\in\{0,1\}^n$ and acquire the corresponding $y_j=g(x_j)$'s ($j=1,\ldots\ell+1$). Then the classical test function $\mathcal{B}:\{0,1\}^k\rightarrow\{0,1\}$ is defined as

$$\mathcal{B}(i)=\begin{cases}1 & \text{all } \ell \text{ identities } y_j\oplus f(i,x_j)=y_{j+1}\oplus f(i,x_{j+1})\ (j=1,\ldots,\ell) \text{ hold},\\ 0 & \text{otherwise.}\end{cases} \tag{6}$$

If $i=i_0$, $y_j\oplus f(i,x_j)$ is equal to the unknown constant C, and $\mathcal{B}(i)=1$ in Eq. (6) holds true with a probability of 1. Otherwise, if $i\neq i_0$, based on the randomness of $f(i,x)$, any of the ℓ identities is fulfilled with probability 2^{-n} and $\mathcal{B}(i)=1$ can only happen with a probability of $2^{-n\ell}$. Therefore, the probability for an incorrect i to pass the $\mathcal{B}(i)=1$ test is at most $(2^k-1)2^{-n\ell}$. Obviously, $\ell=\lceil\frac{2k}{n}\rceil$ is enough to ensure $(2^k-1)2^{-n\ell}<2^{-k}$. Altogether, the test $\mathcal{B}$ defines a unitary operator $S_{\mathcal{B}}:|i\rangle\rightarrow(-1)^{\mathcal{B}}|i\rangle$, which makes $O(k)$ queries to $g(x)$ and $f(i,x)$ (due to $k>\ell$).

Then $\mathcal{A}$ from Theorem 1 is defined as the k-fold Hadamard gate $H^{\otimes k}$. And the uniform superposition of all $i\in\{0,1\}^k$ can be obtained by performing $\mathcal{A}$ to $|0\rangle^{\otimes k}$. After repeatedly performing $O(2^{k/2})$ times the unitary operations $Q=\mathcal{A}(2|0\rangle\langle 0|-I)\mathcal{A}^{-1}S_{\mathcal{B}}$ to the state $\mathcal{A}|0\rangle^{\otimes k}$, i_0 can be returned with probability at least $1-2^{-k}$. As a result, when $F(i,x)=g(x)\oplus f(i,x)$, i_0 is determined with $O(2^{k/2}\cdot k)$ queries to $f(i,x)$ and $g(x)$.

3.2 Our New Algorithm for Solving the Conditional Constant Function Problem

We consider another special case of CCFP. For predefined keyed permutation $f:\{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ (for arbitrary $i \in \{0,1\}^k$, $f(i,x)$ is a permutation over $\{0,1\}^n$ so there is f^{-1} s..t $f^{-1}(i, f(i,x)) \equiv x$ holds for all $x \in \{0,1\}^n$) and function $g:\{0,1\}^n \rightarrow \{0,1\}^n$, $F(i,x)$ is defined as

$$F(i,x) = f(i, g(x)) \tag{7}$$

We also give a quantum algorithm for solving the Eq. (7) defined CCFP by finding the unique $i_0 \in \{0,1\}^k$ s.t. $F(i_0,x) = f(i_0, g(x)) \equiv C$. We have two key observations about the solving process in Sect. 3.1, summarized as follows:

Observation 1 *In Sect. 3.1, when performing the Grover search over $i \in \{0,1\}^k$, a new function $f(i,x)$ is required whenever a new i is tested. However, $g(x)$ is always the same. Because of remaining unchanged, we would like to reduce the number of queries to $g(x)$. This can bring some benefits when making query to $g(x)$ is more expensive than those to $f(i,x)$.*

Observation 2 *For each $i \in \{0,1\}^k$, once we have a quantum state $|\psi_g\rangle = \bigotimes_{j=1}^{\ell+1} |x_j\rangle|g(x_j)\rangle$, we can obtain the quantum state $|\psi_F\rangle = \bigotimes_{j=1}^{\ell+1} |x_j\rangle|f(i,g(x_j))\rangle$ by making quantum oracle access to $f(i,x)$. By applying some Toffoli gates, we can judge if $F(i,x)$ is a constant function (see Remark 1). After that, by performing uncomputations on $|\psi_F\rangle$, $|\psi_g\rangle$ can be recovered and reused in subsequent iterations.*

In the following, we propose a quantum algorithm to solve Eq. (7) defined CCFP as is formally described in Algorithm 2. This is a generalization and improvement of the previous algorithm proposed by Leander and May [18]. Our idea is to separate the query to $g(x)$ and $f(i,x)$, and iteratively reuses $|\psi_g\rangle$. In particular, the first phase is to prepare the quantum state $|\psi_g\rangle$ by making $\ell+1$ queries to $g(x)$. The second phase is to perform the Grover search over $i \in \{0,1\}^k$. For each fixed i, we check if $f(i,g(x))$ is a constant function by utilizing the quantum state $|\psi_g\rangle$ and making queries to $f(i,x)$. After that, uncomputations are performed to recover the quantum state $|\psi_g\rangle$. Detailed explanations of Algorithm 2 are given in Sect. 3.3.

3.3 Detailed Analysis of Algorithm 2

In Algorithm 2, since the quantum state $|\psi_g\rangle$ is non-superposition, it can be prepared by making $\ell+1$ classical queries to $g(x)$. To be specific, for each x_j, we make queries to $g(x_j)$ in a classical manner, and then implement the map of $(|x_j\rangle|0\rangle, g(x_j)) \mapsto |x_j\rangle|g(x_j)\rangle$. Subsequently, $|\psi_g\rangle$ can be acquired by simultaneously performing $\ell+1$ such mappings.

Algorithm 2: Quantum algorithm solving Eq. (7) defined CCFP

```
Input: Oracle $O_f : |i\rangle|x\rangle \to |i\rangle|f(i,x)\rangle$ and classical query access to $g(x)$
Output: $i_0$
Start in the all-zero state.
For $l+1$ random $x_j \in \{0,1\}^n$, make $\ell+1$ queries to $g(x)$ to obtain the quantum state
    $$|\psi_g\rangle = \bigotimes_{j=1}^{\ell+1} |x_j\rangle|g(x_j)\rangle.$$
Apply $H^{\otimes k}$ to attain an uniform superposition over $i \in \{0,1\}^k$
    $$\frac{1}{\sqrt{2^k}} \sum_{i\in\{0,1\}^k} |i\rangle \otimes |\psi_g\rangle.$$
Prepare the quantum state $|b\rangle = |-\rangle$ by performing Hadamard gate $H$ to $|1\rangle$.
Apply $\ell+1$ times oracle $O_f$ to build the quantum state
    $$\frac{1}{\sqrt{2^k}} \sum_{i\in\{0,1\}^k} |i\rangle \bigotimes_{j=1}^{\ell+1} |x_j\rangle|f(i,g(x_j))\rangle|b\rangle = \frac{1}{\sqrt{2^k}} \sum_{i\in\{0,1\}^k} |i\rangle|\psi_F\rangle|b\rangle.$$
In order to run the $\mathcal{B}(i)$ test, we add a new register $|r\rangle$: set $r := 1$ when $f(i,g(x_j)) = f(i,g(x_{j+1}))$ holds for all $j = 1, \cdots, \ell$; otherwise, set $r = 0$. (Remark 1 shows the quantum implementation of $\mathcal{B}(i) = r$)
Add $r$ to $b$, then uncompute $r$ to obtain
    $$\frac{1}{\sqrt{2^k}} \sum_{i\in\{0,1\}^k} |i\rangle|\psi_F\rangle|b \oplus r\rangle.$$
Apply $\ell+1$ times oracle $O_f^{-1}$ (the inverse of $O_f$) to revert the quantum state
    $$\frac{1}{\sqrt{2^k}} \sum_{i\in\{0,1\}^k} |i\rangle \otimes |\psi_g\rangle|b \oplus r\rangle.$$
Perform $O(2^{k/2})$ Grover iteration, then measure to return $i_0$ with a high probability.
```

The test function $\mathcal{B} : \{0,1\}^k \times \{0,1\}^{(\ell+1)n} \times \{0,1\}^{(\ell+1)n} \to \{0,1\}$ is defined in the similar way as Eq. (6) as , which maps $(i, x_1, \cdots, x_{\ell+1}, g(x_1), \cdots, g(x_{\ell+1}))$ to 1 iff

$$\mathcal{B}(i) = \begin{cases} 1 & \text{all } \ell \text{ identities } f(i, g(x_j)) = f(i, g(x_{j+1}))\ (j = 1, \ldots, \ell) \text{ hold,} \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

In Remark 1, we show a way to determine whether all equations in Eq. (8) are satisfied. Similar to the analysis in Eq. (6), the probability that there exists an incorrect i passing the $\mathcal{B}(i) = 1$ test is at most $(2^k - 1)2^{-n\ell}$. $\ell = \lceil \frac{2k}{n} \rceil$ is enough to ensure $(2^k - 1)2^{-n\ell} < 2^{-k}$. Altogether, the test function $\mathcal{B}$ defines a unitary operator $S_{\mathcal{B}} : |i\rangle|\psi_g\rangle|-\rangle \to (-1)^{\mathcal{B}}|i\rangle|\psi_g\rangle|-\rangle$ corresponding to step 4-7. $\mathcal{S}_{\mathcal{B}}$ is accomplished without any new query to $g(x)$. Instead, only $4(\ell + 1)$ quantum queries to $f(i, x)$ (including f^{-1}) are made (assume that making one query to f requires the same complexity as that of f^{-1}). Thus, $\mathcal{S}_{\mathcal{B}}$ requires $O(\ell) = O(k)$ queries to $f(i, x)$ (due to $k > \ell$).

Remark 1. In step 5, we determine if all identities $f(i, g(x_j)) = f(i, g(x_{j+1}))$ $(j = 1, \cdots, \ell)$ are satisfied by solely using some $(\ell + 1)$-control CNOT gates. Taking the example of checking whether $(a_0, b_0) = (a_1, b_1)$ and $(a_1, b_1) = (a_2, b_2)$ $((a_0, b_0), (a_1, b_1)$ and $(a_2, b_2) \in \{0,1\}^2)$ are satisfied. If $a_0 = a_1 = a_2$, then the identity $a_0 \cdot a_1 \cdot a_2 \oplus (1 \oplus a_0) \cdot (1 \oplus a_1) \cdot (1 \oplus a_2) = 1$ holds true; otherwise, this identity does not hold. The check for b_0, b_1 and b_2 is similar. We depict its quantum implementation in Fig. 2, which requires eight 3-control CNOT gates, one Toffoli gate (i.e., 2-control CNOT gate) and three auxiliary qubits. Recursively, in step 5, the checking of whether all ℓ identities (each $f(i, g(x_j)) \in \{0,1\}^n$) are satisfied requires $4n$ $\ell + 1$-control CNOT gates, one n-control CNOT gate and $n + 1$ auxiliary qubits. What's more, since an ℓ-control CNOT gate can be constructed by employing $8\ell - 24$ Toffoli gates without additional qubits [2], step 5 requires approximately $4n(8\ell - 16) + 8n - 24$ Toffoli gates and $n + 1$ auxiliary qubits [6]. As a consequence, we claim that the time complexity of realizing step 5 is $O(nk)$.

Remark 2. For each given i, $f(i, x)$ is a permutation over $\{0,1\}^n$, and its inverse can be computed. Hence, the oracle $O_f : |i\rangle|x\rangle \to |i\rangle|f(i, x)\rangle$ can be implemented in two steps utilizing an additional register:

$$|i\rangle|x\rangle|0\rangle \mapsto |i\rangle|x\rangle|f(i, x)\rangle \mapsto |i\rangle|x \oplus f^{-1}(f(i, x))\rangle|f(i, x)\rangle = |i\rangle|0\rangle|f(i, x)\rangle. \tag{9}$$

The Grover iteration is performed $O(2^{k/2})$ times in step 8 of Algorithm 2. As a result, we obtain the following Theorem 2.

[6] Toffoli gate realizes the function of $|a\rangle|b\rangle|c\rangle \to |a\rangle|b\rangle|c \oplus a \cdot b\rangle$. CNOT gate, NOT gate and Toffoli gate are universal, i.e., any quantum computation can be implemented by using these gates. But the CNOT and NOT gates are significantly less expensive than the Toffoli gate. Thus, in general, we can say that the number of Toffoli gates or Toffoli depth determines the running time of an algorithm.

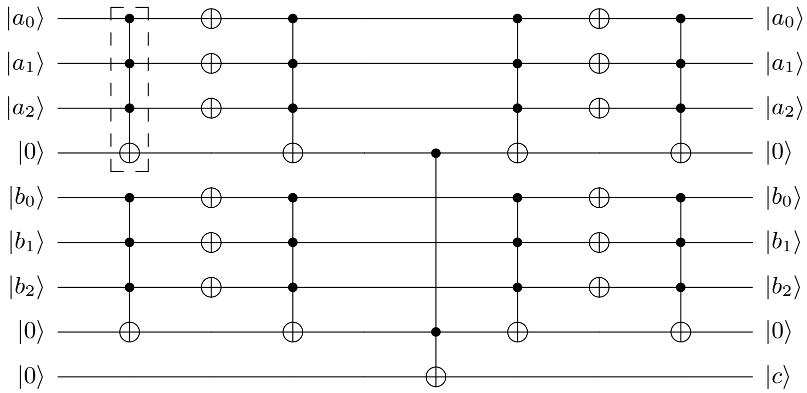

Fig. 2. The quantum implementation for checking whether two identities $(a_0, b_0) = (a_1, b_1)$ and $(a_1, b_1) = (a_2, b_2)$ are satisfied. If $c = 1$ holds, they are satisfied; Otherwise, they are not. The 3-control CNOT gate in the dashed box realizes the function of $|a\rangle|b\rangle|c\rangle|d\rangle \rightarrow |a\rangle|b\rangle|c\rangle|d \oplus a \cdot b \cdot c\rangle$. And "$\oplus$" realizes the function of $|a\rangle \rightarrow |a \oplus 1\rangle$.

Theorem 2. *Algorithm 2 can solve Eq. (7) defined Problem 2 with probability at least $1 - \frac{1}{2^k}$ by performing $O(k)$ classical queries to $g(x)$ and $O(2^{k/2} \cdot k)$ quantum queries to $f(i, x)$. The offline computation (the procedures excluding those for preparing the state $|\psi_g\rangle$) is done in time $O((T_f k + nk)2^{k/2})$, where T_f is the time required to evaluate $f(i, x)$ once. Meanwhile, $O(kn)$ qubits and $O(k)$ classical memory are required.*

Corollary 1. *Actually, Algorithm 2 is a variant of Grover's algorithm. When $O(kn2^p)$ qubits are available, Algorithm 2 can be parallelized, and then i_0 can be determined with $O((T_f k + nk)2^{(k-p)/2})$ time. They are balanced at $p = k/3$. That is, by applying the parallelized Algorithm 2, we can solve Eq. (7) defined Problem 2 in time $O((T_f k + nk)2^{k/3})$ using $O(kn2^{k/3})$ qubits.*

3.4 Summary

To solve Eq. (7) defined CCFP, we separate the query to $g(x)$ from that to $f(i, x)$, and reuse the quantum state $|\psi_g\rangle$. Algorithm 2 will be utilized to achieve an improvement from certain attacks on symmetric schemes by reducing the query complexity from exponential to constant. Indeed, in the context where Grover's algorithm is used, it may be possible to perform the queries to the cryptographic oracle only once to decrease the number of queries to $O(1)$. Besides, Algorithm 2 can be parallelized to decrease the time required for finding i_0.

4 Quantum Attacks on Feistel-KF Constructions

This section introduces a quantum distinguishing attack against Feistel-KF constructions. Based on this distinguisher, we present quantum key recovery attacks by combining Algorithm 2. For the sake of simplicity, we presume that a single evaluation of each primitive, for example a block cipher, can be accomplished in time $O(1)$.

4.1 Quantum Distinguishing Attacks

We propose a quantum distinguisher on the 3-round Feistel-KF cipher in the chosen-plaintext attack (CPA) setting. Fig. 3 illustrates the 3-round Feistel-KF construction. $F_1, F_2, F_3 \in Perm(n/2)$ are public round functions, while k_1, k_2, k_3 are independently chosen subkeys. Let $(x_L^0, x_R^0) \in \{0,1\}^n$ be a plaintext of the 3-round Feistel-KF cipher, the corresponding ciphertext (x_L^3, x_R^3) should be computed as

$$\begin{aligned} x_L^3 &= x_R^0 \oplus F_1(k_1 \oplus x_L^0) \oplus F_3(k_3 \oplus x_L^0 \oplus F_2(k_2 \oplus x_R^0 \oplus F_1(k_1 \oplus x_L^0))), \\ x_R^3 &= x_L^0 \oplus F_2(k_2 \oplus x_R^0 \oplus F_1(k_1 \oplus x_L^0)). \end{aligned} \tag{10}$$

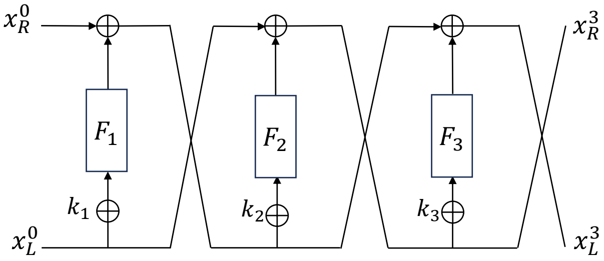

Fig. 3. The 3-round Feistel-KF construction.

We consider the following Problem 3 originated in [16].

Problem 3. ([16]) Let $\mathcal{O} : \{0,1\}^n \mapsto \{0,1\}^n$ be either the 3-round Feistel-KF construction or a random permutation. The goal is to make a distinction between the two cases by making queries to $\mathcal{O}$ while queries to the inverse mapping $\mathcal{O}^{-1}$ of $\mathcal{O}$ are not allowed.

Let $\beta \in \{0,1\}^{n/2}$ be an arbitrary constant. For β and $x \in \{0,1\}^{n/2}$, we take (β, x) as the plaintext. When $\mathcal{O}$ is the 3-round Feistel-KF construction, the right branch of the ciphertext is described as $x_R^3 = \beta \oplus F_2(k_2 \oplus x \oplus F_1(k_1 \oplus \beta))$. Then we can see that $F_2^{-1}(x_R^3 \oplus \beta) \oplus x = k_2 \oplus F_1(k_1 \oplus \beta)$ holds. Even if x is changed, $k_2 \oplus F_1(k_1 \oplus \beta)$ remains unchanged. Thus, we can define a function $f^{\mathcal{O}}$ as

$$\begin{aligned} f^{\mathcal{O}} : \{0,1\}^{n/2} &\to \{0,1\}^{n/2} \\ x &\mapsto F_2^{-1}(d \oplus \beta) \oplus x, \end{aligned} \tag{11}$$

where $\mathcal{O}(\beta, x) = (c, d)$. If $\mathcal{O}$ is the 3-round Feistel-KF construction, the output pair (c, d) from $\mathcal{O}$ represents the ciphertext (x_L^3, x_R^3). Then $f^{\mathcal{O}}$ is described as

$$\begin{aligned} f^{\mathcal{O}}(x) &= F_2^{-1}(\beta \oplus F_2(k_2 \oplus x \oplus F_1(k_1 \oplus \beta)) \oplus \beta) \oplus x \\ &= k_2 \oplus x \oplus F_1(k_1 \oplus \beta) \oplus x \\ &= k_2 \oplus F_1(k_1 \oplus \beta). \end{aligned} \tag{12}$$

For $m + 1$ random $x_i \in \{0,1\}^{n/2}$, we compute their function values. Based on Eq. (12), if $\mathcal{O}$ is the 3-round Feistel-KF construction, then all identities

$$f^{\mathcal{O}}(x_i) = f^{\mathcal{O}}(x_{i+1}) \quad \text{for all} \quad 1 \le i \le m \tag{13}$$

are fulfilled with probability 1. If $\mathcal{O}$ is a random permutation, any identity is fulfilled with probability $2^{-n/2}$, and then all m identities are simultaneously fulfilled with probability at most $2^{-nm/2}$. Obviously, a constant of $m \ge 2$ guarantees that $2^{-nm/2} \le 2^{-n}$. Thus, the following lemma holds.

Lemma 1. *If $\mathcal{O}$ is the 3-round Feistel-KF construction, the function $f^{\mathcal{O}}$ satisfies $f^{\mathcal{O}}(x) = k_2 \oplus F_1(k_1 \oplus \beta)$ for any $x \in \{0,1\}^{n/2}$. That is, $f^{\mathcal{O}}$ is a constant function. In contrary, if all m identities in Eq. (13) are simultaneously fulfilled, $f^{\mathcal{O}}$ is the 3-round Feistel-KF construction with a high probability.*

Note that since we use the inverse F_2^{-1} of round function F_2, it is necessary to assume that F_2 is bijection. Although it is widely known that this is not a requirement for Feistel-KF constructions, we nonetheless hold the view that our attack is meaningful. Based on these, we show a distinguisher against the 3-round Feistel-KF construction with $O(1)$ classical queries.

First, we choose $x_1, x_2, x_3 \in \{0,1\}^{n/2}$ randomly and just prepare their direct product state $|x_1\rangle|x_2\rangle|x_3\rangle$. And then we only make 3 classical queries to $\mathcal{O}$ to get their function values defined in Eq. (11), i.e.,

$$|x_1\rangle|c_1\rangle|f^{\mathcal{O}}(x_1)\rangle|x_2\rangle|c_2\rangle|f^{\mathcal{O}}(x_1)\rangle|x_3\rangle|c_3\rangle|f^{\mathcal{O}}(x_1)\rangle, \tag{14}$$

where $\mathcal{O}(\beta, x_j) = (c_j, d_j), j = 1, 2, 3$ and $\beta \in \{0,1\}^{n/2}$ be an arbitrary constants. At last, we check whether $f^{\mathcal{O}}(x_1) = f^{\mathcal{O}}(x_2)$ and $f^{\mathcal{O}}(x_2) = f^{\mathcal{O}}(x_3)$ holds by using the method mentioned in Remark 1. If the result returns 1, then output "$\mathcal{O}$ is the 3-round Feistel-KF construction." If the result returns 0, then output "$\mathcal{O}$ is a random permutation."

Let $\mathcal{A}$ be designated as an attacker, and denote the 3-round Feistel-KF construction as 3FKF. According to Definition 1, the qPRP-CPA advantage of $\mathcal{A}$ is $Adv_{3FKF}^{qPRP-CPA}(\mathcal{A}) = 1 - 2^{-n}$, which guarantees that we can distinguish the 3-round Feistel-KF construction from a random permutation.

Corollary 2. *The 3-round Feistel-KF construction and a random permutation can be correctly distinguished in $O(1)$ classical queries and $O(n)$ time. Compared with the previous distinguishing attack [11] which requires $O(n)$ quantum queries and $O(n^3)$ time, our attack demands fewer queries and less time.*

Remark 3. The round function F_2 is a public permutation and is assumed to be invertible. Therefore, the circuit mapping $|y\rangle \to |F_2^{-1}(y)\rangle$ can be implemented in two steps utilizing an additional register:

$$|y\rangle|0\rangle \mapsto |y\rangle|F_2^{-1}(y)\rangle \mapsto |y \oplus F_2(F_2^{-1}(y))\rangle|F_2^{-1}(y)\rangle = |0\rangle|F_2^{-1}(y)\rangle. \tag{15}$$

Remark 4. Similar to the previous work [16], we also use the right branch of the output of $\mathcal{O}$. However, instead of using quantum superposition states, we just employ a quantum direct product state, and perform some Toffoli gates to judge whether $f^{\mathcal{O}}$ is a constant function, which doesn't destroy the entanglement between the left and right branch of the output. Thus, we don't need to truncate the output of $\mathcal{O}$ for building the function $f^{\mathcal{O}}(x)$.

4.2 Quantum Key Recovery Attacks

Below we demonstrate that Algorithm 2 can be extended to recover the keys of the 6-round Feistel-KF construction by making use of $O(1)$ classical queries in the quantum CPA setting.

Given the encryption oracle of the 6-round Feistel-KF construction Enc_6 : $\{0,1\}^n \mapsto \{0,1\}^n$, we consider to guess subkeys of the last 3-round, i.e., k_4, k_5, k_6. As depicted in Fig. 4, if the guess is right, then a quantum circuit that implements the first three rounds will be obtained. Otherwise, a quantum circuit that evaluates an almost random function will be gained. Hence, we can check whether the guess is correct by utilizing our 3-round quantum distinguisher. In particular, k_4, k_5, k_6 are guessed by using the Grover search, while a conditional constant function is constructed to judge the correctness of the guess.

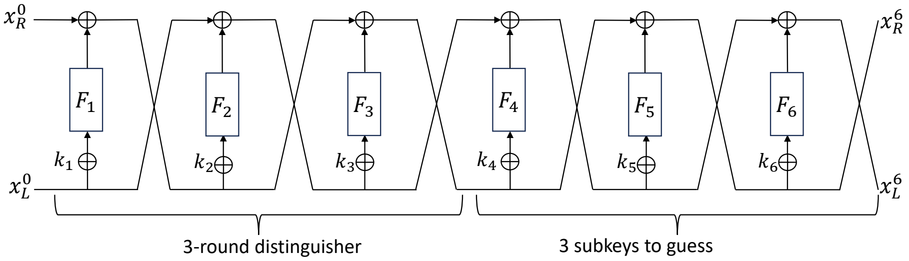

Fig. 4. Key recovery attack on 6-round Feistel-KF constructions.

Next, we provide detailed explanations of our attacks. Let $\beta \in \{0,1\}^{n/2}$ be an arbitrary constant and $Enc_6(\beta, x) = (x_L^6, x_R^6)$, we define the following function:

$$\begin{aligned} G : \{0,1\}^{3n/2} \times \{0,1\}^{n/2} &\to \{0,1\}^{n/2} \\ (k_4, k_5, k_6, x) &\mapsto F_2^{-1}(F_4(F_5(x_L^6 \oplus F_6(x_R^6 \oplus k_6) \\ &\quad \oplus k_5) \oplus x_R^6 \oplus k_4) \oplus x_L^6 \oplus F_6(x_R^6 \oplus k_6) \\ &\quad \oplus \beta) \oplus x. \end{aligned} \tag{16}$$

Under the accurate guess of the key (k_4, k_5, k_6), $G(k_4, k_5, k_6, x) = F_2^{-1}(x_3^R \oplus \beta) \oplus x = F_1(\beta \oplus k_1) \oplus k_2$ is a constant function. Thus, we can applied Algorithm 2 to recover the correct (k_4, k_5, k_6).

First, we make $\ell + 1$ classical queries to Enc_6 to prepare the quantum state $|\psi_g\rangle = \bigotimes_{j=1}^{\ell+1} |x_j\rangle |Enc_6(\beta, x_j)\rangle = \bigotimes_{j=1}^{\ell+1} |x_j\rangle |x_L^6\rangle |x_R^6\rangle$. Then, we run the Grover search over $(k_4, k_5, k_6) \in \{0,1\}^{3n/2}$. For each fixed (k_4, k_5, k_6), we can implement in place:

$$\bigotimes_{j=1}^{\ell+1} |x_j\rangle |x_L^6\rangle |x_R^6\rangle \mapsto \bigotimes_{j=1}^{\ell+1} |x_j\rangle |F_5(x_L^6 \oplus F_6(x_R^6 \oplus k_6) \oplus k_5) \oplus x_R^6\rangle |G(k_4, k_5, k_6, x_j)\rangle, \tag{17}$$

which, when (k_4, k_5, k_6) is correct, is exactly:

$$\bigotimes_{j=1}^{\ell+1} |x_j\rangle |F_5(x_L^6 \oplus F_6(x_R^6 \oplus k_6) \oplus k_5) \oplus x_R^6\rangle |F_1(\beta \oplus k_1) \oplus k_2\rangle. \tag{18}$$

From there, we can check if $G(k_4, k_5, k_6, x)$ is a constant function by using our 3-round quantum distinguisher. After that, by performing uncomputations, we can recover the quantum state $|\psi_g\rangle$, and reuse it in subsequent iterations. After performing $O(2^{3n/4})$ Grover iterations, the correct subkeys k_4, k_5, k_6 can be recovered.

If (k_4, k_5, k_6) is correct, all identities $G(k_4, k_5, k_6, x_j) = G(k_4, k_5, k_6, x_{j+1})$ $(j = 1, 2, \cdots, \ell)$ are fulfilled with probability 1. If (k_4, k_5, k_6) is incorrect, any identity $G(k_4, k_5, k_6, x_j) = G(k_4, k_5, k_6, x_{j+1})$ is fulfilled with probability $2^{-n/2}$. Obvious, when $\ell = 6$, an incorrect (k_4, k_5, k_6) is returned with a probability at most $(2^{3n/2} - 1)2^{-\ell n/2} \leq 2^{-3n/2}$. As a result, according to Theorem 2, the correct (k_4, k_5, k_6) will be returned with probability at least $1 - \frac{1}{2^{3n/2}}$ by making $O(1)$ classical queries to Enc_6 and doing the offline quantum computation in time $O(n2^{3n/4})$.

After obtaining k_4, k_5 and k_6, we can construct a quantum circuit to calculate the first three rounds of the Feistel-KF construction. Hence, for arbitrary β and $\beta' \in \{0,1\}^{n/2}$ such that $\beta' \neq \beta$, we can easily compute $F_1(\beta \oplus k_1) \oplus k_2$ and $F_1(\beta' \oplus k_1) \oplus k_2$ to obtain $F_1(\beta \oplus k_1) \oplus F_1(\beta' \oplus k_1)$. Then, k_1 can be recovered in time $O(2^{n/4})$ by performing Grover's algorithm. Once k_1, k_4, k_5 and k_6 are known, k_2 and k_3 can be recovered trivially.

In summary, we have the following Corollary 3.

Corollary 3. *The whole attack recovering $(k_1, k_2, k_3, k_4, k_5, k_6)$ requires $O(1)$ classical queries, $O(n2^{3n/4})$ time, $O(n)$ qubits and $O(1)$ classical memory. Generally, the key recovery attack on r-round ($r > 3$) Feistel-KF constructions is a similar mechanism to that of the 6-round attack. The time complexity becomes $O(n2^{(r-3)n/4})$.*

When $Q = O(n2^p)$ qubits are available, according to Corollary 1, the key recovery attack on r-round Feistel-KF constructions can recover $nr/2$-bit key with $T = O(n2^{(r-3)n/4-p/2})$ time by using the parallelized Algorithm 2. The tradeoff is $QT^2 = \tilde{O}(2^{(r-3)n/2})$, which balances at $T = Q = \tilde{O}(2^{(r-3)n/6})$.

Comparison. In the Q1 model, we reduce the number of classical queries exponentially required for attacking Feistel-KF constructions. When $Q < 2^{n/2}$, our attack on the 6-round Feistel-KF construction requires a time $T = \tilde{O}(2^{3n/4}/\sqrt{Q})$, which is lower than $\tilde{O}(2^n/Q)$ from the previous Q1 attack [11]. For the case $r \geq 7$, our attack reduces the time complexity from $O(2^{rn/4-13n/12})$ of Xu et al's [36] to $O(2^{rn/4-7n/6})$, under the condition that $Q = \tilde{O}(2^{5n/6})$. Besides, in the case $Q = poly(n)$, we can improve the Q2 attack of Hosoyamada and Sasaki [11] on r-round ($r > 3$) Feistel-KF constructions by reducing the query complexity from exponential to constant, while the time complexity is $O(n2^{(r-3)n/4})$ instead of $O(n^3 2^{(r-3)n/4})$. What's more, when the attacks are compared in terms of a product $DTQM$ (this comparison was mentioned in [11]), which is calculated by multiplying the query complexity D, time complexity T, number of qubits Q and amount of classical memory M, our attack is optimal.

5 Quantum Attacks on Feistel-FK constructions

In this section, we present a quantum distinguishing attack against Feistel-FK constructions. Based on this, we show its quantum key recovery attacks. Once again, for the sake of simplicity, we assume that a single evaluation of each primitive, for example a block cipher, can be performed with the time complexity of $O(1)$.

5.1 Quantum Distinguishing Attacks

We present a quantum CPA distinguisher against the 4-round Feistel-FK construction (see Fig. 5). Since the round function F_i is a public permutation and the output of the first round function F_1 can be computed without the knowledge of the subkeys, we extend the analysis by incorporating the first round.

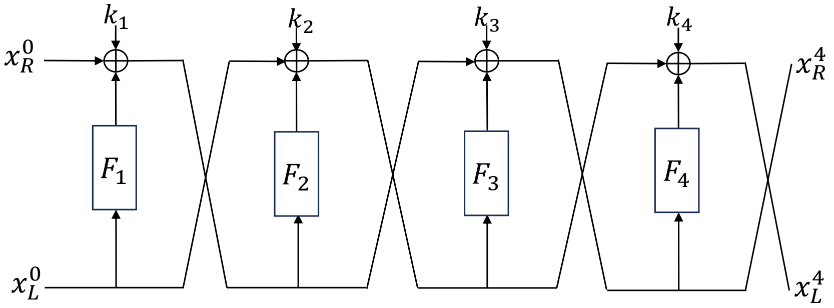

Fig. 5. The 4-round Feistel-FK construction.

In Fig. 5, F_1, F_2, F_3 and $F_4 \in Perm(n/2)$ are the public round functions, while k_1, k_2, k_3 and k_4 are the independently chosen subkeys. If we take

$(x_L^0, x_R^0) \in \{0,1\}^n$ as a plaintext, the corresponding ciphertext (x_L^4, x_R^4) can be computed by

$$\begin{aligned} x_L^4 &= F_4(x_R^0 \oplus F_1(x_L^0) \oplus k_1 \oplus F_3(x_L^0 \oplus F_2(x_R^0 \oplus F_1(x_L^0) \oplus k_1) \oplus k_2) \oplus k_3) \oplus k_4 \\ &\oplus x_L^0 \oplus F_2(x_R^0 \oplus F_1(x_L^0) \oplus k_1) \oplus k_2, \\ x_R^4 &= x_R^0 \oplus F_1(x_L^0) \oplus k_1 \oplus F_3(x_L^0 \oplus F_2(x_R^0 \oplus F_1(x_L^0) \oplus k_1) \oplus k_2) \oplus k_3. \end{aligned} \tag{19}$$

We define the function $f^{\mathcal{O}}$ as

$$\begin{aligned} f^{\mathcal{O}} : \{0,1\}^{n/2} &\rightarrow \{0,1\}^{n/2} \\ x &\mapsto c \oplus F_4(d) \oplus x, \end{aligned} \tag{20}$$

where $\mathcal{O}(x, \beta \oplus F_1(x)) = (c, d)$ and $\beta \in \{0,1\}^{n/2}$ be an arbitrary constant. If $\mathcal{O}$ is the 4-round Feistel-FK construction, the output pair (c, d) from $\mathcal{O}$ represents the ciphertext (x_L^4, x_R^4) corresponding to the plaintext $(x, \beta \oplus F_1(x))$. Then the function $f^{\mathcal{O}}$ can be described as

$$f^{\mathcal{O}}(x) = c \oplus F_4(d) \oplus x = F_2(\beta \oplus k_1) \oplus k_2 \oplus k_4. \tag{21}$$

That is, $f^{\mathcal{O}}(x)$ becomes a constant for any x.

Thus, we can build a quantum distinguisher against the 4-round Feistel-FK construction by utilizing the function $f^{\mathcal{O}}$ in Eq. (20). Similar to the analysis in Sect. 4.1, the distinguisher requires $O(1)$ queries and $O(n)$ time.

5.2 Quantum Key Recovery Attacks

As with the key recovery attacks against the Feistel-KF construction described in Sect. 4.2, the distinguisher mentioned above can be combined with Algorithm 2 to formulate key-recovery attacks. In the quantum CPA setting, the keys of the r-round ($r > 4$) Feistel-KF construction can be recovered with $O(1)$ classical queries.

Our attack idea follows the attack against the Feistel-KF construction. Recall that the attack in Sect. 4.2 guesses the last $(r-3)$-round subkeys since a 3-round distinguisher is available. On the other hand, for the Feistel-FK construction, we can use the 4-round distinguisher instead of the 3-round distinguisher (see Fig. 6). Hence, it is sufficient to guess only the last $(r-4)$-round subkeys (instead of the last $r-3$-round subkeys) in attacking the Feistel-FK construction. As a result, Algorithm 2 can be applied to find the last $(r-4)$-round subkeys (i.e., $(r-4)n/2$ bits) in time $O(n2^{(r-4)n/4})$, requiring O(1) classical queries, $O(n)$ qubits and $O(1)$ classical memory.

After recovering the last $(r-4)$-round subkeys, we can construct a quantum circuit that calculates the first four rounds. Hence, if we take arbitrary $\beta, \beta' \in \{0,1\}^{n/2}$ such that $\beta' \neq \beta$, we can easily compute $F_2(\beta \oplus k_1) \oplus k_2 \oplus k_4$ and $F_2(\beta' \oplus k_1) \oplus k_2 \oplus k_4$ to obtain $F_2(\beta \oplus k_1) \oplus F_2(\beta' \oplus k_1)$. Then k_1 can be recovered

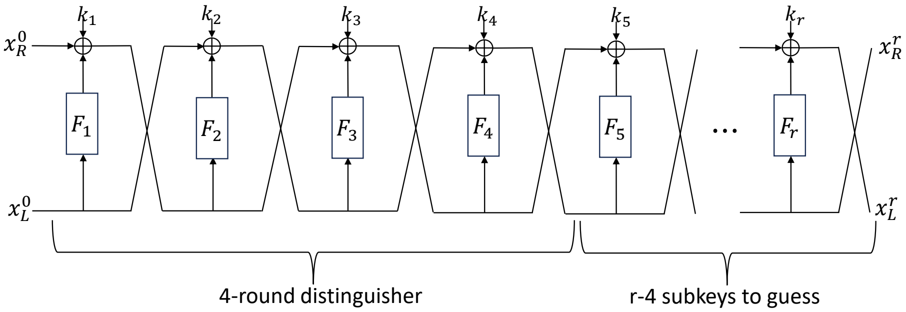

Fig. 6. Key recovery attacks on r-round Feistel-FK constructions.

in time $O(2^{n/4})$ by employing Grover's algorithm. Besides, when the plaintext is $(x, \beta \oplus F_1(x))$, according to Eq. (19), x_R^4 can be written as

$$x_R^4 = \beta \oplus k_1 \oplus F_3(x \oplus F_2(\beta \oplus k_1) \oplus k_2) \oplus k_3. \tag{22}$$

Taking x as α and α' respectively, we can easily compute $\beta \oplus k_1 \oplus F_3(\alpha \oplus F_2(\beta \oplus k_1) \oplus k_2) \oplus k_3$ and $\beta \oplus k_1 \oplus F_3(\alpha' \oplus F_2(\beta \oplus k_1) \oplus k_2) \oplus k_3$ to obtain $F_3(\alpha \oplus F_2(\beta \oplus k_1) \oplus k_2) \oplus F_3(\alpha' \oplus F_2(\beta \oplus k_1) \oplus k_2)$. Since k_1 is known, k_2 can be recovered in time $O(2^{n/4})$ by utilizing the Grover search. k_3 and k_4 can be recovered trivially.

In summary, we obtain the following Corollary 4.

Corollary 4. *Our key recovery attack on r-round ($r > 4$) Feistel-FK constructions can recover $nr/2$-bit key with $O(1)$ classical queries, $O(n2^{(r-4)n/4})$ time, $O(n)$ qubits and $O(1)$ classical memory.*

When $Q = O(n2^p)$ qubits are available, according to Corollary 1, the key can be recovered in time $T = O(n2^{(r-4)n/4-p/2})$ by using the parallelized Algorithm 2. The tradeoff is $QT^2 = \tilde{O}(2^{(r-4)n/2})$, which balances at $T = Q = \tilde{O}(2^{(r-4)n/6})$.

Comparison. In the Q1 model, when $Q = \tilde{O}(2^{n/3})$ qubits are available, our attack on the 6-round Feistel-FK construction requires the same query complexity and time complexity as that of the previous one [22]. But we only use $O(1)$ classical memory instead of $O(2^{n/3})$. Besides, for the case $r \geq 6$, our key recovery attack reduces the quantum query complexity from exponential in [12] to constant, under the condition that $Q = poly(n)$. Although our time complexity is slightly higher than that in [12], our attack belongs to the Q1 model, which is more realistic.

6 Conclusion

In this paper, we utilized constant functions, instead of periodic functions, to improve the quantum chosen-plaintext key recovery attack on Feistel-KF constructions and Feistel-FK constructions in the Q1 model for the first time. Before that, we introduced the conditional constant function problem, and proposed a quantum algorithm to solve one of its special variants based on the parallelized Grover search. We constructed constant functions to distinguish the 3-round Feistel-KF construction and the 4-round Feistel-FK construction from random permutations, which requires $O(1)$ classical queries instead of $O(n)$ quantum queries. After obtaining quantum distinguishers, we combined them with our quantum algorithm to devise the key recovery attack on the Feistel-KF and Feistel-FK constructions. Compared with previous attacks (including Q1 attack and Q2 attack) on the Feistel-KF construction, our attack not only reduces the number of classical queries exponentially, but also requires less time. For the 6-round Feistel-FK construction, our attack outperforms the previous Q1 attack in terms of the classical memory complexity. Compared with the previous Q2 attack, we also reduce the query complexity significantly.

The acceleration of our attacks on Feistel-KF and Feistel-FK comes from two aspects. One is our quantum algorithm solving the conditional constant function problem, which reduces the number of queries to constant. The other is the parallelized Grover search, which generates the tradeoffs between time complexity and number of qubits. In the future, there remain several interesting research directions worthy of exploration. First, further improving the time complexity or extending the number of rounds in attacks against Feistel ciphers presents an interesting challenge. Second, it would be valuable to explore the extension of our attack to other symmetric-key schemes, such as the generalized Feistel construction [46], MISTY construction [25], FX construction [14], Lai-Massey construction [34] and others.

Acknowledge. This work was supported by the National Natural Science Foundation of China (Grant Nos. 62372048, 62272056, 62301454, 62471070), the Fundamental Research Funds for the Central Universities (Grant No. SWU-KQ22049), the Fundamental Research Funds for Heilongjiang Universities (Grant No. 2024-KYYWF-0137), the Natural Science Foundation of Chongqing (Grant No. CSTB2023NSCQ-MSX0739), the National Key Research and Development Program of China (2020YFA0309704).

References

1. Aoki, K., Ichikawa, T., Kanda, M., Matsui, M., Moriai, S., Nakajima, J., Tokita, T.: Camellia: A 128-bit block cipher suitable for multiple platformsdesign andanalysis. In: Selected Areas in Cryptography: 7th Annual International Workshop, SAC 2000 Waterloo, Ontario, Canada, August 14–15, 2000 Proceedings 7. pp. 39–56. Springer (2001)
2. Barenco, A., Bennett, C.H., Cleve, R., DiVincenzo, D.P., Margolus, N., Shor, P., Sleator, T., Smolin, J.A., Weinfurter, H.: Elementary gates for quantum computation. Physical review A **52**(5), 3457 (1995)
3. Brassard, G., Hoyer, P., Mosca, M., Tapp, A.: Quantum amplitude amplification and estimation. Contemporary Mathematics **305**, 53–74 (2002)

4. Chartouny, M., Cogliati, B., Patarin, J.: Classical and quantum generic attacks on 6-round feistel schemes. Cryptology ePrint Archive (2024)
5. Chartouny, M., Patarin, J., Toulemonde, A.: Quantum cryptanalysis of 5 rounds feistel schemes and benes schemes. In: International Conference on Codes, Cryptology, and Information Security. pp. 196–203. Springer (2023)
6. Childs, A.M., Eisenberg, J.M.: Quantum algorithms for subset finding. arXiv preprint quant-ph/0311038 (2003)
7. Dong, X., Wang, X.: Quantum key-recovery attack on feistel structures. Science China Information Sciences **61**(10), 102501 (2018)
8. Grover, L., Rudolph, T.: How significant are the known collision and element distinctness quantum algorithms. Quantum Info. Comput. **4**(3), 201206 (may 2004)
9. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. pp. 212–219 (1996)
10. Harrow, A.W., Hassidim, A., Lloyd, S.: Quantum algorithm for linear systems of equations. Physical review letters **103**(15), 150502 (2009)
11. 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. pp. 386–403. Springer (2018)
12. Ito, G., Hosoyamada, A., Matsumoto, R., Sasaki, Y., Iwata, T.: 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. pp. 391–411. Springer (2019)
13. Jaques, S., Naehrig, M., Roetteler, M., Virdia, F.: Implementing grover oracles for quantum key search on aes and lowmc. In: Advances in Cryptology–EUROCRYPT 2020: 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10–14, 2020, Proceedings, Part II 30. pp. 280–310. Springer (2020)
14. Kilian, J., Rogaway, P.: How to protect des against exhaustive key search. In: Advances in CryptologyCRYPTO96: 16th Annual International Cryptology Conference Santa Barbara, California, USA August 18–22, 1996 Proceedings 16. pp. 252–267. Springer (1996)
15. Kim, P., Han, D., Jeong, K.C.: Time–space complexity of quantum search algorithms in symmetric cryptanalysis: applying to aes and sha-2. Quantum Information Processing **17**(12), 339 (2018)
16. Kuwakado, H., Morii, M.: Quantum distinguisher between the 3-round feistel cipher and the random permutation. In: 2010 IEEE international symposium on information theory. pp. 2682–2685. IEEE (2010)
17. Kuwakado, H., Morii, M.: Security on the quantum-type even-mansour cipher. In: 2012 international symposium on information theory and its applications. pp. 312–316. IEEE (2012)
18. Leander, G., May, A.: Grover meets simon–quantumly attacking the fx-construction. In: Advances in Cryptology–ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II 23. pp. 161–178. Springer (2017)
19. Li, L., Li, J., Song, Y., Qin, S., Wen, Q., Gao, F.: An efficient quantum proactive incremental learning algorithm. Science China Physics, Mechanics & Astronomy **68**(1), 1–9 (2025)

20. Liu, H.L., Wu, Y.S., Wan, L.C., Pan, S.J., Qin, S.J., Gao, F., Wen, Q.Y.: Variational quantum algorithm for the poisson equation. Physical Review A **104**(2), 022418 (2021)
21. Liu, N., Rebentrost, P.: Quantum machine learning for quantum anomaly detection. Physical Review A **97**(4), 042315 (2018)
22. Liu, W., Wang, M., Li, Z.: Quantum all-subkeys-recovery attacks on 6-round feistel-2* structure based on multi-equations quantum claw finding. Quantum Information Processing **22**(3), 142 (2023)
23. Lloyd, S., Mohseni, M., Rebentrost, P.: Quantum principal component analysis. Nature physics **10**(9), 631–633 (2014)
24. Mao, S., Guo, T., Wang, P., Hu, L.: Quantum attacks on lai-massey structure. In: International Conference on Post-Quantum Cryptography. pp. 205–229. Springer (2022)
25. Matsui, M.: New block encryption algorithm misty. In: Fast Software Encryption: 4th International Workshop, FSE97 Haifa, Israel, January 20–22 1997 Proceedings 4. pp. 54–68. Springer (1997)
26. Nielsen, M.A., Chuang, I.L.: Quantum computation and quantum information, vol. 2. Cambridge university press Cambridge (2001)
27. Pan, S.J., Wan, L.C., Liu, H.L., Wang, Q.L., Qin, S.J., Wen, Q.Y., Gao, F.: Improved quantum algorithm for a-optimal projection. Physical Review A **102**(5), 052402 (2020)
28. National Academies of Sciences, E., Medicine: Quantum Computing: Progress and Prospects. The National Academies Press, Washington, DC (2019). https://doi.org/10.17226/25196, `https://nap.nationalacademies.org/catalog/25196/quantum-computing-progress-and-prospects`
29. Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T., Shirai, T.: Piccolo: an ultra-lightweight blockcipher. In: Cryptographic Hardware and Embedded Systems–CHES 2011: 13th International Workshop, Nara, Japan, September 28–October 1, 2011. Proceedings 13. pp. 342–357. Springer (2011)
30. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM review **41**(2), 303–332 (1999)
31. Simon, D.R.: On the power of quantum computation. SIAM journal on computing **26**(5), 1474–1483 (1997)
32. Song, Y., Wu, Y., Wu, S., Li, D., Wen, Q., Qin, S., Gao, F.: A quantum federated learning framework for classical clients. Science China Physics, Mechanics & Astronomy **67**(5), 250311 (2024)
33. Standard, D.E., et al.: Data encryption standard. Federal Information Processing Standards Publication **112**, 3 (1999)
34. Vaudenay, S.: On the lai-massey scheme. In: Advances in Cryptology-ASIACRYPT99: International Conference on the Theory and Application of Cryptology and Information Security, Singapore, November 14-18, 1999. Proceedings. pp. 8–19. Springer (1999)
35. Wan, L.C., Yu, C.H., Pan, S.J., Gao, F., Wen, Q.Y., Qin, S.J.: Asymptotic quantum algorithm for the toeplitz systems. Physical Review A **97**(6), 062322 (2018)
36. Xu, Y., Yuan, Z.: Quantum meet-in-the-middle attack on feistel construction. Quantum Information Processing **22**(3), 155 (2023)
37. Yang, D., Qi, W.F., Tian, T.: All-subkeys-recovery attacks on a variation of feistel-2 block ciphers. IET Information Security **11**(5), 230–234 (2017)
38. Yang, G., Zhu, B., Suder, V., Aagaard, M.D., Gong, G.: The simeck family of lightweight block ciphers. In: International workshop on cryptographic hardware and embedded systems. pp. 307–329. Springer (2015)

39. Yu, B., Shi, T., Dong, X., Shen, X., Luo, Y., Sun, B.: Quantum attacks: A view of data complexity on offline simons algorithm. In: International Conference on Information Security and Cryptology. pp. 329–342. Springer (2023)
40. Yu, C.H., Gao, F., Lin, S., Wang, J.: Quantum data compression by principal component analysis. Quantum Information Processing **18**(8), 249 (2019)
41. Yu, C.H., Gao, F., Wen, Q.Y.: An improved quantum algorithm for ridge regression. IEEE Transactions on Knowledge and Data Engineering **33**(3), 858–866 (2019)
42. Zalka, C.: Grovers quantum searching algorithm is optimal. Physical Review A **60**(4), 2746 (1999)
43. Zhandry, M.: A note on the quantum collision and set equality problems. arXiv preprint arXiv:1312.1027 (2013)
44. Zhandry, M.: How to construct quantum random functions. Journal of the ACM (JACM) **68**(5), 1–43 (2021)
45. Zhang, S.: Promised and distributed quantum search. In: Computing and Combinatorics: 11th Annual International Conference, COCOON 2005 Kunming, China, August 16–19, 2005 Proceedings 11. pp. 430–439. Springer (2005)
46. Zheng, Y., Matsumoto, T., Imai, H.: On the construction of block ciphers provably secure and not relying on any unproved hypotheses. In: Advances in Cryptology-CRYPTO89 Proceedings 9. pp. 461–480. Springer (1990)