Quantum Chosen-Ciphertext Attacks against Feistel Ciphers*

Gembu Ito[1], Akinori Hosoyamada[1,2], Ryutaroh Matsumoto[1,3], Yu Sasaki[2], and Tetsu Iwata[1]

[1] Nagoya University, Nagoya, Japan
g_itou@echo.nuee.nagoya-u.ac.jp, ryutaroh.matsumoto@nagoya-u.jp, tetsu.iwata@nagoya-u.jp
[2] NTT Secure Platform Laboratories, Tokyo, Japan
hosoyamada.akinori@lab.ntt.co.jp, sasaki.yu@lab.ntt.co.jp
[3] Aalborg University, Aalborg, Denmark

Abstract. Seminal results by Luby and Rackoff show that the 3-round Feistel cipher is secure against chosen-plaintext attacks (CPAs), and the 4-round version is secure against chosen-ciphertext attacks (CCAs). However, the security significantly changes when we consider attacks in the quantum setting, where the adversary can make superposition queries. By using Simon's algorithm that detects a secret cycle-period in polynomial-time, Kuwakado and Morii showed that the 3-round version is insecure against quantum CPA by presenting a polynomial-time distinguisher. Since then, Simon's algorithm has been heavily used against various symmetric-key constructions. However, its applications are still not fully explored.

In this paper, based on Simon's algorithm, we first formalize a sufficient condition of a quantum distinguisher against block ciphers so that it works even if there are multiple collisions other than the real period. This distinguisher is similar to the one proposed by Santoli and Schaffner, and it does not recover the period. Instead, we focus on the dimension of the space obtained from Simon's quantum circuit. This eliminates the need to evaluate the probability of collisions, which was needed in the work by Kaplan et al. at CRYPTO 2016. Based on this, we continue the investigation of the security of Feistel ciphers in the quantum setting. We show a quantum CCA distinguisher against the 4-round Feistel cipher. This extends the result of Kuwakado and Morii by one round, and follows the intuition of the result by Luby and Rackoff where the CCA setting can extend the number of rounds by one. We also consider more practical cases where the round functions are composed of a public function and XORing the subkeys. We show the results of both distinguishing and key recovery attacks against these constructions.

Keywords: Feistel cipher · Quantum chosen-ciphertext attacks · Simon's algorithm

* This is the full version of the paper that appears in the proceedings of CT-RSA 2019.

1 Introduction

A block cipher is an important cryptographic primitive that is widely adopted in various secure communication protocols and security products. A block cipher is a pseudo-random permutation (PRP), i.e. it takes a key as input and provides distinct permutations that cannot be distinguished from a random permutation for distinct key inputs.

Designing an efficient block cipher is a long-term challenge in symmetric-key cryptography. One of the most popular approaches is to use the Feistel network, in which an n-bit state is divided into $n/2$-bit halves denoted by a_i and b_i, and the state is updated by iteratively applying the following two operations;

$$b_{i+1} \leftarrow a_i \oplus F_{K_i}(b_i), \qquad a_{i+1} \leftarrow b_i,$$

where F_{K_i} is a keyed function taking a subkey K_i as input. The construction is known as the Luby-Rackoff construction. In this paper, we call it *Feistel-F* to make the name consistent with other constructions. The diagram of the construction is drawn in the left of Fig. 1. Luby and Rackoff [18] proved that when F_{K_i} is a pseudo-random function (PRF), 3-round and 4-round Feistel ciphers are PRPs up to $O(2^{n/4})$ queries against chosen-plaintext attacks (CPAs) and chosen-ciphertext attacks (CCAs), respectively. Luby and Rackoff also showed the tightness of the number of rounds by demonstrating efficient attacks against 2 and 3 rounds in the corresponding attack models.

While the provable security bounds derived by Luby and Rackoff are attractive, using a PRF for F_{K_i} requires significant implementation costs, and this is often practically infeasible. To design a block cipher for practical usage, the subkey space is often limited to $\{0,1\}^{n/2}$, and $F_{K_i}(b_i)$ is defined as

$$b_{i+1} \leftarrow a_i \oplus F(K_i \oplus b_i), \qquad a_{i+1} \leftarrow b_i,$$

where F is a public function. In this paper, we call this construction *Feistel-KF*. See the middle figure of Fig. 1. Feistel-KF includes a lot of practical designs, e.g. DES [19] and Camellia [1], where the function $x \mapsto F(K_i \oplus x)$ is not a PRF, and generic attacks on this construction have been widely studied, e.g. impossible differential attacks [14], meet-in-the-middle attacks [12,10], dissection attacks [5] and division property [23].

It is also possible to inject a subkey $K_i \in \{0,1\}^{n/2}$ outside the F function as

$$b_{i+1} \leftarrow a_i \oplus F(b_i) \oplus K_i, \qquad a_{i+1} \leftarrow b_i.$$

We call this construction *Feistel-FK*, which is illustrated on the right of Fig. 1. This construction provides implementation advantages and can be seen in several lightweight designs e.g. Piccolo [21], Simon [2] and Simeck [24].

The discussion so far is about the classical computation setting, while the security of symmetric-key schemes against quantum computers has become active recently. Owing to less mathematical structure in symmetric-key schemes than public-key schemes, there was a belief that simply doubling the key size in order

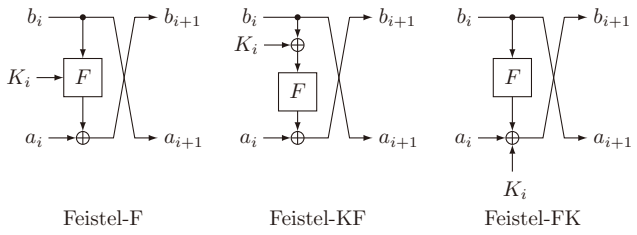

Fig. 1. Our target constructions.

to resist the exhaustive key search by Grover's algorithm [9] is sufficient to protect symmetric-key schemes from quantum computers. However, Kuwakado and Morii [15] demonstrated that, by exploiting Simon's algorithm [22], the Feistel ciphers can be distinguished from a random permutation only in polynomial-time of the output size under the assumption that the adversary can make quantum superposition queries. Since then, many polynomial-time attacks using Simon's algorithm have been proposed e.g. key recovery against Even-Mansour construction [16], forgery on various CBC-like MACs [13], and cryptanalysis of AEZ [3]. Moreover, Leander and May [17] showed a clever method to combine Grover's and Simon's algorithms to recover the key against the FX construction. See also [20].

The attack model that adversaries can make quantum queries is worth investigating. This model is a natural extension of the classical attack models, and theoretically interesting. Any symmetric scheme broken in this model should not be implemented on a quantum computer. Moreover, the threat of this attack model becomes significant if an adversary has access to its white-box implementation. Because arbitrary classical circuit can be converted into quantum one, the adversary can construct a quantum circuit from the classical source code given by the white-box implementation.

There are several attacks on Feistel ciphers in the quantum setting. Besides the first work in [15], a meet-in-the-middle attack in the quantum setting was discussed in [11] and appending key-recovery rounds by applying the algorithm by Leander and May [17] was discussed in [11,8,7]. However, the following important issues have not been discussed by the previous work.

- Security analysis of Feistel ciphers against chosen-ciphertext adversaries is missing. In the classical setting, the tight bound of the number of rounds is known for the Feistel-F construction, and clarifying the number of rounds that can be attacked in the quantum setting leads us a deeper understanding of the Feistel-F construction. Furthermore, the quantum setting assumes strong power of adversaries, hence considering CCAs is more reasonable. We note that there are results in a CCA setting on Feistel ciphers with a specific key scheduling function called 2 key- or 4 key-alternation Feistel ciphers and their variants [6,4], however, we are considering more general constructions.

– Discussion on practical constructions is missing. Although the Luby-Rackoff construction is a good object to study theoretical aspects of the Feistel ciphers, in general, it cannot be implemented efficiently in practice. Therefore, the analyses of practical constructions like Feistel-KF and Feistel-FK are needed. Again, we are interested in general constructions that do not rely on a specific key scheduling function.

Our Contributions. In this paper, we further investigate the security of the Feistel ciphers against quantum adversaries. In particular, we show CCA distinguishers that can distinguish more rounds than the previous CPA distinguishers. In addition, we extend the distinguishers to key recovery attacks for the practical constructions, i.e. Feistel-KF and Feistel-FK.

We start with several fundamental observations about Simon's algorithm that detects a secret cycle-period in polynomial-time. The usage of Simon's algorithm in the previous work can be classified into two types; the first type uses Simon's algorithm for key recovery attacks, namely, the recovered secret cycle-period corresponds to the key of the construction such as [16] and [13], whereas the second type uses Simon's algorithm for distinguishers, e.g. to distinguish the construction from an ideal one [15,20] or to distinguish the right key guess from wrong key guesses [17,11,8,7].

We observe that, for the second type, recovering the secret cycle-period is not necessary as long as a non-ideal behavior is detected. If we follow [13] to recover the secret cycle-period by using Simon's algorithm, one has to derive the upper bound on the probability of a collision other than the period. However, there are cases where obtaining the upper bound is non-obvious, and it may be difficult to prove it in attacks on complicated constructions. This motivates us to relax the requirement of recovering the period in Simon's algorithm. Technically, we focus on the property that the dimension of the space spanned by the vectors in Simon's algorithm, instead of the exact period s. Namely, the dimension of the space is at most $\ell - 1$ if the target function has a period s, where $\{0,1\}^\ell$ is the domain of the function evaluated by Simon's algorithm. This modification eliminates the need to derive the upper bound on the probability of a collision other than the period s. Note that Santoli and Schaffner pointed out a similar observation [20], and we are dealing with a general class of block ciphers, and we also formalize a sufficient condition so that the distinguisher works.

We then apply the above observations to attack several Feistel ciphers. For the Feistel-F construction, we show that a cycle-period can be formed for 4 rounds in the CCA setting. This leads to a 4-round polynomial-time CCA distinguisher, which is 1-round longer than the CPA distinguisher by Kuwakado and Morii [15]. The attack is then extended to the practical constructions; Feistel-KF and Feistel-FK. For Feistel-KF, although the distinguisher is the same as the one for Feistel-F, we can now discuss the key recovery attack owing to the practical size of the secret key. We obtain 7-round key recovery attacks that recover $7n/2$-bit key with $O(2^{3n/4})$ complexity. For Feistel-FK, the CCA distinguisher is extended to 6 rounds and we obtain 9-round key recovery attacks that recover

Table 1. Comparison of the number of attacked rounds in various settings. "Dist." and "KR" denote distinguisher and key recovery attack, respectively. Superscript ^{P} denotes that the attack complexity is only a polynomial of the function's output size, while the others require exponential complexity.

Construction	Classic-CPA		Classic-CCA		Quantum-CPA		Quantum-CCA	
	Dist.	KR	Dist.	KR	Dist.	KR	Dist.	KR
Feistel-F	2 [18]	-	3 [18]	-	3^P [15]	-	4^P Ours	-
Feistel-KF	5 [10]	6 [10]	5 [10]	6 [10]	5 [11] 3^P [15]	6 [11]	4^P Ours	7 Ours
Feistel-FK	-	-	-	-	5^P Ours	8 Ours	6^P Ours	9 Ours

$9n/2$-bit key with $O(2^{3n/4})$ complexity. In addition, the CPA distinguisher is extended to 5 rounds and we obtain 8-round key recovery attacks that recover $8n/2$-bit key with $O(2^{3n/4})$ complexity. A comparison of the number of attacked rounds is given in Table 1. Note that Table 1 focuses on attacks with complexity at most $O(2^n)$, and it does not include attacks with higher complexities. Also, we consider only general constructions, so it does not include attacks against constructions with a particular key scheduling function such as [6,4].

Paper Outline. This paper is organized as follows. Section 2 describes preliminaries. Section 3 introduces previous works. Section 4 explains the formalization of a distinguishing technique that relaxes Simon's algorithm. Section 5 presents our CCA distinguisher against the 4-round Feistel-F constructions. The attack is then applied to chosen-ciphertext key-recovery attacks on Feistel-KF constructions in Sect. 6. Section 7 explains distinguishing and key-recovery attacks against Feistel-FK constructions in both CCA and CPA settings. We conclude the paper in Sect. 8.

2 Preliminaries

2.1 Notation

For a positive integer n, let $\{0,1\}^n$ be the set of all n-bit strings. Let $\mathrm{Perm}(n)$ be the set of all permutations on $\{0,1\}^n$, and let $\mathrm{Func}(n)$ be the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$. For bit strings a and b, $a \,\|\, b$ denotes their concatenation. We also regard a and b as binary vectors, and let $|a|$ be the dimension of the vector a. When $|a| = |b|$, we denote their inner product as $a \cdot b$. In this paper, e denotes Napier's number. For a finite set $\mathcal{X}$, we write $X \xleftarrow{\$} \mathcal{X}$ for the process of sampling an element uniformly from $\mathcal{X}$ and assigning the result to X.

2.2 Simon's Algorithm

In this section, we describe Simon's algorithm [22] that is used in our quantum algorithms. Throughout this paper, we assume that readers have basic knowledge

about quantum computation. Simon's algorithm can solve the following problem.

Problem 1. Given a function $f : \{0,1\}^n \to \{0,1\}^n$, assume that there exists a period $s \in \{0,1\}^n \backslash \{0^n\}$ such that for any distinct $x, x' \in \{0,1\}^n$, it holds that $f(x) = f(x') \Leftrightarrow x' = x \oplus s$. The goal is to find the period s.

We assume that Simon's algorithm has access to the quantum oracle U_f, which is defined as $U_f |x\rangle |z\rangle = |x\rangle |z \oplus f(x)\rangle$. We use the Hadamard transform $H^{\otimes n}$ that is applied on n-qubit state $|x\rangle$ and gives $H^{\otimes n} |x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle$. Simon proposed a circuit $\mathcal{S}_f$ that computes vectors that are orthogonal to s by using the quantum oracle U_f. $\mathcal{S}_f$ is described as $(H^{\otimes n} \otimes I_n) \cdot U_f \cdot (H^{\otimes n} \otimes I_n)$ and works as follows:

1. We first apply the Hadamard transform $H^{\otimes n}$ on the first n qubits of $2n$-qubit state $|0^n\rangle |0^n\rangle$ to obtain the state $\frac{1}{\sqrt{2^n}} \sum_x |x\rangle |0^n\rangle$.
2. Then, we apply the unitary operator U_f to obtain the state $\frac{1}{\sqrt{2^n}} \sum_x |x\rangle |f(x)\rangle$.
3. Finally, we apply the Hadamard transform $H^{\otimes n}$ on the first n qubits to obtain the state
$$\frac{1}{2^n} \sum_{x,y} (-1)^{x \cdot y} |y\rangle |f(x)\rangle . \tag{1}$$

As we assume that f satisfies $f(x) = f(x') \Leftrightarrow x' = x \oplus s$, we have $|y\rangle |f(x)\rangle = |y\rangle |f(x \oplus s)\rangle$ for each y and x. Therefore, equation (1) is described as

$$\frac{1}{2^n} \sum_{x \in V, y} \left((-1)^{x \cdot y} + (-1)^{(x \oplus s) \cdot y} \right) |y\rangle |f(x)\rangle ,$$

where V is a linear subspace of $\{0,1\}^n$ of dimension $n-1$ that partitions $\{0,1\}^n$ into cosets V and $V + s$. The vector y such that $y \cdot s \equiv 1 \pmod 2$ will satisfy $(-1)^{x \cdot y} + (-1)^{(x \oplus s) \cdot y} = 0$. Thus, we will obtain a random vector y such that $y \cdot s \equiv 0 \pmod 2$ by measuring the first n qubits. By repeating this routine that obtains a random vector y for $O(n)$ times, with a high probability, we obtain $n-1$ linearly independent such vectors, and then the period s can be recovered by solving the system of linear equations.

We note that, in Simon's algorithm, we assume that the function f has a period s. In latter sections, we will use the circuit $\mathcal{S}_f$ to a function f that may not have any period, or may have multiple periods.

2.3 Kaplan et al.'s Observation

To apply Simon's algorithm, the function f has to satisfy $f(x) = f(x') \Leftrightarrow x' = x \oplus s$. We call this property Simon's promise. If f does not satisfy this property and has other collisions in addition to s, then there is no guarantee that Simon's algorithm works. However, Kaplan et al. showed that Simon's algorithm can find

s even if f has partial periods, where the partial period is defined as $t \neq s$ such that $f(x) = f(x \oplus t)$ holds for some x [13].

More precisely, suppose that a function $f : \{0,1\}^\ell \to \{0,1\}^m$ satisfies only the condition that $f(x) = f(x') \Leftarrow x' = x \oplus s$ for any distinct $x, x' \in \{0,1\}^\ell$. Since now the counter condition $f(x) = f(x') \Rightarrow x' = x \oplus s$ does not always hold, there may exist partial periods of f. Intuitively, if there exist many partial periods $t_1, t_2, \ldots$ which are very close to complete periods (i.e., $\Pr_x [f(x) = f(x \oplus t_j)]$ is close to 1 for each j), then it becomes hard to recover s. To describe this intuition formally, Kaplan et al. introduced the parameter $\epsilon(f, s)$ defined as

$$\epsilon(f, s) = \max_{t \in \{0,1\}^\ell \setminus \{0^\ell, s\}} \Pr_x [f(x) = f(x \oplus t)]. \tag{2}$$

This shows the maximum probability of partial periods of f. Notice that if f is a constant function, then $\epsilon(f, s) = 1$ and s cannot be recovered. On the other hand, if f satisfies Simon's promise, then $\epsilon(f, s) = 0$. The following theorem about the success probability of Kaplan et al.'s observation was proved.

Theorem 1 ([13]). *If $\epsilon(f, s) \leq p_0$ for some positive number $p_0 < 1$, the probability that Simon's algorithm returns s after $c\ell$ queries is at least $1 - (2(\frac{1+p_0}{2})^c)^\ell$.*

This theorem shows that we still obtain s with $O(\ell)$ quantum queries and the complexity does not increase significantly.

3 Previous Works

3.1 Quantum Distinguisher against the 3-Round Feistel Cipher

Here we review the distinguishing algorithm of the 3-round Feistel cipher by Kuwakado and Morii [15]. Kuwakado and Morii considered the case where F_{K_i} in Fig. 1 is a random permutation, and we write P_i for F_{K_i}.

Let FP_3 denote the encryption algorithm of the 3-round Feistel cipher, where random permutations $P_1, P_2, P_3 \xleftarrow{\$} \mathrm{Perm}(n/2)$ are used as internal functions. FP_3 takes a plaintext $(a, b) \in (\{0,1\}^{n/2})^2$ as input and outputs a ciphertext $(c, d) \in (\{0,1\}^{n/2})^2$, where

$$\begin{aligned} c &= b \oplus P_2(a \oplus P_1(b)), \\ d &= a \oplus P_1(b) \oplus P_3(b \oplus P_2(a \oplus P_1(b))). \end{aligned}$$

Figure 2 illustrates FP_3.

Kuwakado and Morii considered the following problem.

Problem 2. Let $\mathcal{O} : \{0,1\}^n \to \{0,1\}^n$ be either FP_3 or a random permutation $\Pi \xleftarrow{\$} \mathrm{Perm}(n)$. Given access to the quantum oracle $U_\mathcal{O} : |x\rangle |y\rangle \mapsto |x\rangle |y \oplus \mathcal{O}(x)\rangle$, where $x, y \in \{0,1\}^n$, the goal is to distinguish the two cases.

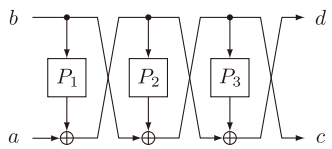

Fig. 2. The 3-round Feistel cipher with $P_i \xleftarrow{\$} \mathrm{Perm}(n/2)$ being used as the internal function.

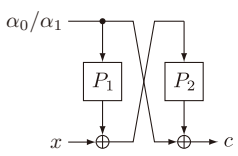

Fig. 3. $\mathrm{FP}_3(x, \alpha_\beta)$ and the lower half c of the ciphertext.

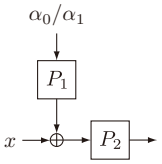

Fig. 4. $P_2(x \oplus P_1(\alpha_\beta))$.

Let $\alpha_0, \alpha_1 \in \{0,1\}^{n/2}$ be arbitrary distinct constants. For $\beta \in \{0,1\}$ and $x \in \{0,1\}^n$, Kuwakado and Morii used (x, α_β) as the plaintext (a, b). When $\mathcal{O}$ is FP_3, the lower half c of the ciphertext is described as

$$c = \alpha_\beta \oplus P_2(x \oplus P_1(\alpha_\beta)).$$

Figure 3 illustrates c. Then, we see that $c \oplus \alpha_\beta = P_2(x \oplus P_1(\alpha_\beta))$ holds, which is illustrated in Fig. 4. If we change the value of β, i.e., if we let β to $\beta \oplus 1$, we see that the input value of P_2 remains the same value by changing x to $x \oplus P_1(\alpha_0) \oplus P_1(\alpha_1)$. Thus, we can construct a function $f^{\mathcal{O}}(\beta \parallel x)$ that has the period $1 \parallel P_1(\alpha_0) \oplus P_1(\alpha_1)$ by defining $f^{\mathcal{O}}$ as

$$\begin{aligned} f^{\mathcal{O}} : \{0,1\} \times \{0,1\}^{n/2} &\to \{0,1\}^{n/2} \\ (\beta \parallel x) &\mapsto c \oplus \alpha_\beta, \quad \text{where } (c,d) = \mathcal{O}(x, \alpha_\beta). \end{aligned} \tag{3}$$

Note that $f^{\mathcal{O}}$ can also be evaluated in quantum superpositions. We can realize the unitary operator $U_{f^{\mathcal{O}}} : |x\rangle |y\rangle \mapsto |x\rangle |y \oplus f^{\mathcal{O}}(x)\rangle$ which makes $O(1)$ quantum queries to $U_{\mathcal{O}}$. If $\mathcal{O}$ is FP_3, then the function $f^{\mathcal{O}}$ is described as

$$\begin{aligned} f^{\mathcal{O}}(\beta \parallel x) &= \alpha_\beta \oplus P_2(x \oplus P_1(\alpha_\beta)) \oplus \alpha_\beta \\ &= P_2(x \oplus P_1(\alpha_\beta)), \end{aligned}$$

and the following lemma holds.

Lemma 1. *If $\mathcal{O}$ is FP_3, the function $f^{\mathcal{O}}$ satisfies $f^{\mathcal{O}}(\beta \parallel x) = f^{\mathcal{O}}(\beta' \parallel x') \Leftrightarrow \beta' \parallel x' = (\beta \parallel x) \oplus (1 \parallel P_1(\alpha_0) \oplus P_1(\alpha_1))$ for any $x, x' \in \{0,1\}^{n/2}$ such that $x \neq x'$. That is, $f^{\mathcal{O}}$ has the period $s = 1 \parallel (P_1(\alpha_0) \oplus P_1(\alpha_1))$.*

For completeness, we present a proof in Appendix A.

Lemma 1 guarantees that the function $f^{\mathcal{O}}$ defined in equation (3) satisfies Simon's promise if $\mathcal{O}$ is FP_3, and we can recover the period s by applying Simon's algorithm to $f^{\mathcal{O}}$. Define a unitary operator $\mathcal{S}_{f^{\mathcal{O}}}$ by $\mathcal{S}_{f^{\mathcal{O}}} = (H^{\otimes n/2+1} \otimes I_{n/2}) \cdot U_{f^{\mathcal{O}}} \cdot (H^{\otimes n/2+1} \otimes I_{n/2})$. The quantum distinguisher by Kuwakado and Morii works as follows.

1. Measure the first $n/2+1$ qubits of $\mathcal{S}_{f^{\mathcal{O}}} |0^{n+1}\rangle$ to obtain the vector $y \in \{0,1\}^{n/2+1}$.
2. Repeat Step 1 until we obtain $n/2$ linearly independent vectors. If obtained, compute s by solving the system of linear equations.
3. Choose $\beta \in \{0,1\}$ and $z \in \{0,1\}^{n/2}$ randomly, and compute $f^{\mathcal{O}}(\beta \parallel z)$ and $f^{\mathcal{O}}((\beta \parallel z) \oplus s)$. If $f^{\mathcal{O}}(\beta \parallel z) = f^{\mathcal{O}}((\beta \parallel z) \oplus s)$, then output "$\mathcal{O}$ is FP_3," otherwise output "$\mathcal{O}$ is Π."

If $\mathcal{O}$ is FP_3, we obtain the period s in Step 2 with a high probability and it passes the test in Step 3. On the other hand, according to [15], if $\mathcal{O}$ is Π, with a high probability, Simon's algorithm returns a random string s', and the probability that $f^{\mathcal{O}}(\beta \parallel z) = f^{\mathcal{O}}((\beta \parallel z) \oplus s')$ is about $2^{-n/2}$. Therefore, the distinguisher above returns a correct answer by making $O(n)$ quantum queries.

Remark 1. We need to truncate outputs of $\mathcal{O}$ for constructing the function $f^{\mathcal{O}}$, since we use only the lower $n/2$ bits of the output of $\mathcal{O}$. However, the oracle may return outputs of which the lower and upper parts are entangled, and it is not trivial to truncate such outputs without destroying the entanglement, as pointed out by Kaplan et al. [13]. To solve this problem, Hosoyamada and Sasaki showed how to simulate truncation of outputs of the oracles without destroying quantum entanglements [11], and the same technique can be used in our case.

3.2 Key Recovery Attacks against the Feistel-KF Construction

Next, we introduce the idea of the key recovery attacks against the Feistel-KF construction by Hosoyamada and Sasaki [11], and Dong and Wang [8]. They combined the quantum distinguisher against the 3-round Feistel cipher (see Sect. 3.1) with the Grover search. The attack is a quantum chosen-plaintext attack, and recovers the keys of the r-round Feistel cipher in time $\tilde{O}(2^{(r-3)n/4})$.

Attack Idea. Given the quantum encryption oracle of the r-round Feistel-KF construction, run the following procedures (on a quantum circuit).

1. Implement a quantum circuit which
   - takes the intermediate state value after the first $(r-3)$ rounds and the subkeys for the first $(r-3)$ rounds as input,
   - computes the plaintext by decrypting the first $(r-3)$ rounds,
   - makes a quantum query of the computed plaintext to the oracle,
   - and returns the oracle output.

   The input and output of this circuit correspond to those of the last 3 rounds. We denote this circuit by $\mathcal{E}$, which is depicted in Fig. 5.

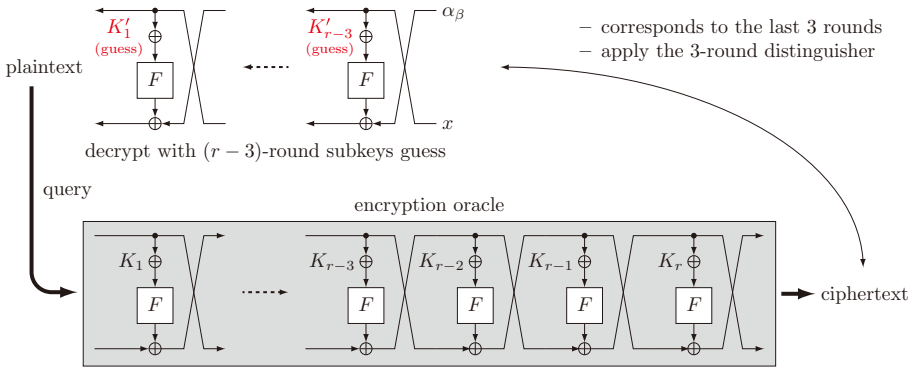

Fig. 5. Construction of $\mathcal{E}$ in the key recovery attack against the r-round Feistel-KF construction. The ciphertext corresponds to the output of the 3-round Feistel-KF construction which takes (K_{r-2}, K_{r-1}, K_r) as subkeys and (x, α_β) as input.

2. Guess the subkeys of the first $(r-3)$ rounds.
3. For each guess, check its correctness with the following procedure.
   (a) Apply the 3-round distinguisher to $\mathcal{E}$.
   (b) If the distinguisher returns that "this is a random permutation", then judge that the guess is wrong. Otherwise judge that the guess is correct.

Attack Complexity. The total length of the subkeys of the first $(r-3)$ rounds is $((r-3)n/2)$ bits. Thus the exhaustive search of the first $(r-3)$ rounds can be done in time $O(\sqrt{2^{(r-3)n/2}})$ by using the Grover search. Moreover, the 3-round distinguisher in the third step runs in time $O(n)$ for each subkeys guess. The running time of the attack is $O(\sqrt{2^{(r-3)n/2}}) \times O(\mathrm{poly}(n)) = \tilde{O}(2^{(r-3)n/4})$.

Although how to formally combine the Grover search and the 3-round distinguisher is non-trivial, the technique developed by Leander and May [17] guarantees that those can be combined. See the previous papers [11,8] for details.

4 Relaxing Simon's Algorithm

This section presents quantum distinguishers that are based on the relaxed version of Simon's algorithm [22]. In a nutshell, we discuss that it is enough to obtain several vectors that are orthogonal to the period, and thus we eliminate the need to recover the actual period. This is similar to the one by Santoli and Schaffner [20], while we are dealing with a general class of block ciphers, and we also formalize a sufficient condition so that the distinguisher works.

In more detail, instead of using the period for the basis of the distinguisher, we focus on the dimension of the space spanned by the vectors $y_1, y_2, \ldots$ that are obtained by using $\mathcal{S}_f$ (recall that $\mathcal{S}_f$ is defined in Sect. 2.2). If f has the non-zero period s, then the dimension is at most $|s|-1$, since the vectors $y_1, y_2, \ldots$ are all

orthogonal to the period s. On the other hand, as we prove in Theorem 2 below, if the function f does not have any period, the dimension of the space spanned by the vectors $y_1, y_2, \ldots$ can reach $|s|$ with a high probability. In other words, we can distinguish f by checking the dimension of the space spanned by the vectors $y_1, y_2, \ldots$ without computing the actual period s. Thus, there will not be a problem if there are several partial periods or periods other than s because our distinguisher does not need the period s.

Note that this technique works only if we do not need the value of s. This technique cannot be applied to the key recovery attacks on Even-Mansour construction and forgery attacks on authentication and authenticated encryption schemes since the goal of these attacks needs s [13].

Below we formally explain how our distinguisher works. Let $\mathcal{O} : \{0,1\}^n \to \{0,1\}^n$ be either an encryption scheme E_K or a random permutation $\Pi \xleftarrow{\$} \mathrm{Perm}(n)$, and suppose that the quantum oracles of $\mathcal{O}$ and $\mathcal{O}^{-1}$ are given. Our goal is to distinguish whether $\mathcal{O} = E_K$ or $\mathcal{O} = \Pi$. In what follows, when we use the symbol π for a permutation, we consider that π is a fixed (or constant) permutation.

Settings. Our distinguisher can be applied when there is a function family $\{f^\pi : \{0,1\}^\ell \to \{0,1\}^m\}_{\pi \in \mathrm{Perm}(n)}$ that satisfies the following conditions:

1. There is a (classical) algorithm $\mathcal{A}$ that makes black-box access to π, π^{-1}, and computes f^π. That is, for each permutation π, $\mathcal{A}^{\pi,\pi^{-1}}$ computes $f^\pi(x)$ if x is given as input. We assume that $\mathcal{A}$ makes $O(1)$ queries and runs in time $O(\mathrm{poly}(\ell, m))$.
2. For the encryption scheme E and any key K, f^{E_K} has a period, i.e., there exists $s \in \{0,1\}^\ell$ such that $f^{E_K}(x \oplus s) = f^{E_K}(x)$ holds for all x (note that s depends on K).

Moreover, informally we expect that f^Π has no period with a high probability when Π is a random permutation. Note that the first condition implies that we can make a quantum circuit that realizes the unitary operator $U_{f^\mathcal{O}} : |x\rangle |z\rangle \mapsto |x\rangle |z \oplus f^\mathcal{O}(x)\rangle$ by making $O(1)$ quantum queries to $\mathcal{O}$ and $\mathcal{O}^{-1}$, since any classical deterministic algorithm can be converted to a corresponding quantum algorithm.

Description of the Distinguisher. Let $\mathcal{S}_{f^\mathcal{O}}$ be the unitary operator that is defined as in Sect. 2. Recall that $\mathcal{S}_{f^\mathcal{O}} = (H^{\otimes \ell} \otimes I_m) \cdot U_{f^\mathcal{O}} \cdot (H^{\otimes \ell} \otimes I_m)$. Our distinguisher is described in Algorithm 1.

Analysis of the Distinguisher. Our distinguisher always returns the correct answer if $\mathcal{O} = E_K$, since by assumption, f^{E_K} has a period for any K, and thus the dimension of the space spanned by $\mathcal{Y}$ becomes strictly less than ℓ. Our distinguisher fails only if $\mathcal{O} = \Pi$ and the dimension of the space spanned by $\mathcal{Y}$ becomes less than ℓ. Below we analyze the failure probability, assuming that η (the number of iterations in Step 2) is sufficiently large.

Algorithm 1 Distinguisher without recovering the period

1. Prepare an empty set $\mathcal{Y}$.
2. For $1 \leq i \leq \eta$, do:
3. Measure the first ℓ qubits of $\mathcal{S}_{f^{\mathcal{O}}} |0^{\ell+m}\rangle$ and add the obtained vector y to $\mathcal{Y}$.
4. End For
5. Calculate the dimension d of the vector space spanned by $\mathcal{Y}$.
6. If $d = \ell$, then output "$\mathcal{O}$ is Π." If $d < \ell$, output "$\mathcal{O}$ is E_K."

The failure probability increases if the distribution of y in Step 3 is highly biased. Moreover, we obtain a vector y which is orthogonal to a partial period t of f^{Π} with a high probability in Step 3 if $\Pr_x \left[f^{\Pi}(x) = f^{\Pi}(x \oplus t) \right]$ is large (i.e., t is close to a complete period) by definition of $\mathcal{S}_{f^{\mathcal{O}}}$. To capture how much the distribution of y is biased under the condition that random permutation Π matches a fixed permutation π, we introduce a parameter ϵ_f^{π} defined as

$$\epsilon_f^{\pi} = \max_{t \in \{0,1\}^{\ell} \setminus \{0^{\ell}\}} \Pr_x \left[f^{\pi}(x) = f^{\pi}(x \oplus t) \right]. \tag{4}$$

We expect that, if π is chosen uniformly at random, this parameter ϵ_f^{π} is small on average.

Now take a small constant $0 \leq \delta < 1$ arbitrarily and say that a permutation π is irregular if $\epsilon_f^{\pi} > 1 - \delta$, i.e., ϵ_f^{π} is relatively large. In addition, define the set of irregular permutations irr_f^{δ} as

$$\mathsf{irr}_f^{\delta} = \{\pi \in \mathrm{Perm}(n) \mid \epsilon_f^{\pi} > 1 - \delta\}. \tag{5}$$

Our intuition is that the failure probability becomes small if $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{\delta}]$ is sufficiently small, and actually the following theorem holds.

Theorem 2. *Let ℓ and m be positive integers that are $O(n)$. Assume that we have a quantum circuit with $O(\mathrm{poly}(\ell, m))$ qubits which computes $f^{\mathcal{O}}$ by making $O(1)$ queries to $\mathcal{O}$, and runs in time $T = T(\ell, m)$. Then, our distinguisher makes $O(\eta)$ quantum queries, runs in time $O(\eta T + \ell^3)$, and distinguishes E_K from Π with probability at least*

$$1 - 2^{\ell}/e^{\delta\eta/2} - \Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{\delta}]. \tag{6}$$

A proof is presented in Appendix B. This theorem guarantees that we can distinguish E_K from Π if $2^{\ell}/e^{\delta\eta/2}$ and $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{\delta}]$ are small. In later sections, we apply the above theorem with $\eta = 2\ell/\delta$, in which case we have $2^{\ell}/e^{\delta\eta/2} = (2/e)^{\ell}$.

If we use the technique by Kaplan et al. (Theorem 1) to analyze a success probability of a distinguisher, we have to upper bound the parameter $\epsilon(f^{E_K}, s)$ that depends on the real construction E_K, which may become hard if E_K has a complex structure. On the other hand, our technique (Theorem 2) requires only upper bounds of the terms that are not related to the real construction. Thus our technique makes analysis of a distinguisher easier than the technique by Kaplan et al. We remark that the probability evaluation in the ideal case that is similar to the last term of equation (6) is needed in the previous works [15,13,7] as well.

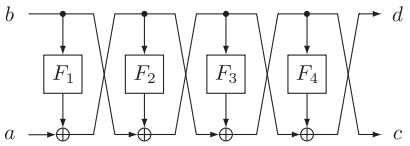

Fig. 6. The 4-round Feistel-F construction. $F_i \in \mathrm{Func}(n/2)$.

5 Quantum Distinguishing Attacks against Feistel-F

In this section, we present our distinguisher against the 4-round Feistel-F construction with quantum chosen-ciphertext attacks. Based on this, we present in Sect. 6 quantum distinguishing attacks and key recovery attacks against the Feistel-KF construction.

We write F_{K_i} as F_i. Note that F_i is still a keyed function and the absence of K_i does not imply that it is a keyless function. Let FF_4 denote the encryption algorithm of the 4-round Feistel-F construction, and FF_4^{-1} denote its decryption algorithm. Figure 6 illustrates FF_4. Let $F_1, \ldots, F_4 \in \mathrm{Func}(n/2)$ be the round functions of Feistel-F. FF_4 takes a plaintext $(a, b) \in (\{0,1\}^{n/2})^2$ as input and outputs a ciphertext $(c, d) \in (\{0,1\}^{n/2})^2$, where $\mathrm{FF}_4 : (a, b) \mapsto (c, d)$ is

$$\begin{aligned} c &= a \oplus F_1(b) \oplus F_3(b \oplus F_2(a \oplus F_1(b))), \\ d &= b \oplus F_2(a \oplus F_1(b)) \oplus F_4\Big(a \oplus F_1(b) \oplus F_3(b \oplus F_2(a \oplus F_1(b)))\Big). \end{aligned}$$

The decryption $\mathrm{FF}_4^{-1} : (c, d) \mapsto (a, b)$ is defined as

$$\begin{aligned} a &= c \oplus F_3(d \oplus F_4(c)) \oplus F_1\Big(d \oplus F_4(c) \oplus F_2(c \oplus F_3(d \oplus F_4(c)))\Big), \\ b &= d \oplus F_4(c) \oplus F_2(c \oplus F_3(d \oplus F_4(c))). \end{aligned}$$

Let $\Pi \xleftarrow{\$} \mathrm{Perm}(n)$ be a random permutation and Π^{-1} be the inverse permutation of Π. Π takes a plaintext $(a, b) \in (\{0,1\}^{n/2})^2$ as input and outputs a ciphertext $(c, d) \in (\{0,1\}^{n/2})^2$, and Π^{-1} takes a ciphertext (c, d) as input and outputs a plaintext (a, b).

Given the quantum oracles of $\mathcal{O}$ and $\mathcal{O}^{-1}$, where $\mathcal{O}$ is either the 4-round Feistel-F FF_4 or a random permutation $\Pi \xleftarrow{\$} \mathrm{Perm}(n)$, our goal is to distinguish the two cases. We now construct the function $f^{\mathcal{O}}$ to use Algorithm 1. We first fix two arbitrary distinct constants $\alpha_0, \alpha_1 \in \{0,1\}^{n/2}$, and we define the function $f^{\mathcal{O}}$ as

$$\begin{aligned} f^{\mathcal{O}} : \{0,1\} \times \{0,1\}^{n/2} &\to \{0,1\}^{n/2} \\ (\beta \,\|\, x) &\mapsto b \oplus \alpha_\beta, \quad \text{where } (c, d) = \mathcal{O}(x, \alpha_\beta), \\ &\qquad\qquad (a, b) = \mathcal{O}^{-1}(c, d \oplus \alpha_0 \oplus \alpha_1). \end{aligned}$$

That is, $f^{\mathcal{O}}$ is obtained by first encrypting (x, α_β) to obtain the ciphertext (c, d), then decrypting $(c, d \oplus \alpha_0 \oplus \alpha_1)$ to obtain the plaintext (a, b), and we define $f^{\mathcal{O}}$ as $b \oplus \alpha_\beta$.

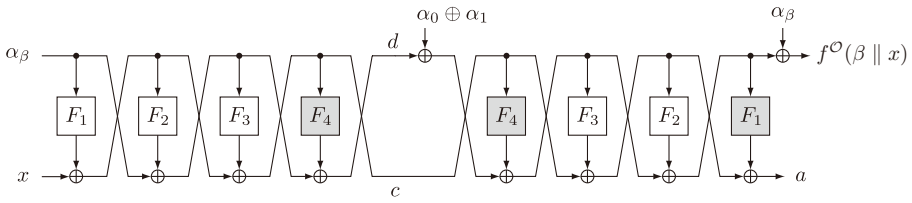

Fig. 7. The function $f^{\mathcal{O}}$ with FF_4 and FF_4^{-1}, where $\mathcal{O}$ is FF_4.

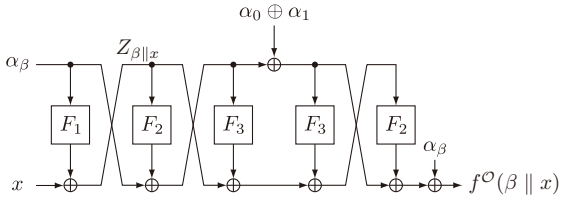

Fig. 8. A circuit that is equivalent to $f^{\mathcal{O}}$.

If $\mathcal{O}$ is FF_4, then by connecting FF_4 and FF_4^{-1}, our function $f^{\mathcal{O}}$ can be illustrated as in Fig. 7. We observe that F_4 has no effect on the computation of $f^{\mathcal{O}}$, and F_1 in FF_4^{-1} does not contribute to $f^{\mathcal{O}}$. They are shown in gray in Fig. 7. We see that Fig. 7 is equivalent to Fig. 8, and the function $f^{\mathcal{O}}$ is described as

$$\begin{aligned} f^{\mathcal{O}}(\beta \parallel x) = \alpha_0 \oplus \alpha_1 \oplus F_2(x \oplus F_1(\alpha_\beta)) \\ \oplus F_2\Big(x \oplus F_1(\alpha_\beta) \oplus F_3(\alpha_\beta \oplus F_2(x \oplus F_1(\alpha_\beta))) \\ \oplus F_3(\alpha_\beta \oplus \alpha_0 \oplus \alpha_1 \oplus F_2(x \oplus F_1(\alpha_\beta)))\Big). \end{aligned} \tag{7}$$

Our main observation is the following lemma.

Lemma 2. *If $\mathcal{O} = \mathrm{FF}_4$, $f^{\mathcal{O}}$ satisfies $f^{\mathcal{O}}(\beta \parallel x) = f^{\mathcal{O}}(\beta \oplus 1 \parallel x \oplus F_1(\alpha_0) \oplus F_1(\alpha_1))$. That is, $f^{\mathcal{O}}$ has the period $s = 1 \parallel F_1(\alpha_0) \oplus F_1(\alpha_1)$.*

Proof. Let $Z_{\beta\|x} = x \oplus F_1(\alpha_\beta)$ (See Fig. 8). We prove the lemma based on two claims. The first claim is that $Z_{\beta\|x}$ already has the period $s = 1 \parallel F_1(\alpha_0) \oplus F_1(\alpha_1)$, and the second claim is that the subsequent computation of $f^{\mathcal{O}}$ does not depend on β nor x.

First, $Z_{\beta\|x}$ has the period s, since

$$\begin{aligned} Z_{(\beta\|x)\oplus s} &= x \oplus F_1(\alpha_0) \oplus F_1(\alpha_1) \oplus F_1(\alpha_{\beta\oplus 1}) \\ &= x \oplus F_1(\alpha_\beta) \\ &= Z_{\beta\|x}. \end{aligned}$$

We next show that the subsequent computation of $f^{\mathcal{O}}$ does not depend on β nor x. If we describe $f^{\mathcal{O}}$ in equation (7) by using $Z_{\beta\|x}$, then we obtain

$$f^{\mathcal{O}}(\beta \parallel x) = \alpha_0 \oplus \alpha_1 \oplus F_2(Z_{\beta\|x})$$

$$\oplus F_2\Big(Z_{\beta\|x} \oplus F_3(\alpha_\beta \oplus F_2(Z_{\beta\|x})) \oplus F_3(\alpha_\beta \oplus \alpha_0 \oplus \alpha_1 \oplus F_2(Z_{\beta\|x}))\Big).$$

Now this is equivalent to

$$\begin{aligned} f^{\mathcal{O}}(\beta \,\|\, x) &= \alpha_0 \oplus \alpha_1 \oplus F_2(Z_{\beta\|x}) \\ &\quad \oplus F_2\Big(Z_{\beta\|x} \oplus F_3(\alpha_0 \oplus F_2(Z_{\beta\|x})) \oplus F_3(\alpha_1 \oplus F_2(Z_{\beta\|x}))\Big) \end{aligned} \tag{8}$$

since $\{\alpha_\beta, \alpha_\beta \oplus \alpha_0 \oplus \alpha_1\} = \{\alpha_0, \alpha_1\}$. We see that $f^{\mathcal{O}}$ depends on $Z_{\beta\|x}$ that has the period $s = 1 \,\|\, F_1(\alpha_0) \oplus F_1(\alpha_1)$, and hence the lemma follows. □

Therefore, we can construct a distinguisher against the 4-round Feistel-F construction by using the function $f^{\mathcal{O}}$. From Theorem 2, the success probability of the distinguisher with measuring $(2n+4)$ times is $1-(2/e)^{n/2+1} - \Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$, where we use $\delta = 1/2$ and $\eta = 2n+4$.

It is clear that $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$ is a small value, since it is highly unlikely that $f^{\mathcal{O}}$ obtained from a random permutation has periods. In Appendix C, we present experimental results for small values of n to show that $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$ is indeed a small value.

6 Quantum Attacks against Feistel-KF

The distinguisher in the previous section can obviously be applied to the 4-round Feistel-KF construction, and we can distinguish it from random permutations in polynomial time. Similarly to the previous key recovery attacks against the Feistel-KF [11,8] construction (see Sect. 3.2), our 4-round distinguisher can be combined with the Grover search to develop key recovery attacks. Our new key recovery attack recovers the keys of the r-round Feistel-KF construction in time $\tilde{O}(2^{(r-4)n/4})$ in the quantum CCA setting.

Attack Idea. Our attack idea is almost the same as that of the previous attacks [11,8], except that our attack uses not only the encryption oracle but also the decryption oracle. Given the quantum encryption and decryption oracles of the r-round Feistel-KF construction, run the following procedures (on a quantum circuit).

1. Implement a quantum circuit $\mathcal{E}$ that takes the intermediate state value after the first $(r-4)$ rounds and the subkeys for the first $(r-4)$ rounds as input, and computes the last 4 rounds, in the same way as the first step of the attack idea in Sect. 3.2.
2. Implement a quantum circuit $\mathcal{D}$ that computes the inverse of $\mathcal{E}$. That is, implement a quantum circuit which
   - takes the ciphertext and the subkeys for the first $(r-4)$ rounds as input,
   - makes a quantum decryption query of the ciphertext to the oracle to obtain the plaintext,

- computes the intermediate state value after the first $(r-4)$ rounds from the plaintext and the subkeys for the first $(r-4)$ rounds,
- and returns the intermediate state.

3. Guess the subkeys of the first $(r-4)$ rounds.
4. For each guess, check its correctness with the following procedure.
   (a) Apply the 4-round distinguisher to $\mathcal{E}$ and $\mathcal{D}$.
   (b) If the distinguisher returns that "this is a random permutation", then judge that the guess is wrong. Otherwise judge that the guess is correct.

Attack Complexity. The length of the first $(r-4)$-round subkeys is $((r-4)n/2)$ bits in total. Thus the exhaustive search on the first $(r-4)$ rounds can be done in time $O(\sqrt{2^{(r-4)n/2}})$ by using the Grover search. Moreover, the 4-round distinguisher in the fourth step runs in time $O(n)$ for each candidate subkeys. Therefore the running time of the attack becomes $O(\sqrt{2^{(r-4)n/2}}) \times O(\mathrm{poly}(n)) = \tilde{O}(2^{(r-4)n/4})$.

Our new attack reduces the time complexity $\tilde{O}(2^{(r-3)n/4})$ of the previous attacks to $\tilde{O}(2^{(r-4)n/4})$, by using our new CCA 4-round distinguisher instead of the previous CPA 3-round distinguisher by Kuwakado and Morii. Our attack is a chosen-ciphertext attack unlike that the previous attacks are chosen-plaintext attacks, since our 4-round distinguisher is a CCA distinguisher.

7 Quantum Attacks against Feistel-FK

In Sect. 7.1, we show a quantum distinguishing attack against Feistel-FK. Based on this, we present in Sect. 7.2 a key recovery attack. The main difference from the previous sections is that the number of the distinguishable rounds increases. In Sect. 7.3, we present a quantum chosen-plaintext attack.

7.1 Distinguishers against Feistel-FK

We present our distinguisher against the 6-round Feistel-FK construction with quantum chosen-ciphertext attacks. This attack is based on the distinguisher against the 4-round Feistel-F construction described in Sect. 5. We increase the number of rounds by adding the first and last rounds, and this is possible because we can compute the output of the first F function and the last F function in encryption (or decryption) without knowing the subkeys.

Let $(a,b) \in (\{1,0\}^{n/2})^2$ denote a plaintext and $(c,d) \in (\{1,0\}^{n/2})^2$ denote a ciphertext. Let $\mathrm{FFK}_6 : (a,b) \mapsto (c,d)$ denote the encryption algorithm of the 6-round Feistel-FK construction, and $\mathrm{FFK}_6^{-1} : (c,d) \mapsto (a,b)$ denote its decryption algorithm. Figure 9 illustrates the 6-round Feistel-FK construction.

Given the quantum oracles of $\mathcal{O}$ and $\mathcal{O}^{-1}$, we define the function $f^{\mathcal{O}}$ as

$$\begin{aligned} f^{\mathcal{O}} : \{0,1\} \times \{0,1\}^{n/2} &\to \{0,1\}^{n/2} \\ (\beta \parallel x) &\mapsto a \oplus F(b) \oplus \alpha_\beta \\ \text{where } (c,d) &= \mathcal{O}(\alpha_\beta \oplus F(x), x), \end{aligned}$$

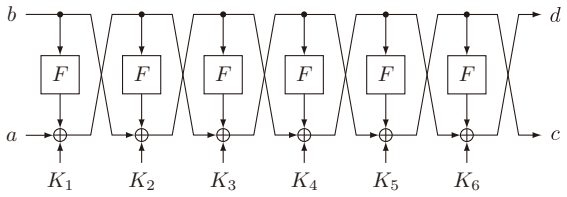

Fig. 9. The 6-round Feistel-FK construction.

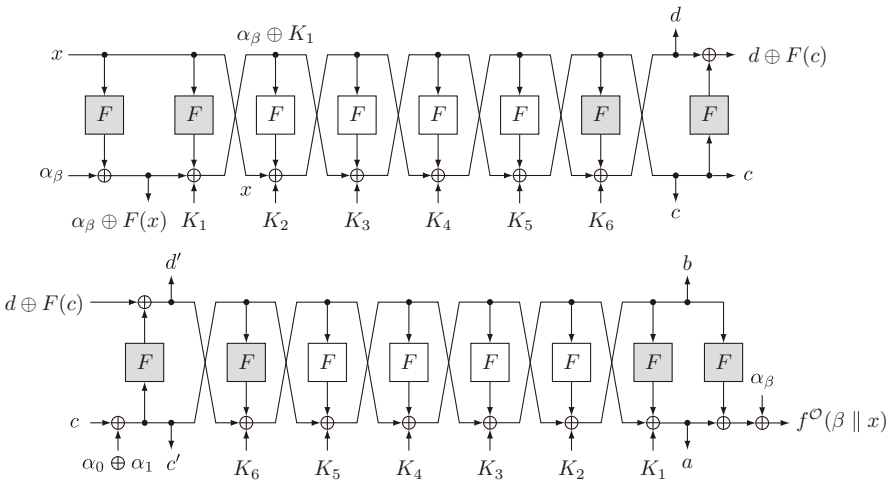

Fig. 10. The function $f^{\mathcal{O}}$ with FFK_6 and FFK_6^{-1}, where $\mathcal{O}$ is FFK_6. $c' = c \oplus \alpha_0 \oplus \alpha_1$ and $d' = d \oplus F(c) \oplus F(c \oplus \alpha_0 \oplus \alpha_1)$.

$$(a, b) = \mathcal{O}^{-1}(c \oplus \alpha_0 \oplus \alpha_1, d \oplus F(c) \oplus F(c \oplus \alpha_0 \oplus \alpha_1)).$$

If $\mathcal{O}$ is FFK_6, then our function $f^{\mathcal{O}}$ can be illustrated as in Fig. 10. We observe that the F functions shown in gray in Fig. 10 and the subkeys K_6 have no effect on the computation of $f^{\mathcal{O}}$. By connecting FFK_6 and FFK_6^{-1}, we obtain Fig. 11 that is equivalent to Fig. 10. If we replace α_β with $\alpha_\beta \oplus K_1$ and $F_i(x)$ with $F(x) \oplus K_{i+1}$ in Fig. 7, we see that Fig. 7 is equivalent to Fig. 11. Therefore, from equation (7) and equation (8), the function $f^{\mathcal{O}}$ is described as

$$\begin{aligned} &f^{\mathcal{O}}(\beta \parallel x) \\ &\quad = \alpha_0 \oplus \alpha_1 \oplus F(x \oplus F(\alpha_\beta \oplus K_1) \oplus K_2) \\ &\qquad \oplus F\Big(x \oplus F(\alpha_\beta \oplus K_1) \oplus K_2 \oplus F\big(\alpha_0 \oplus F(x \oplus F(\alpha_\beta \oplus K_1) \oplus K_2) \oplus K_3\big) \\ &\qquad\quad \oplus F\big(\alpha_1 \oplus F(x \oplus F(\alpha_\beta \oplus K_1) \oplus K_2) \oplus K_3\big)\Big) \end{aligned}$$

and it has the period $s = 1 \parallel F(\alpha_0 \oplus K_1) \oplus F(\alpha_1 \oplus K_1)$.

Therefore, we can construct a distinguisher against the 6-round Feistel-FK construction by using the function $f^{\mathcal{O}}$. From Theorem 2, the success probability

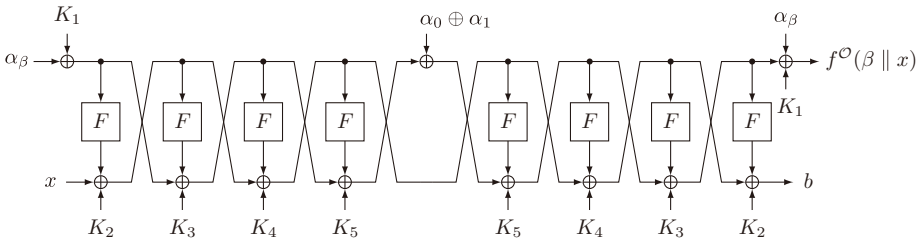

Fig. 11. A circuit that is equivalent to Fig. 10.

of the distinguisher with measuring $(2n+4)$ times is $1-(2/e)^{n/2+1}-\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$, where we set $\delta = 1/2$ and $\eta = 2n+4$. Note that $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$ is a small value, as it is unlikely that $f^{\mathcal{O}}$ obtained from a random permutation has periods.

7.2 Key Recovery Attacks against Feistel-FK

Similarly to the key recovery attacks against the Feistel-KF construction in Sect. 6, the distinguisher introduced above can be combined with the Grover search to develop key recovery attacks. We can recover keys of the r-round Feistel-FK construction in time $\tilde{O}(2^{(r-6)n/4})$ in the quantum CCA setting.

Our attack idea follows the attack against the Feistel-KF construction in Sect. 6. Recall that the attack in Sect. 6 guesses the first $(r-4)$-round subkeys since a 4-round distinguisher is available. On the other hand, as for the Feistel-FK construction, we can use the 6-round distinguisher in Sect. 7.1 instead of the 4-round distinguisher. Hence it is sufficient to guess only the first $(r-6)$-round subkeys (instead of the first $(r-4)$-round subkeys) when we attack the Feistel-FK construction. The time complexity of our attack becomes $\tilde{O}(2^{(r-6)n/4})$, since the Grover search on the first $(r-6)$-round subkeys ($\frac{(r-6)n}{2}$ bits in total) requires $O(\sqrt{2^{(r-6)n/2}}) = O(2^{(r-6)n/4})$ evaluations.

7.3 Quantum CPA Attacks against Feistel-FK

We can also construct a distinguisher and recover the key of the Feistel-FK construction in the quantum CPA setting. As in Sect. 7.1, we can construct a 5-round distinguisher by following the 3-round distinguisher in Sect. 3.1 and by computing the output of the first F function and the last F function in encryption. Specifically, we use $(\alpha_\beta \oplus F(x), x)$ as the input of the oracle $\mathcal{O}$ and use $d \oplus F(c) \oplus \alpha_\beta$ as the output of the function $f^{\mathcal{O}}(\beta \parallel x)$, where $(c,d) = \mathcal{O}(\alpha_\beta \oplus F(x), x)$. This function has the period $s = 1 \parallel F(\alpha_0 \oplus K_1) \oplus F(\alpha_1 \oplus K_1)$.

Combined with the 5-round distinguisher, we can recover the subkeys of the r-round Feistel-FK construction as in Sect. 6. The time complexity of our key recovery attack is $\tilde{O}(2^{(r-5)n/4})$, since the Grover search on the first $(r-5)$-round subkeys ($\frac{(r-5)n}{2}$ bits in total) requires $O(\sqrt{2^{(r-5)n/2}}) = O(2^{(r-5)n/4})$ evaluations.

8 Concluding Remarks

In this paper, we first formalized a distinguishing algorithm against block ciphers that does not recover the period. We then considered quantum chosen-ciphertext attacks against Feistel ciphers. We gave a new quantum CCA distinguisher against Feistel ciphers that can distinguish more rounds than the previous CPA distinguishers. Our quantum CCA distinguishers can distinguish the 4-round Feistel-F and Feistel-KF constructions, and the 6-round Feistel-FK construction, from random permutations in polynomial-time of the output size. Moreover, we extended the distinguishers to key recovery attacks for the Feistel-KF and Feistel-FK constructions. Our quantum CCA key recovery attacks against the r-round Feistel-KF and Feistel-FK constructions recover keys in time $\tilde{O}(2^{(r-4)n/4})$ and $\tilde{O}(2^{(r-6)n/4})$, and quantum CPA key recovery attacks against the r-round Feistel-FK constructions recover keys in time $\tilde{O}(2^{(r-5)n/4})$, respectively.

There are interesting open questions. First, we still do not know the tight bound on the number of rounds that we can distinguish the Feistel-F construction. From the result of Kuwakado and Morii, we know that the 3-round construction can be distinguished with quantum CPA, and this paper shows that the 4-round construction can be distinguished with quantum CCA. However, there is a possibility that these rounds can be extended, and deriving the tight number of rounds remains as a challenging question. Improving the complexity or extending the number of rounds of the attacks against Feistel-KF and Feistel-FK constructions is also an interesting question.

Acknowledgments. The authors would like to thank participants of Dagstuhl seminar 18021, Symmetric Cryptography, for insightful feedback. We also would like to thank the anonymous reviewers of CT-RSA 2019 for helpful comments.

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 platforms - design and analysis. In: Stinson, D.R., Tavares, S.E. (eds.) SAC 2000. LNCS, vol. 2012, pp. 39–56. Springer (2000), https://doi.org/10.1007/3-540-44983-3_4
2. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK lightweight block ciphers. In: Proceedings of the 52nd Annual Design Automation Conference. pp. 175:1–175:6. ACM (2015), http://doi.acm.org/10.1145/2744769.2747946
3. Bonnetain, X.: Quantum Key-Recovery on Full AEZ. In: Adams, C., Camenisch, J. (eds.) SAC 2017. LNCS, vol. 10719, pp. 394–406. Springer (2017), https://doi.org/10.1007/978-3-319-72565-9_20
4. Bonnetain, X., Naya-Plasencia, M., Schrottenloher, A.: On Quantum Slide Attacks. IACR Cryptology ePrint Archive **2018**, 1067 (2018), https://eprint.iacr.org/2018/1067

5. Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: New attacks on Feistel structures with improved memory complexities. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015, Part I. LNCS, vol. 9215, pp. 433–454. Springer (2015), https://doi.org/10.1007/978-3-662-47989-6_21
6. Dong, X., Dong, B., Wang, X.: Quantum attacks on some Feistel block ciphers. IACR Cryptology ePrint Archive **2018**, 504 (2018), https://eprint.iacr.org/2018/504
7. Dong, X., Li, Z., Wang, X.: Quantum cryptanalysis on some generalized Feistel schemes. IACR Cryptology ePrint Archive **2017**, 1249 (2017), http://eprint.iacr.org/2017/1249
8. Dong, X., Wang, X.: Quantum key-recovery attack on Feistel structures. IACR Cryptology ePrint Archive **2017**, 1199 (2017), http://eprint.iacr.org/2017/1199
9. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Miller, G.L. (ed.) STOC 1996. pp. 212–219. ACM (1996), http://doi.acm.org/10.1145/237814.237866
10. Guo, J., Jean, J., Nikolic, I., Sasaki, Y.: Meet-in-the-middle attacks on generic Feistel constructions. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014, Part I. LNCS, vol. 8873, pp. 458–477. Springer (2014), https://doi.org/10.1007/978-3-662-45611-8_24
11. Hosoyamada, A., Sasaki, Y.: Quantum Demiric-Selçuk meet-in-the-middle attacks: Applications to 6-round generic Feistel constructions. In: Catalano, D., Prisco, R.D. (eds.) SCN 2018. LNCS, vol. 11035, pp. 386–403. Springer (2018), https://doi.org/10.1007/978-3-319-98113-0_21
12. Isobe, T., Shibutani, K.: Generic key recovery attack on Feistel scheme. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 464–485. Springer (2013), https://doi.org/10.1007/978-3-642-42033-7_24
13. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Breaking symmetric cryptosystems using quantum period finding. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part II. LNCS, vol. 9815, pp. 207–237. Springer (2016), https://doi.org/10.1007/978-3-662-53008-5_8
14. Knudsen, L.R.: The security of Feistel ciphers with six rounds or less. J. Cryptology **15**(3), 207–222 (2002), https://doi.org/10.1007/s00145-002-9839-y
15. Kuwakado, H., Morii, M.: Quantum distinguisher between the 3-round Feistel cipher and the random permutation. In: ISIT 2010. pp. 2682–2685. IEEE (2010), https://doi.org/10.1109/ISIT.2010.5513654
16. Kuwakado, H., Morii, M.: Security on the quantum-type Even-Mansour cipher. In: ISITA 2012. pp. 312–316. IEEE (2012), http://ieeexplore.ieee.org/document/6400943/
17. Leander, G., May, A.: Grover meets Simon - Quantumly attacking the FX-construction. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part II. LNCS, vol. 10625, pp. 161–178. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_6
18. Luby, M., Rackoff, C.: How to construct pseudorandom permutations from pseudorandom functions. SIAM J. Comput. **17**(2), 373–386 (1988), https://doi.org/10.1137/0217022
19. National Bureau of Standards: Data encryption standard. FIPS 46 (January 1977)
20. Santoli, T., Schaffner, C.: Using Simon's algorithm to attack symmetric-key cryptographic primitives. Quantum Information & Computation **17**(1&2), 65–78 (2017), http://www.rintonpress.com/xxqic17/qic-17-12/0065-0078.pdf

21. Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T., Shirai, T.: Piccolo: An ultra-lightweight blockcipher. In: Preneel, B., Takagi, T. (eds.) CHES 2011. LNCS, vol. 6917, pp. 342–357. Springer (2011), https://doi.org/10.1007/978-3-642-23951-9_23
22. Simon, D.R.: On the power of quantum computation. SIAM J. Comput. **26**(5), 1474–1483 (1997), https://doi.org/10.1137/S0097539796298637
23. Todo, Y.: Structural evaluation by generalized integral property. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 287–314. Springer (2015), https://doi.org/10.1007/978-3-662-46800-5_12
24. Yang, G., Zhu, B., Suder, V., Aagaard, M.D., Gong, G.: The Simeck family of lightweight block ciphers. In: Güneysu, T., Handschuh, H. (eds.) CHES 2015. LNCS, vol. 9293, pp. 307–329. Springer (2015), https://doi.org/10.1007/978-3-662-48324-4_16

A Proof of Lemma 1

Proof. Suppose that $\beta' = \beta$. Then we have

$$f^{\mathcal{O}}(\beta \parallel x) = f^{\mathcal{O}}(\beta' \parallel x') \Leftrightarrow P_2(x \oplus P_1(\alpha_\beta)) = P_2(x' \oplus P_1(\alpha_\beta)) \Leftrightarrow x = x'$$

since P_2 is a permutation. That is, in this case, we have $\beta \parallel x = \beta' \parallel x'$.

Next, suppose that $\beta' = \beta \oplus 1$. We have

$$\begin{aligned} f^{\mathcal{O}}(\beta \parallel x) = f^{\mathcal{O}}(\beta' \parallel x') &\Leftrightarrow P_2(x \oplus P_1(\alpha_\beta)) = P_2(x' \oplus P_1(\alpha_{\beta\oplus 1})) \\ &\Leftrightarrow x' = x \oplus P_1(\alpha_\beta) \oplus P_1(\alpha_{\beta\oplus 1}) \\ &\Leftrightarrow x' = x \oplus P_1(\alpha_0) \oplus P_1(\alpha_1), \end{aligned}$$

where the last equivalence follows from $\{\alpha_\beta, \alpha_\beta \oplus \alpha_0 \oplus \alpha_1\} = \{\alpha_0, \alpha_1\}$. That is, $\beta' \parallel x' = (\beta \parallel x) \oplus s$.

The other direction easily follows as we have

$$\begin{aligned} f^{\mathcal{O}}((\beta \parallel x) \oplus (1 \parallel P_1(\alpha_0) \oplus P_1(\alpha_1))) &= P_2(x \oplus P_1(\alpha_0) \oplus P_1(\alpha_1) \oplus P_1(\alpha_{\beta\oplus 1})) \\ &= P_2(x \oplus P_1(\alpha_\beta)) \\ &= f^{\mathcal{O}}(\beta \parallel x). \end{aligned}$$

Therefore, the lemma follows. □

B Proof of Theorem 2

Proof. Remember that our distinguisher fails if and only if $\mathcal{O} = \Pi$ and the dimension of the space spanned by $\mathcal{Y}$ is less than ℓ. Thus the failure probability equals to

$$\Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y_i, w_i) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) \text{ for } 1 \le i \le \eta : \\ \dim(\mathrm{Span}(y_1, \ldots, y_\eta)) \le \ell - 1\Big]$$

$$
\begin{aligned}
&= \Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y_i, w_i) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) \text{ for } 1 \le i \le \eta : \\
&\qquad \dim(\mathrm{Span}(y_1, \ldots, y_\eta)) \le \ell - 1 \wedge \Pi \notin \mathsf{irr}_f^\delta\Big] \\
&\quad + \Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y_i, w_i) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) \text{ for } 1 \le i \le \eta : \\
&\qquad \dim(\mathrm{Span}(y_1, \ldots, y_\eta)) \le \ell - 1 \wedge \Pi \in \mathsf{irr}_f^\delta\Big] \\
&\le \Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y_i, w_i) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) \text{ for } 1 \le i \le \eta : \\
&\qquad \dim(\mathrm{Span}(y_1, \ldots, y_\eta)) \le \ell - 1 \Big| \Pi \notin \mathsf{irr}_f^\delta\Big] \\
&\quad + \Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n) : \Pi \in \mathsf{irr}_f^\delta\Big] \qquad (9)
\end{aligned}
$$

The first term of the right hand side of equation (9) is upper bounded as

$$
\begin{aligned}
&\Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y_i, w_i) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) \text{ for } 1 \le i \le \eta : \\
&\qquad \dim(\mathrm{Span}(y_1, \ldots, y_\eta)) \le \ell - 1 \Big| \Pi \notin \mathsf{irr}_f^\delta\Big] \\
&\le \Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y_i, w_i) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) \text{ for } 1 \le i \le \eta : \\
&\qquad \exists t \in \{0,1\}^\ell \backslash \{0^\ell\} \text{ s.t. } t \perp y_1 \wedge \cdots \wedge t \perp y_\eta \Big| \Pi \notin \mathsf{irr}_f^\delta\Big] \\
&\le \sum_{t \in \{0,1\}^\ell \backslash \{0^\ell\}} \Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y_i, w_i) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) \text{ for } 1 \le i \le \eta : \\
&\qquad\qquad t \perp y_1 \wedge \cdots \wedge t \perp y_\eta \Big| \Pi \notin \mathsf{irr}_f^\delta\Big] \\
&\le \sum_{t \in \{0,1\}^\ell \backslash \{0^\ell\}} \Big(\Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y, w) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) : \\
&\qquad\qquad t \perp y \Big| \Pi \notin \mathsf{irr}_f^\delta\Big]\Big)^\eta \\
&\le 2^\ell \cdot \max_{t \in \{0,1\}^\ell \backslash \{0^\ell\}} \Big(\Pr\Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y, w) \leftarrow (\text{measure } \mathcal{S}_{f^\Pi} |0^{\ell+m}\rangle) : \\
&\qquad\qquad t \perp y \Big| \Pi \notin \mathsf{irr}_f^\delta\Big]\Big)^\eta .
\end{aligned}
$$

Now we introduce the following lemma, which is proven as a subordinate result in the proof of Theorem 1 in [13].

Lemma 3 (Kaplan et al. [13]). *Let $f : \{0,1\}^\ell \to \{0,1\}^m$ be a fixed function, and $U_f : |x\rangle |y\rangle \mapsto |x\rangle |y \oplus f(x)\rangle$ be the corresponding unitary operator. Then,*

$$
\begin{aligned}
&\Pr\left[(x, y) \leftarrow (\text{measure } \mathcal{S}_f |0^{\ell+m}\rangle) : x \perp t\right] \\
&\quad = \frac{1}{2}\left(1 + \Pr\left[x \xleftarrow{\$} \{0,1\}^\ell : f(x \oplus t) = f(x)\right]\right)
\end{aligned}
$$

holds for any $t \in \{0,1\}^\ell$.

From the above lemma it follows that

$$2^{\ell} \cdot \max_{t \in \{0,1\}^{\ell} \setminus \{0^{\ell}\}} \Big(\Pr \Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), (y, w) \leftarrow (\text{measure } \mathcal{S}_{f^{\Pi}} |0^{\ell+m}\rangle) : t \perp y \Big| \Pi \notin \mathsf{irr}_f^{\delta} \Big] \Big)^{\eta}$$

$$= 2^{\ell} \cdot \max_{t \in \{0,1\}^{\ell} \setminus \{0^{\ell}\}} \left(\frac{1}{2} \Big(1 + \Pr \Big[\Pi \xleftarrow{\$} \mathrm{Perm}(n), x \xleftarrow{\$} \{0,1\}^{\ell} : f^{\Pi}(x \oplus t) = f^{\Pi}(x) \Big| \Pi \notin \mathsf{irr}_f^{\delta} \Big] \Big) \right)^{\eta}. \quad (10)$$

In addition, by assumption, for any fixed permutation $\pi \notin \mathsf{irr}_f^{\delta}$ it holds that

$$\Pr \Big[x \xleftarrow{\$} \{0,1\}^{\ell} : f^{\pi}(x \oplus t) = f^{\pi}(x) \Big] = \epsilon_f^{\pi} \leq 1 - \delta.$$

Hence the right hand side of equation (10) is upper bounded by $2^{\ell} \cdot (\frac{1}{2}(1 + (1 - \delta)))^{\eta} = 2^{\ell-\eta} (2 - \delta)^{\eta}$. Moreover, from the fact that $(1 - x)^{-1/x} \geq e$ holds for $0 < x < 1$, we have

$$2^{\ell-\eta} (2 - \delta)^{\eta} \leq 2^{\ell} \left((1 - \delta/2)^{-2/\delta} \right)^{\delta\eta/2} \leq 2^{\ell} \cdot e^{-\delta\eta/2}. \quad (11)$$

Therefore, from equations (9) and (11), the failure probability of our algorithm is upper bounded by

$$\frac{2^{\ell}}{e^{\delta\eta/2}} + \Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{\delta}], \quad (12)$$

which completes the proof. □

C Experimental Results on $\mathbf{Pr}_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$

We present our experimental results about $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$ where we use f that is described in Sect. 5. For $n = 4, 6, 8$, we are interested in the number of permutations that belong to $\mathsf{irr}_f^{1/2}$. For $n = 4$, there are $(2^4)! \approx 2^{44.25}$ permutations, and it is possible to precisely derive the value of $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$. For $n \geq 6$, it is not possible to exhaustively evaluate all the permutations. We thus generated permutations randomly and see if they belong to $\mathsf{irr}_f^{1/2}$ or not. Our results are represented in Table 2 and Table 3. The exponent of the ratio is rounded off to the second decimal place. We see that $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$ is already sufficiently small for $n = 4$, and as n becomes larger, the ratio becomes even smaller. Note that for instance $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}] = 0.8$ is sufficiently small for our distinguishing attack to practically succeed. In fact, in this case, the success probability of our distinguisher is at least $0.2 - (2/e)^{n/2+1}$ with $\delta = 1/2$ and $\eta = 2n + 4$, which is sufficiently large.

Table 2. The number of permutations $\pi \in \mathrm{Perm}(4)$ and the corresponding value of $\epsilon_f^{\mathcal{O}}$. $\Pr_{\Pi}[\Pi \in \mathsf{irr}_f^{1/2}]$ is about $2^{-1.49}$. Note that when $n = 4$, the number of input $\beta \parallel x$ of f is $2^3 = 8$.

number of permutations	value of $\epsilon_f^{\mathcal{O}}$
1,141,521,776,640	1/4
12,311,490,723,840	1/2
3,947,762,810,880	3/4
3,522,014,576,640	1

Table 3. The ratio of the number of permutations $\pi \in \mathrm{Perm}(6)$ and $\pi \in \mathrm{Perm}(8)$ that belong to $\mathsf{irr}_f^{1/2}$ with f described in Sect. 5. The left column shows the total number of permutations we generated randomly. The middle and right columns show the ratio of the number of permutations that belong to $\mathsf{irr}_f^{1/2}$ to the total number of permutations we generated.

number of permutations	ratio of $\mathsf{irr}_f^{1/2}$ $(n = 6)$	ratio of $\mathsf{irr}_f^{1/2}$ $(n = 8)$
10,000,000	$2^{-4.97}$	$2^{-17.30}$
20,000,000	$2^{-4.97}$	$2^{-17.18}$
30,000,000	$2^{-4.97}$	$2^{-17.22}$
40,000,000	$2^{-4.97}$	$2^{-17.26}$
50,000,000	$2^{-4.97}$	$2^{-17.24}$
60,000,000	$2^{-4.97}$	$2^{-17.24}$
70,000,000	$2^{-4.97}$	$2^{-17.25}$
80,000,000	$2^{-4.97}$	$2^{-17.26}$
90,000,000	$2^{-4.97}$	$2^{-17.28}$
100,000,000	$2^{-4.97}$	$2^{-17.30}$
200,000,000	$2^{-4.97}$	$2^{-17.34}$