Classical and Quantum Full Plaintext Recovery for Low-Round Feistel-Type Designs ⋆

Tingting Guo[1], Peng Wang[2(✉)], Jiwu Jing[2], Shuping Mao[3], and Gang Liu[4]

[1] Ningbo Key Laboratory of Information Technology Application Innovation and Security, School of Cyber Science and Engineering, Ningbo University of Technology, Ningbo, China
guotingting@nbut.edu.cn
[2] School of Cryptology, University of Chinese Academy of Sciences, Beijing, China
p-wang@ucas.ac.cn, jwjing@ucas.ac.cn
[3] Beijing Electronic Science and Technology Institute, Beijing, China
maoshuping19@mails.ucas.ac.cn
[4] National Key Laboratory of Security Communication, Chengdu, China
liugang@hnu.edu.cn

Abstract. The Feistel (Luby-Rackoff) structure underlies numerous block-cipher and mode-of-operation designs, whose security is traditionally assessed via indistinguishability. For low-round Feistel constructions, a variety of classical and quantum distinguishing attacks are known. In this work, we show that such distinguishing attacks can be systematically upgraded to *full plaintext recovery* with essentially the same query complexity. We establish classical recovery attacks on the 2-round Feistel under CPA and the 3-round Feistel under CCA using only three queries, and introduce quantum-assisted forward/backward extension techniques based on Simons algorithm that yield recovery attacks on the 3-round Feistel under qCPA and the 4-round Feistel under qCCA. We further prove that the attacks extend to the Unified Feistel-Lai-Massey (UFLM) framework and therefore apply to a broad class of two-branch constructions. As a consequence, we obtain plaintext-recovery attacks on 4/5/6-round Feistel-FK and on several practical enciphering schemes, including AEZ-core, FMix, OleF, double-decker, and docked-double-decker. Overall, our results reveal a fundamental connection between distinguishing and full plaintext recovery in low-round two-branch Feistel-type designs, in both classical and quantum settings.

Keywords: Feistel · Luby-Rackoff · Lai-Massey · UFLM · Full plaintext recovery attack · Truncated boomerang differential.

1 Introduction

Feistel-type designs and the usual security yardstick. Two-branch Feistel-type structures, including the classical Luby–Rackoff (Feistel) network and its

⋆ A preliminary version of this paper appears in CRYPTO 2026. This is the full version.
©IACR 2026.

variants, form a core design paradigm in block ciphers and symmetric modes. A Feistel network iterates a simple round transformation on two n-bit branches, yielding a $2n$-bit permutation while keeping encryption and decryption structurally symmetric. The celebrated theorem of Luby and Rackoff [23,24] shows that 3 rounds suffice for pseudorandom permutation (PRP) security and 4 rounds for strong pseudorandom permutation (sPRP) security when the round functions are modeled as independent random functions. Accordingly, the security of low-round Feistel-type designs is typically evaluated via *indistinguishability* from an ideal random permutation under under classical chosen-plaintext/chosen-ciphertext attacks (CPA/CCA) and their quantum counterparts (qCPA/qCCA).

Low-round distinguishers (classical and quantum). Tight bounds and matching distinguishers are known for small round numbers. Let each branch have n bits. The 2-round Feistel admits a CPA distinguisher with 2 classical queries [23]. The 3-round Feistel is PRP-secure up to $\mathcal{O}(2^{n/2})$ classical queries [24], yet it admits 3-query CCA distinguishers [23] and $\mathcal{O}(n)$-query qCPA distinguishers [21,19]. The 4-round Feistel is sPRP-secure up to $\mathcal{O}(2^{n/2})$ classical queries [26]. In the quantum setting, although it was initially claimed to be qPRP-secure up to $\mathcal{O}(2^{n/3})$ queries [17], a recent analysis identified a flaw in the proof [5]; instead, the same work established security against non-adaptive quantum chosen-plaintext queries up to $\mathcal{O}(2^{n/6})$ [5]. Furthermore, the 4-round structure remains vulnerable to quantum chosen-ciphertext attacks, admitting $\mathcal{O}(n)$-query qCCA distinguishers [18]. To close this gap and achieve quantum adaptive security, Carolan recently introduced a compressed permutation oracle technique, proving that the 7-round Feistel is strong qPRP secure under quantum chosen-ciphertext attacks [8].

Many quantum distinguishers exploit Simon's period-finding [28]; in particular, Xiang et al. [32] showed how probability-1 truncated differentials yield qCPA distinguishers, and Wang et al. [31] upgraded this approach to qCCA via *improved truncated boomerang* differentials with probability 1.

Beyond distinguishing: from "can we tell" to "can we decrypt." In the ideal-cipher abstraction (round functions as independent random functions), key recovery is not the right question; the canonical notion is distinguishing. However, *full plaintext recovery* is strictly stronger: given a challenge ciphertext, recovering its entire plaintext is a complete break and, a fortiori, implies distinguishing.[5] While plaintext recovery attacks are known in restricted domains (e.g., Feistel-based FPE such as FF1/FF3) [12,3,2,16,11,15], those attacks crucially exploit small domain size and do not reflect a structural limitation of Feistel itself. This motivates the following structural question:

> *Do low-round Feistel-type designs admit* ***full plaintext recovery*** *with query complexity comparable to the best-known distinguishers (classical and quantum)?*

[5] We stress that the converse does not hold: distinguishing does not necessarily imply plaintext recovery. A concrete counterexample separating the two notions is provided in Appendix A.

Why this question matters in practice. Low-round Feistel-type constructions are not merely textbook abstractions: they serve as fundamental building blocks in a wide range of symmetric cryptographic designs, including format-preserving encryption (FPE) [12], authenticated encryption (AE) [25], and tweakable enciphering schemes (TES) [10]. Importantly, Feistel-type structures are frequently employed *either as global permutation cores or as local internal components* within larger constructions, where a small number of rounds is often chosen for efficiency and simplicity.

Consequently, determining whether "low-round indistinguishability gaps" in these structures already imply *full plaintext recovery* is of direct practical relevance. Indeed, whenever a higher-level construction exposes (or allows one to efficiently simulate) the corresponding encryption and decryption oracle interfaces, a plaintext-recovery attack against the underlying low-round Feistel-type component propagates to the overall scheme. Such an attack represents a strictly stronger form of cryptanalytic break than mere distinguishing, as it enables direct reconstruction of sensitive plaintexts rather than only separation from an ideal random permutation.

A hint from enciphering modes. Recent attacks on the tweakable enciphering mode XCB—built from a low-round unbalanced Feistel—show that carefully alternating encryption and decryption queries can expose plaintext information: Bhati et al. [4] recovered partial plaintext with two classical queries, and Wang et al. [30] achieved *full* plaintext recovery with one additional query. These works suggest that, for two-branch designs, some distinguishing patterns may already carry enough algebraic structure to *recover* plaintexts, rather than merely distinguish the construction.

Our main message: recovery is a *structural* phenomenon. We show that, for several canonical low-round two-branch constructions, *distinguishing essentially upgrades to full plaintext recovery* at (nearly) the same query complexity. More precisely, our attacks are *structure-dependent* rather than round-function-dependent: they rely only on the two-branch wiring pattern (e.g., XOR-and-swap together with public linear layers) and on the availability of the corresponding encryption/decryption oracle access, while remaining agnostic to the specific internal realization of the round functions. In particular, in the standard idealized setting where each round function is modeled as an independent random function, our recovery procedures apply uniformly to any construction that instantiates the same low-round Feistel-type structural skeleton and exposes equivalent oracle interfaces. We further show that this structural phenomenon extends to the UFLM framework, and consequently yields plaintext-recovery implications for a broad class of Feistel-based enciphering constructions.

Our contributions. We summarize our results and techniques (see Fig. 1 for dependencies).

(1) Classical full plaintext recovery for low-round Feistel. We recover the full plaintext for any challenge ciphertext of (i) 2-round Feistel under CPA using only three encryption queries, and (ii) 3-round Feistel under CCA using

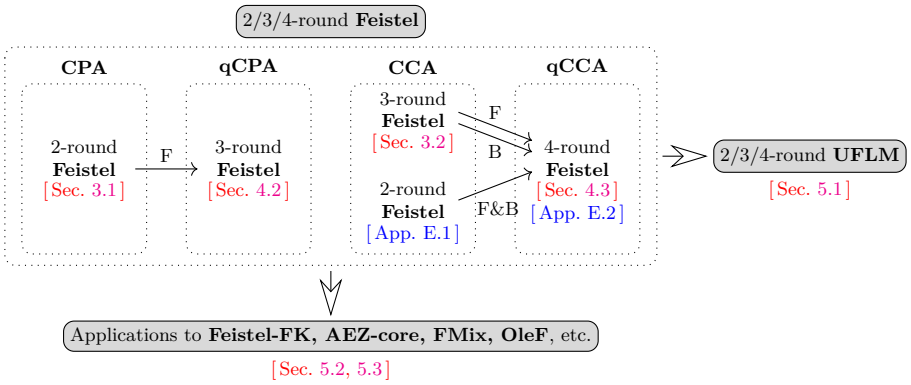


Fig. 1: Logic diagram for our full plaintext recovery attacks on Feistel-type constructions. Here, F denotes forward extension, and B denotes backward extension.

three alternating encryption/decryption queries. The second attack follows a structured *oracle alternation* pattern reminiscent of XCB-style attacks [4,30].

(2) Quantum-enabled round extension via Simon period recovery. We develop a forward/backward single-round extension methodology that combines (i) a classical plaintext-recovery core on a reduced-round segment with (ii) Simon-based recovery of the missing XOR-differences needed to "align" the extension round(s). This yields full plaintext recovery for (i) 3-round Feistel under qCPA and (ii) 4-round Feistel under qCCA using $\mathcal{O}(n)$ quantum queries.

(3) Extension to Lai–Massey and the UFLM framework. We lift our attacks from Feistel to UFLM (and hence to Lai–Massey instances), showing that the same round thresholds and query complexities persist. This supports the interpretation that the vulnerability arises from *shared two-branch structural properties* rather than from any special property of the round functions.

(4) Implications for Feistel-FK and Feistel-based enciphering constructions. We relate an r-round Feistel-FK to an $(r-2)$-round Feistel segment, obtaining plaintext recovery for 4/5/6-round Feistel-FK. (For 2/3/4-round Feistel-KF, which is a direct special case of the standard Feistel structure, our attacks apply directly without any round reduction.) Moreover, we adapt the 4-round qCCA recovery attack to practical Feistel-based enciphering designs, including AEZ-core, FMix, OleF, double-decker, and docked-double-decker.

(5) A unifying view via (improved) truncated boomerang differentials. We explain how the CCA-style plaintext recovery pattern can be understood as instantiating an improved truncated boomerang differential on an internal segment, thereby clarifying the conceptual link between recent boomerang-based distinguishers and full plaintext recovery.

Organization. Section 2 presents preliminaries. Section 3 gives classical plaintext recovery attacks on 2- and 3-round Feistel. Section 4 develops Simon-based

round extension and derives qCPA/qCCA recovery results. Section 5 discusses applications to UFLM/Lai–Massey, Feistel-FK and Feistel-based enciphering constructions. Section 6 clarifies the link between improved truncated boomerang differentials and plaintext recovery. Section 7 concludes.

2 Preliminaries

2.1 Notations

Let $|\mathcal{X}|$ denote the number of elements in a set $\mathcal{X}$. The Cartesian product of two sets $\mathcal{X}$ and $\mathcal{Y}$ is denoted by $\mathcal{X} \times \mathcal{Y}$, i.e., $\mathcal{X} \times \mathcal{Y} = \{(x, y) \mid x \in \mathcal{X}, y \in \mathcal{Y}\}$. Let $X \xleftarrow{\$} \mathcal{X}$ denote that X is drawn uniformly at random from $\mathcal{X}$. We write $\{0,1\}^n$ for the set of all binary strings of length n. For a binary string x of length n, we denote by $x_{[a:b]}$ the substring of x from index a (inclusive) to b (inclusive), where $0 \le a \le b \le n-1$. The concatenation of two binary strings x and y is written as $x \parallel y$, while $x \coloneqq y$ indicates defining x as y. For binary vectors $x = x_0 x_1 \dots x_{n-1}$ and $y = y_0 y_1 \dots y_{n-1}$ with $x_i, y_i \in \{0,1\}$, we use $x \oplus y$ to denote bitwise exclusive-OR and $x \cdot y \coloneqq x_0 y_0 \oplus x_1 y_1 \oplus \cdots \oplus x_{n-1} y_{n-1}$ to denote their inner product over $\mathbb{F}_2$. For $f : \mathcal{X} \to \mathcal{Y}$ and a puncture set $\mathcal{Z}$, define the punctured function $\mathring{f}^{\mathcal{Z}} : \mathcal{X} \cup \mathcal{Z} \to \mathcal{Y} \cup \{\perp\}$ as follows.

$$\mathring{f}^{\mathcal{Z}}(x) \coloneqq \begin{cases} \perp, & \text{if } x \in \mathcal{Z}, \\ f(x), & \text{if } x \in \mathcal{X} \setminus \mathcal{Z}. \end{cases} \tag{1}$$

The invalid symbol $\perp$ is treated as an absorbing element: any function or operation involving $\perp$ (including XOR and truncation) strictly evaluates to $\perp$. For two maps $f : \mathcal{X} \to \mathcal{Y}$ and $g : \mathcal{Y} \to \mathcal{Z}$, we write $g \circ f$ for their composition, defined by $(g \circ f)(x) = g(f(x))$. For a positive integer n, let transformation $\mathbf{T}_{(W,V)}(X, Y) \coloneqq (X, Y) \oplus (W, V)$ for $X, W \in \{0,1\}^n, Y, V \in \{0,1\}^n$.

2.2 Feistel Structure

The Feistel structure is a classical method for constructing a permutation from round functions. The input is first divided into two n-bit branches. Let keyed function $F_i : \{0,1\}^n \to \{0,1\}^n$ be the round function in the i-th round. The encryption of an r-round Feistel network is denoted by $\mathbf{E}_{[r]}$. Given an input (X_1, X_0), the output is (X_r, X_{r+1}), where

$$X_{i+1} = F_i(X_i) \oplus X_{i-1}, \quad i = 1, 2, \dots, r.$$

The round transformation of the i-th Feistel round is defined as $\mathbf{R}_i(X, Y) \coloneqq (F_i(X) \oplus Y,\, X)$, with two variants: $\mathbf{R}'_i(X, Y) \coloneqq (Y,\, F_i(Y) \oplus X)$, $\mathbf{R}''_i(X, Y) \coloneqq (X,\, F_i(X) \oplus Y)$, for $X, Y \in \{0,1\}^n$. These transformations are illustrated in Fig. 2.

The Feistel encryption from round i to j (where j is the last round) is defined as $\mathbf{E}_{[i,j]} \coloneqq \mathbf{R}''_j \circ \mathbf{R}_{j-1} \circ \cdots \circ \mathbf{R}_i$, with the shorthand $\mathbf{E}_{[r]} \coloneqq \mathbf{E}_{[1,r]}$. Extension to

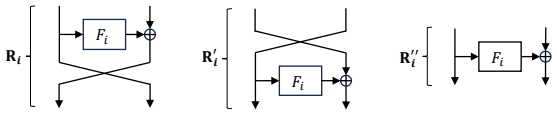


Fig. 2: Round transformation of Feistel $\mathbf{R}_i$ and its variants $\mathbf{R}''_i, \mathbf{R}'_i$.

earlier or later rounds is given by $\mathbf{E}_{[i-1,j]} = \mathbf{E}_{[i,j]} \circ \mathbf{R}_{i-1}$, $\mathbf{E}_{[i,j+1]} = \mathbf{R}'_{j+1} \circ \mathbf{E}_{[i,j]}$. Finally, decryption from round j back to round i is defined as $\mathbf{D}_{[i,j]} \coloneqq \mathbf{E}^{-1}_{[i,j]}$, with the shorthand $\mathbf{D}_{[j]} \coloneqq \mathbf{D}_{[1,j]}$.

2.3 Simon's Algorithm

Simon's algorithm [28] finds the period of a periodic function in polynomial time.

Periodic function. Let n, d be two positive integers, $f : \{0,1\}^n \to \{0,1\}^d$ be a Boolean function. We call f a *periodic function* if there is a unique $s \in \{0,1\}^n \setminus \{0^n\}$ such that $f(x) = f(x \oplus s)$ for all $x \in \{0,1\}^n$.

Simon's problem. Let n, d be two positive integers, $f : \{0,1\}^n \to \{0,1\}^d$ be a periodic function with a period s. Find s.

Classically, if f is a periodic function, we can find its period via collision search with $\mathcal{O}(2^{n/2})$ queries. However, if f is given as a quantum oracle, Simon's algorithm [28] can solve the problem with only $\mathcal{O}(n)$ quantum queries. Let

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

This parameter quantifies the disturbance of other partial periods, i.e., $f(x) = f(x \oplus t)$ where $t \in \{0,1\}^n \setminus \{0^n, s\}$. Kaplan et al. [19] have proved the following theorem.

Theorem 1 (Simon's algorithm with approximate promise [19]). *Let n, d, f, s be defined as in Simon's problem. Let $\varepsilon(f)$ be defined as in Eq. (2), and c be a positive integer. Then Simon's algorithm with cn quantum queries to the periodic function f and $\mathcal{O}(n+d)$ qubits will recover s with probability at least $1 - \left(2(\frac{1+\varepsilon(f)}{2})^c\right)^n$.*

For our plaintext recovery attack, the target ciphertext cannot be queried for decryption. Simon's algorithm remains applicable to the punctured periodic function constructed from such encryption and decryption functions. Its success probability is slightly adjusted. The formal result is given below.

Theorem 2 (Simon's algorithm with approximate promise and a non-queryable element). *Let n, d, f, s be defined as in Simon's problem. Let $\varepsilon(f)$ be defined as in Eq. (2) and c be a positive integer. Let $x^* \in \{0,1\}^n$. Then Simon's algorithm with cn quantum queries to the punctured periodic function $\mathring{f}^{\{x^*\}}$ and $\mathcal{O}(n+d)$ qubits will recover s with probability at least $1 - \frac{cn}{2^n} - \left(2\left(\frac{1+\varepsilon(f)}{2}\right)^c\right)^n$.*

See Appendix B for the proof.

In particular, choosing $c = \lceil \frac{3}{1-\varepsilon(f)} \rceil$ guarantees that the success probability becomes exponentially close to 1 for both Theorems 1 and 2. Therefore, if $\varepsilon(f) \leq \frac{1}{2}$, then $c = 6$ ensures that the probability is exponentially close to 1 with Simon's algorithm using $\mathcal{O}(n)$ quantum queries.

2.4 (Improved) Truncated Boomerang Attack

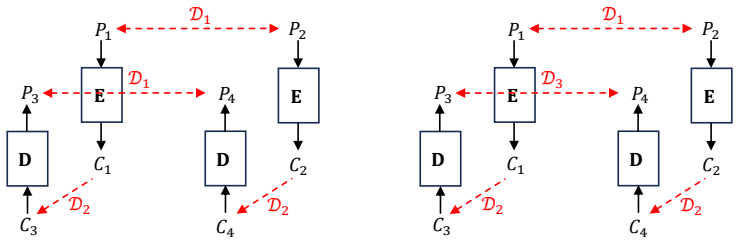


Fig. 3: Truncated boomerang attack (a) vs. improved version (b)

Let $\mathbf{E} = \mathbf{E}_2 \circ \mathbf{E}_1$ be a cipher with inverse (decryption) function $\mathbf{D} = \mathbf{E}^{-1}$. Let $\mathcal{K}$ denote the key space, and $\mathcal{P}$ the plaintext space. The boomerang attack [29] constructs a distinguisher against the cipher $\mathbf{E}$ by combining two short differentials for $\mathbf{E}_1$ and $\mathbf{E}_2$ in the adaptive chosen-ciphertext setting. The *truncated boomerang attack* [29,1], which operates on sets of differences rather than fixed values, offers greater flexibility. Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be sets of differences corresponding to high-probability truncated differentials for $\mathbf{E}_1$ and $\mathbf{E}_2$, respectively. As illustrated in Figure 3a, the attack proceeds as follows:

(1) Choose a random plaintext P_1. Choose a difference $\Delta_1 \in \mathcal{D}_1$ uniformly at random and construct another plaintext $P_2 = P_1 \oplus \Delta_1$. Obtain the corresponding ciphertexts $(C_1, C_2) = (\mathbf{E}(P_1), \mathbf{E}(P_2))$.
(2) Choose differences $\Delta_2, \Delta_2' \in \mathcal{D}_2$ uniformly at random. Construct new ciphertexts $C_3 = C_1 \oplus \Delta_2$ and $C_4 = C_2 \oplus \Delta_2'$. Obtain their decryptions $(P_3, P_4) = (\mathbf{D}(C_3), \mathbf{D}(C_4))$.
(3) If $P_3 \oplus P_4 \in \mathcal{D}_1$, then (P_1, P_2, P_3, P_4) is called a *truncated boomerang quartet*.

The success probability of the truncated boomerang differential $(\mathcal{D}_1, \mathcal{D}_2)$ is

$$\Pr_{\substack{K \leftarrow\$ \mathcal{K}; P_1 \leftarrow\$ \mathcal{P}; \\ \Delta_1 \leftarrow\$ \mathcal{D}_1; \Delta_2, \Delta_2' \leftarrow\$ \mathcal{D}_2}} \left[\mathbf{D}(C_3) \oplus \mathbf{D}(C_4) \in \mathcal{D}_1 \,\middle|\, \begin{array}{l} P_1 \oplus P_2 = \Delta_1, \\ \mathbf{E}(P_1) \oplus C_3 = \Delta_2, \\ \mathbf{E}(P_2) \oplus C_4 = \Delta_2' \end{array} \right].$$

If this probability is less than 1, it is possible to find $\mathcal{D}_3$ with $\mathcal{D}_1 \subsetneqq \mathcal{D}_3 \subsetneqq \{0,1\}^n$ such that the probability becomes 1. Wang et al. [31] relaxed the condition in step

(3) to $P_3 \oplus P_4 \in \mathcal{D}_3$, leading to an *improved* truncated boomerang differential denoted as $(\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3)$, as illustrated in Figure 3b. In this work, we consider the fully generalized case where $\mathcal{D}_1 \subseteq \mathcal{D}_3 \subseteq \{0,1\}^n$. The success probability of this boomerang is defined as:

$$\Pr_{\substack{K \leftarrow\$ \mathcal{K}; P_1 \leftarrow\$ \mathcal{P}; \\ \Delta_1 \leftarrow\$ \mathcal{D}_1; \Delta_2, \Delta_2' \leftarrow\$ \mathcal{D}_2}} \left[\mathbf{D}(C_3) \oplus \mathbf{D}(C_4) \in \mathcal{D}_3 \,\middle|\, \begin{array}{l} P_1 \oplus P_2 = \Delta_1, \\ \mathbf{E}(P_1) \oplus C_3 = \Delta_2, \\ \mathbf{E}(P_2) \oplus C_4 = \Delta_2' \end{array} \right]. \tag{3}$$

3 Classical Plaintext Recovery Attacks on Low-Round Feistel

Given a target ciphertext C of an r-round Feistel cipher, how can one recover the corresponding plaintext P without direct decryption? We refer to this as a *full plaintext recovery attack*[6] if every bit of P can be recovered. In the following, we assume the plaintext length is $2n$ bits, with each branch being n bits in length. Let X_i^j be the value of X_i in the j-th query to the Feistel structure.

3.1 Two-Round Feistel under CPA

We present a full plaintext recovery attack against the 2-round Feistel, which requires 3 classical encryption queries.

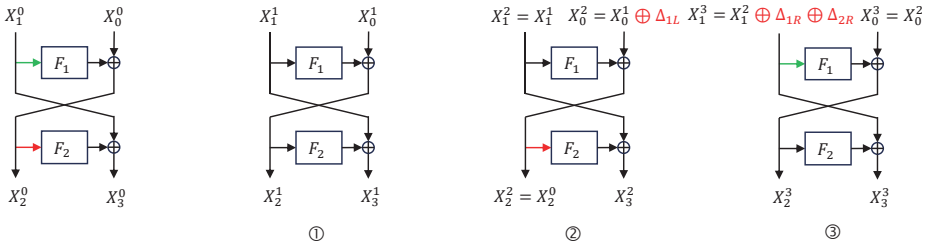


Fig. 4: The process of three classical encryption queries against $\mathbf{E}_{[2]}$. The values in the red (green) branches are identical.

Attack steps. Given an arbitrary ciphertext $(X_2^0, X_3^0) = \mathbf{E}_{[2]}(X_1^0, X_0^0)$, the attacker recovers the corresponding plaintext (X_1^0, X_0^0) through the following steps. The process is illustrated in Fig. 4.

(1) Query the encryption oracle with an arbitrary plaintext (X_1^1, X_0^1) and obtain $(X_2^1, X_3^1) = \mathbf{E}_{[2]}(X_1^1, X_0^1)$. We define $(\Delta_{1L}, \Delta_{1R}) \coloneqq (X_2^0, X_3^0) \oplus (X_2^1, X_3^1)$.

[6] Formal definitions of plaintext recovery security under (quantum) CPA/CCA are given in Appendix C.

(2) Query the encryption oracle with plaintext $(X_1^2, X_0^2) = \mathbf{T}_{(0^n, \Delta_{1L})}(X_1^1, X_0^1)$ and obtain $(X_2^2, X_3^2) = \mathbf{E}_{[2]}(X_1^2, X_0^2)$. We define $(\Delta_{2L}, \Delta_{2R}) \coloneqq (X_2^1, X_3^1) \oplus (X_2^2, X_3^2)$.
(3) Query the encryption oracle with plaintext $(X_1^3, X_0^3) = \mathbf{T}_{(\Delta_{1R} \oplus \Delta_{2R}, 0^n)}(X_1^2, X_0^2)$ and obtain $(X_2^3, X_3^3) = \mathbf{E}_{[2]}(X_1^3, X_0^3)$. We define $(\Delta_{3L}, \Delta_{3R}) \coloneqq (X_2^2, X_3^2) \oplus (X_2^3, X_3^3)$.
(4) Compute the target plaintext $(X_1^0, X_0^0) = \mathbf{T}_{(0^n, \Delta_{3L})}(X_1^3, X_0^3)$.

Correctness. The correctness of the above attack follows from the following proposition.

Proposition 1. *For an arbitrary ciphertext* $(X_2^0, X_3^0) = \mathbf{E}_{[2]}(X_1^0, X_0^0)$*, the following equality holds:*

$$\left(\mathbf{T}_{(0^n, \Delta_{3L})} \circ \mathbf{T}_{(\Delta_{1R} \oplus \Delta_{2R}, 0^n)} \circ \mathbf{T}_{(0^n, \Delta_{1L})}\right)(X_1^1, X_0^1) = (X_1^0, X_0^0),$$

where (X_1^1, X_0^1) *is an arbitrary plaintext and*

$$\begin{aligned}
&(\Delta_{iL}, \Delta_{iR}) = \mathbf{E}_{[2]}(X_1^{i-1}, X_0^{i-1}) \oplus \mathbf{E}_{[2]}(X_1^i, X_0^i), i = 1, 2, 3, \\
&(X_1^2, X_0^2) = \mathbf{T}_{(0^n, \Delta_{1L})}(X_1^1, X_0^1), \\
&(X_1^3, X_0^3) = \mathbf{T}_{(\Delta_{1R} \oplus \Delta_{2R}, 0^n)}(X_1^2, X_0^2).
\end{aligned}$$

Proof. The difference between the left half of the first query's output and the left half of the target ciphertext is given by $X_2^1 \oplus X_2^0 = \Delta_{1L}$. The second encryption query input $(X_1^2, X_0^2) = \mathbf{T}_{(0^n, \Delta_{1L})}(X_1^1, X_0^1)$ ensures that this same difference Δ_{1L} holds for the left halves of the second and first query outputs, i.e., $X_2^2 \oplus X_2^1 = \Delta_{1L}$. Consequently, this offset cancels Δ_{1L}, causing a collision at the input to F_2 between the second query and the target ciphertext generation, i.e.,

$$X_2^2 = X_2^0. \tag{4}$$

From Eq. (4), we have $F_2(X_2^2) = F_2(X_2^0)$. This further establishes the following relations between the second query's encryption process and the target ciphertext generation: $X_1^2 \oplus X_3^2 = X_1^0 \oplus X_3^0$, or equivalently, $X_1^2 \oplus \Delta_{1R} \oplus \Delta_{2R} = X_1^0$. Therefore, the third encryption query input $(X_1^3, X_0^3) = \mathbf{T}_{(\Delta_{1R} \oplus \Delta_{2R}, 0^n)}(X_1^2, X_0^2)$ guarantees

$$X_1^3 = X_1^0. \tag{5}$$

Therefore, the left half of the target plaintext has been recovered.

We now recover the whole target plaintext. Eq. (5) also implies a collision between the inputs to F_1 in the third query and the target ciphertext generation, hence $F_1(X_1^3) = F_1(X_1^0)$. This equality leads to the relation $X_0^3 \oplus X_2^3 = X_0^0 \oplus X_2^0$. From Eq. (4), this becomes $X_0^3 \oplus X_2^3 = X_0^0 \oplus X_2^2$, or equivalently,

$$X_0^3 \oplus \Delta_{3L} = X_0^0. \tag{6}$$

Consequently, $\mathbf{T}_{(0^n,\Delta_{3L})}(X_1^3, X_0^3) = (X_1^0, X_0^0)$ by Eqs. (5) and (6). Thus, the full target plaintext has been recovered.

From the above, we obtain

$$\left(\mathbf{T}_{(0^n,\Delta_{3L})} \circ \mathbf{T}_{(\Delta_{1R}\oplus\Delta_{2R},0^n)} \circ \mathbf{T}_{(0^n,\Delta_{1L})}\right)(X_1^1, X_0^1) = (X_1^0, X_0^0). \qquad \square$$

Success probability and complexity analysis. From the above analysis, this plaintext recovery attack on the 2-round Feistel structure succeeds with probability 1 and requires only 3 classical encryption queries.

3.2 Three-Round Feistel under CCA

In this section, we present a full plaintext recovery attack against the 3-round Feistel cipher. The attack requires 3 classical queries, alternating between decryption and encryption oracles.

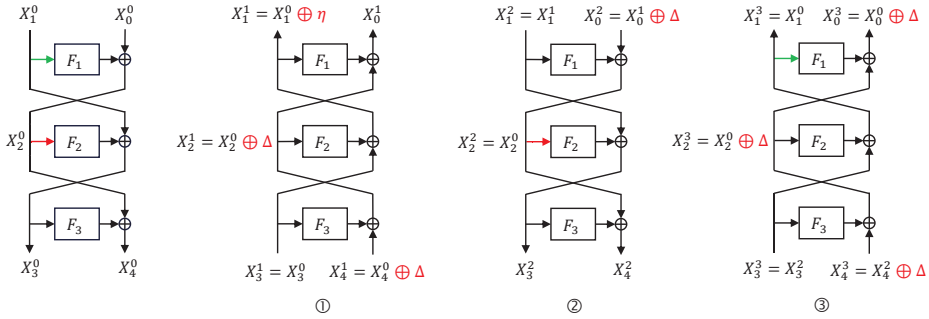


Fig. 5: The process of $\mathring{\mathbf{D}}_{[3]}^{\{C^0\}}$, $\mathbf{E}_{[3]}$, and $\mathring{\mathbf{D}}_{[3]}^{\{C^0\}}$ (Query 1 to Query 3). The values in the red (green) branches are identical.

Attack steps. Given an arbitrary ciphertext $C^0 = (X_3^0, X_4^0) = \mathbf{E}_{[3]}(X_1^0, X_0^0)$, the attacker aims to recover the corresponding plaintext (X_1^0, X_0^0). Note that the attacker is not allowed to query the target ciphertext C^0 directly to the decryption oracle, so it is given access to a punctured decryption oracle $\mathring{\mathbf{D}}_{[3]}^{\{C^0\}}$. As defined in Eq. (1), this oracle behaves identically to the standard decryption oracle except that it returns $\perp$ on input C^0. Formally, for any input $x \in \{0,1\}^{2n}$,

$$\mathring{\mathbf{D}}_{[3]}^{\{C^0\}}(x) = \begin{cases} \perp, & \text{if } x = C^0, \\ \mathbf{D}_{[3]}(x), & \text{if } x \neq C^0. \end{cases}$$

The attacker recovers the target plaintext (X_1^0, X_0^0) through the following steps. The process is illustrated in Fig. 5.

(1) Fix an arbitrary value Δ in $\{0,1\}^n \setminus \{0^n\}$.

(2) Query the decryption oracle with ciphertext $(X_3^1, X_4^1) = \mathbf{T}_{(0^n,\Delta)}(X_3^0, X_4^0)$ and obtain $(X_1^1, X_0^1) = \mathring{\mathbf{D}}_{[3]}^{\{C^0\}}(X_3^1, X_4^1)$.
(3) Query the encryption oracle with plaintext $(X_1^2, X_0^2) = \mathbf{T}_{(0^n,\Delta)}(X_1^1, X_0^1)$, and obtain $(X_3^2, X_4^2) = \mathbf{E}_{[3]}(X_1^2, X_0^2)$.
(4) Query the decryption oracle with ciphertext $(X_3^3, X_4^3) = \mathbf{T}_{(0^n,\Delta)}(X_3^2, X_4^2)$ and obtain $(X_1^3, X_0^3) = \mathring{\mathbf{D}}_{[3]}^{\{C^0\}}(X_3^3, X_4^3)$.
(5) Compute the target plaintext $(X_1^0, X_0^0) = \mathbf{T}_{(0^n,\Delta)}(X_1^3, X_0^3)$.

Correctness. The correctness of the above attack follows from the following proposition.

Proposition 2. *Let Δ be an arbitrary element in $\{0,1\}^n \backslash \{0^n\}$. For an arbitrary ciphertext $C^0 = (X_3^0, X_4^0) = \mathbf{E}_{[3]}(X_1^0, X_0^0)$,*

$$\left(\mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[3]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathbf{E}_{[3]} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[3]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)}\right)(X_3^0, X_4^0) = (X_1^0, X_0^0).$$

Proof. The first decryption query uses the input $(X_3^1, X_4^1) = \mathbf{T}_{(0^n,\Delta)}(X_3^0, X_4^0)$. Since $\Delta \neq 0^n$, this ciphertext differs from the target ciphertext (X_3^0, X_4^0). Hence, $\mathring{\mathbf{D}}_{[3]}^{\{C^0\}}(X_3^1, X_4^1) = \mathbf{D}_{[3]}(X_3^1, X_4^1) \neq \perp$. The map $\mathbf{T}_{(0^n,\Delta)}$ from (X_3^0, X_4^0) to (X_3^1, X_4^1) leaves the left half unchanged, so $X_3^1 = X_3^0$, and ensures the input to F_2 differs from that in the target ciphertext by Δ, i.e.,

$$X_2^1 \oplus X_2^0 = \Delta. \tag{7}$$

As a result, the left halves of the first decryption output and the target plaintext differ by $\eta \coloneqq F_2(X_2^0) \oplus F_2(X_2^0 \oplus \Delta)$, meaning

$$X_1^1 \oplus X_1^0 = \eta. \tag{8}$$

The input to the second encryption query is set as $(X_1^2, X_0^2) = \mathbf{T}_{(0^n,\Delta)}(X_1^1, X_0^1)$. This keeps the left half unchanged, i.e., $X_1^2 = X_1^1 = X_1^0 \oplus \eta$, and also ensures that the input to F_2 in this encryption differs from its input during the first decryption by Δ, namely

$$X_2^2 \oplus X_2^1 = \Delta.$$

This difference cancels the previous Δ in Eq. (7), resulting in a collision between the input to F_2 in the second encryption query and that in the target ciphertext generation, i.e.,

$$X_2^2 = X_2^0. \tag{9}$$

The input to the third decryption query is set as $(X_3^3, X_4^3) = \mathbf{T}_{(0^n,\Delta)}(X_3^2, X_4^2)$. The ciphertext (X_3^3, X_4^3) differs from the target ciphertext (X_3^0, X_4^0); otherwise we would have $(X_3^2, X_4^2) = (X_3^0, X_4^0 \oplus \Delta)$, which would

then give $X_2^2 = X_2^0 \oplus \Delta$. Since $\Delta \neq 0^n$, this contradicts Eq. (9). Hence, $\mathring{\mathbf{D}}_{[3]}^{\{C^0\}}(X_3^3, X_4^3) = \mathbf{D}_{[3]}(X_3^3, X_4^3) \neq \perp$. The map $\mathbf{T}_{(0^n,\Delta)}$ from (X_3^2, X_4^2) to (X_3^3, X_4^3) leaves the left half unchanged with $X_3^3 = X_3^2$ and ensures that the input to F_2 in this decryption differs from its input in the second encryption by Δ, so that

$$X_2^3 \oplus X_2^2 = \Delta.$$

Combining it with Eq. (9), it follows that

$$X_2^3 = X_2^0 \oplus \Delta. \tag{10}$$

Thus, the same difference η holds between the left half of the third decryption output and the left half of the second encryption plaintext, i.e.,

$$X_1^3 \oplus X_1^2 = \eta.$$

Since Eq. (8), the term η cancels out, yielding

$$X_1^3 = X_1^0. \tag{11}$$

Thus, the left half of the target plaintext has been recovered.

Next, we recover the full target plaintext. Eq. (11) also implies a collision between the input to F_1 in the third decryption query and that in the target ciphertext generation. Thus, $F_1(X_1^3) = F_1(X_1^0)$. This yields the intermediate relation $X_2^3 \oplus X_0^3 = X_2^0 \oplus X_0^0$. Combined with Eq. (10), this relation simplifies to

$$X_0^3 \oplus \Delta = X_0^0. \tag{12}$$

From Eqs. (11) and (12), it follows that $\mathbf{T}_{(0^n,\Delta)}(X_1^3, X_0^3) = (X_1^0, X_0^0)$. The full target plaintext has been recovered now.

Therefore, the composed operator satisfies:

$$\left(\mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[3]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathbf{E}_{[3]} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[3]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)}\right)(X_3^0, X_4^0)$$
$$= (X_1^0, X_0^0). \qquad \square$$

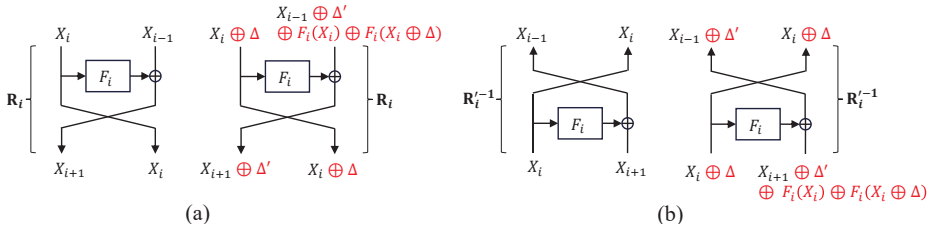
Success probability and complexity analysis. Our analysis shows that the plaintext recovery attack against the 3-round Feistel structure succeeds with probability 1 using only 3 classical encryption and decryption queries.

4 Quantum Extensions on More-Round Feistel via Simon Period Recovery

How can plaintext recovery attacks be extended to Feistel structures with more rounds? This section introduces quantum-enhanced extension methods and demonstrates how they can enable attacks against 3-round Feistel under qCPA and 4-round Feistel under qCCA.

4.1 Forward and Backward Extension via Period Finding

We introduce one-round extension methods for plaintext recovery attacks on Feistel structures.



(a) (b)

Fig. 6: Single-round extensions of the Feistel: (a) forward; (b) backward.

When encryption queries are available, the forward single-round extension can be applied. For any $\Delta', \Delta, X_i, X_{i-1} \in \{0,1\}^n$, define the forward extension transformation

$$\mathbf{T}^{\mathsf{f},s_1}_{(\Delta',\Delta)} \coloneqq \mathbf{T}_{(\Delta,\ \Delta'\oplus s_1)},$$

where $s_1 = F_i(X_i) \oplus F_i(X_i \oplus \Delta)$. Applying $\mathbf{T}^{\mathsf{f},s_1}_{(\Delta',\Delta)}$ to (X_i, X_{i-1}) before round $\mathbf{R}_i$ guarantees that the state after $\mathbf{R}_i$ is transformed by $\mathbf{T}_{(\Delta',\Delta)}$, as illustrated in Fig. 6a. This follows from

$$\begin{aligned}
&\mathbf{R}_i\Big(\mathbf{T}^{\mathsf{f},s_1}_{(\Delta',\Delta)}(X_i, X_{i-1})\Big) = \mathbf{R}_i\Big(\mathbf{T}_{(\Delta,\Delta'\oplus s_1)}(X_i, X_{i-1})\Big)\\
=&\mathbf{R}_i\Big(X_i \oplus \Delta, X_{i-1} \oplus \Delta' \oplus s_1\Big) = \Big(X_{i-1} \oplus F_i(X_i \oplus \Delta) \oplus \Delta' \oplus s_1, X_i \oplus \Delta\Big)\\
=&(X_{i-1} \oplus F_i(X_i), X_i) \oplus (\Delta', \Delta) = (X_{i+1}, X_i) \oplus (\Delta', \Delta) = \mathbf{T}_{(\Delta',\Delta)}(X_{i+1}, X_i).
\end{aligned}$$

Similarly, when decryption queries are available, the backward single-round extension can be applied. For any $\Delta', \Delta, X_i, X_{i+1} \in \{0,1\}^n$, define the backward extension transformation

$$\mathbf{T}^{\mathsf{b},s_2}_{(\Delta',\Delta)} \coloneqq \mathbf{T}_{(\Delta,\ \Delta'\oplus s_2)},$$

with $s_2 = F_i(X_i) \oplus F_i(X_i \oplus \Delta)$. Applying $\mathbf{T}^{\mathsf{b},s_2}_{(\Delta',\Delta)}$ to the state (X_i, X_{i+1}) before $\mathbf{R}_i'^{-1}$ guarantees that the state after $\mathbf{R}_i'^{-1}$ is transformed by $\mathbf{T}_{(\Delta',\Delta)}$, as illustrated in Fig. 6b.

For $\Delta \neq 0^n$, the unknown term s requires recovery via Simon's quantum algorithm. Conversely, $\Delta = 0^n$ enables a classical round extension.

4.2 Three-Round Feistel under qCPA

We extend the plaintext recovery attack against 2-round Feistel presented in Section 3.1 to 3-round Feistel using the forward single-round extension method in Section 4.1. This attack makes $\mathcal{O}(n)$ quantum encryption queries.

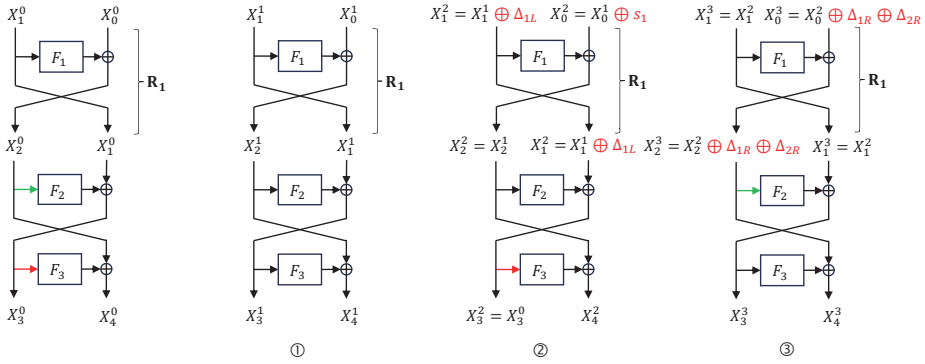


Fig. 7: Process of the three classical encryption queries against $\mathbf{E}_{[3]}$. The values in the red (green) branches are identical.

Attack steps. Given an arbitrary ciphertext $(X_3^0, X_4^0) = \mathbf{E}_{[3]}(X_1^0, X_0^0)$, the attacker recovers the corresponding plaintext (X_1^0, X_0^0) through the following steps. The process is illustrated in Fig. 7.

(1) Query the encryption oracle with an arbitrary plaintext (X_1^1, X_0^1) and obtain $(X_3^1, X_4^1) = \mathbf{E}_{[3]}(X_1^1, X_0^1)$. We define $(\Delta_{1L}, \Delta_{1R}) \coloneqq (X_3^0, X_4^0) \oplus (X_3^1, X_4^1)$.
(2) Recover the secret state $s_1 \coloneqq F_1(X_1^1) \oplus F_1(X_1^1 \oplus \Delta_{1L})$.
(3) Query the encryption oracle with plaintext $(X_1^2, X_0^2) = \mathbf{T}^{\mathsf{f},s_1}_{(0^n,\Delta_{1L})}(X_1^1, X_0^1)$ and obtain $(X_3^2, X_4^2) = \mathbf{E}_{[3]}(X_1^2, X_0^2)$. We define $(\Delta_{2L}, \Delta_{2R}) \coloneqq (X_3^1, X_4^1) \oplus (X_3^2, X_4^2)$.
(4) Query the encryption oracle with plaintext $(X_1^3, X_0^3) = \mathbf{T}^{\mathsf{f},0^n}_{(\Delta_{1R}\oplus\Delta_{2R},0^n)}(X_1^2, X_0^2)$ and obtain $(X_3^3, X_4^3) = \mathbf{E}_{[3]}(X_1^3, X_0^3)$. We define $(\Delta_{3L}, \Delta_{3R}) \coloneqq (X_3^2, X_4^2) \oplus (X_3^3, X_4^3)$.
(5) Recover the secret state $s_3 \coloneqq F_1(X_1^3) \oplus F_1(X_1^3 \oplus \Delta_{3L})$.
(6) Compute the target plaintext $(X_1^0, X_0^0) = \mathbf{T}^{\mathsf{f},s_3}_{(0^n,\Delta_{3L})}(X_1^3, X_0^3)$.

In this attack, it is feasible to recover both s_1 and s_3. If $\Delta_{1L} = 0^n$ (resp. $\Delta_{3L} = 0^n$), then $s_1 = 0^n$ (resp. $s_3 = 0^n$). Else, when $\Delta_{1L} \neq 0^n$ and $\Delta_{3L} \neq 0^n$, both states s_1 and s_3 are of the form $F_1(\alpha_0) \oplus F_1(\alpha_1)$ where $\alpha_0, \alpha_1 \in \{0,1\}^n$ and $\alpha_0 \neq \alpha_1$. Kuwakado and Morii [21] proposed a periodic function $f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$ defined as

$$f(\beta, x) = \mathbf{E}_{[3]}(\alpha_\beta, x)_{[0:n-1]} \oplus \alpha_\beta, \tag{13}$$

which has a secret period $(1, s)$, where $s = F_1(\alpha_0) \oplus F_1(\alpha_1)$. Using Simon's algorithm with quantum queries to f, the value s can be recovered for any $\alpha_0, \alpha_1 \in \{0,1\}^n$ with $\alpha_0 \neq \alpha_1$. Therefore, s_1 and s_3 can be recovered.

Correctness. The attack uses queries to $\mathbf{E}_{[3]}$ to simulate the plaintext recovery attack on the reduced 2-round structure with encryption $\mathbf{E}_{[2,3]}$. The correctness of this attack follows from Proposition 1.

Proposition 3. *For an arbitrary ciphertext* $(X_3^0, X_4^0) = \mathbf{E}_{[3]}(X_1^0, X_0^0)$*, the following equality holds:*

$$\left(\mathbf{T}^{\mathsf{f},s_3}_{(0^n,\Delta_{3L})} \circ \mathbf{T}^{\mathsf{f},0^n}_{(\Delta_{1R}\oplus\Delta_{2R},0^n)} \circ \mathbf{T}^{\mathsf{f},s_1}_{(0^n,\Delta_{1L})}\right)(X_1^1, X_0^1) = (X_1^0, X_0^0),$$

where (X_1^1, X_0^1) *is an arbitrary plaintext and*

$$\begin{aligned}
&(\Delta_{iL}, \Delta_{iR}) = \mathbf{E}_{[3]}(X_1^{i-1}, X_0^{i-1}) \oplus \mathbf{E}_{[3]}(X_1^i, X_0^i), i = 1, 2, 3,\\
&(X_1^2, X_0^2) = \mathbf{T}^{\mathsf{f},s_1}_{(0^n,\Delta_{1L})}(X_1^1, X_0^1),\\
&(X_1^3, X_0^3) = \mathbf{T}^{\mathsf{f},0^n}_{(\Delta_{1R}\oplus\Delta_{2R},0^n)}(X_1^2, X_0^2),\\
&s_j = F_1(X_1^j) \oplus F_1(X_1^j \oplus \Delta_{jL}), j = 1, 3.
\end{aligned}$$

Success probability and complexity analysis. Proposition 3 shows that the plaintext is always recovered if the secret states s_1 and s_3 are correctly recovered.

We now analyze the success probability of recovering these states. All cases can be classified into two types.

(1) $\Delta_{1L} = \Delta_{3L} = 0^n$: In this case, $s_1 = s_3 = 0^n$. Thus, they are recovered with probability 1.

(2) $\Delta_{1L} \neq 0^n$ or $\Delta_{3L} \neq 0^n$: We take $\Delta_{1L} \neq 0^n$ as an example. In this case, $s_1 = F_1(X_1^1) \oplus F_1(X_1^1 \oplus \Delta_{1L}) \neq 0^n$. For the periodic function f defined in Eq. (13), Kaplan et al. [19] proved that $\varepsilon(f) \leq \frac{1}{2}$ holds with overwhelming probability. Consequently, according to Theorem 1, Simon's algorithm recovers such s with a success probability exponentially close to 1, using $\mathcal{O}(n)$ quantum queries to f and $\mathcal{O}(n)$ qubits. This result applies to recovering s_1. The same holds for s_3 when $\Delta_{3L} \neq 0^n$.

In summary, our plaintext recovery attack on the 3-round Feistel structure succeeds with probability at least exponentially close to 1. Combined with the other 3 classical queries to $\mathbf{E}_{[3]}$, and considering that f requires one call to $\mathbf{E}_{[3]}$, the entire attack requires at most $\mathcal{O}(n)$ quantum queries to $\mathbf{E}_{[3]}$ and $\mathcal{O}(n)$ qubits.

4.3 Four-Round Feistel under qCCA

For 4-round Feistel under qCCA, plaintext-recovery attacks can be built by extending the 3-round attack of Section 3.2 via a forward or backward one-round extension, as described in Section 4.1.

4-Round Feistel Plaintext Recovery Attack Using the Forward Round Extension We first describe the attack obtained through forward extension, which uses only $\mathcal{O}(n)$ quantum encryption and decryption queries.

Attack steps. Given an arbitrary ciphertext $C^0 = (X_4^0, X_5^0) = \mathbf{E}_{[4]}(X_1^0, X_0^0)$, the attacker aims to recover the corresponding plaintext (X_1^0, X_0^0). Since the attacker is not allowed to query the target ciphertext C^0 directly to the decryption

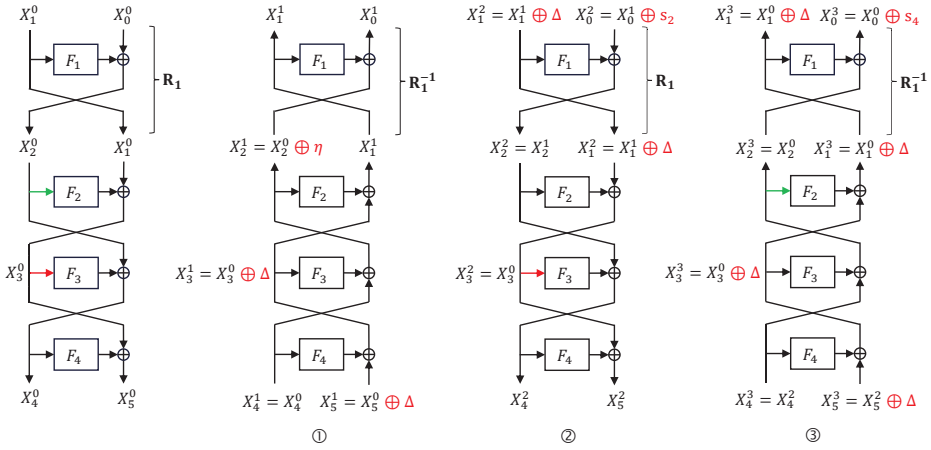


Fig. 8: The process of $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$, $\mathbf{E}_{[4]}$, and $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$ (Query 1 to Query 3). The values in the red (green) branches are identical.

oracle, it is provided access to a punctured quantum decryption oracle $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$. For any classical input $x \in \{0,1\}^{2n}$,

$$\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}(x) = \begin{cases} \perp, & \text{if } x = C^0, \\ \mathbf{D}_{[4]}(x), & \text{if } x \neq C^0. \end{cases}$$

The attacker recovers the target plaintext (X_1^0, X_0^0) through the following steps. The process is illustrated in Fig. 8.

(1) Fix an arbitrary value Δ in $\{0,1\}^n \setminus \{0^n\}$.
(2) Query the decryption oracle with ciphertext $(X_4^1, X_5^1) = \mathbf{T}_{(0^n,\Delta)}(X_4^0, X_5^0)$ and obtain $(X_1^1, X_0^1) = \mathring{\mathbf{D}}_{[4]}^{\{C^0\}}(X_4^1, X_5^1)$.
(3) Recover the secret state $s_2 \coloneqq F_1(X_1^1) \oplus F_1(X_1^1 \oplus \Delta)$.
(4) Query the encryption oracle with plaintext $(X_1^2, X_0^2) = \mathbf{T}_{(0^n,\Delta)}^{\mathbf{f},s_2}(X_1^1, X_0^1)$, and obtain $(X_4^2, X_5^2) = \mathbf{E}_{[4]}(X_1^2, X_0^2)$.
(5) Query the decryption oracle with ciphertext $(X_4^3, X_5^3) = \mathbf{T}_{(0^n,\Delta)}(X_4^2, X_5^2)$ and obtain $(X_1^3, X_0^3) = \mathring{\mathbf{D}}_{[4]}^{\{C^0\}}(X_4^3, X_5^3)$.
(6) Recover the secret state $s_4 \coloneqq F_1(X_1^3) \oplus F_1(X_1^3 \oplus \Delta)$.
(7) Compute the target plaintext $(X_1^0, X_0^0) = \mathbf{T}_{(0^n,\Delta)}^{\mathbf{f},s_4}(X_1^3, X_0^3)$.

In this attack, it is feasible to recover both s_2 and s_4. Given that $\Delta \neq 0^n$, the states s_2 and s_4 are of the form $F_1(\alpha_0) \oplus F_1(\alpha_1)$ where $\alpha_0, \alpha_1 \in \{0,1\}^n$ and $\alpha_0 \neq \alpha_1$. Ito et al. [18] proposed a periodic function $f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$ defined as

$$f(\beta, x) = \mathbf{D}_{[4]}\left(\mathbf{E}_{[4]}(\alpha_\beta, x) \oplus (0^n, \alpha_0 \oplus \alpha_1)\right)_{[0:n-1]} \oplus \alpha_\beta, \tag{14}$$

which has a secret period $(1, s)$, where $s = F_1(\alpha_0) \oplus F_1(\alpha_1)$. Let $\mathcal{Z}$ be the set of all (β, x) such that the input passed to $\mathbf{D}_{[4]}$ in Eq. (14) equals the target ciphertext, i.e.,

$$\mathcal{Z} = \left\{(\beta, x) : \mathbf{E}_{[4]}(\alpha_\beta, x) \oplus (0^n, \alpha_0 \oplus \alpha_1) = C^0\right\}.$$

In fact, $|\mathcal{Z}| \leq 1$. It is easy to verify that $\mathring{f}^{\mathcal{Z}}$ can be constructed from $\mathbf{E}_{[4]}$ and $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$ as follows.

$$\mathring{f}^{\mathcal{Z}}(\beta, x) = \mathring{\mathbf{D}}_{[4]}^{\{C^0\}}\left(\mathbf{E}_{[4]}(\alpha_\beta, x) \oplus (0^n, \alpha_0 \oplus \alpha_1)\right)_{[0:n-1]} \oplus \alpha_\beta. \tag{15}$$

Hence one can query $\mathring{f}^{\mathcal{Z}}$ via quantum oracles to $\mathbf{E}_{[4]}$ and $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$. Simon's algorithm with quantum queries to $\mathring{f}^{\mathcal{Z}}$ recovers the secret value s for any distinct $\alpha_0, \alpha_1 \in \{0,1\}^n$. Therefore, s_2 and s_4 can be recovered.

Correctness. The attack allows queries to $\mathbf{E}_{[4]}$ and $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$ to simulate the plaintext recovery attack on the reduced 3-round structure with encryption $\mathbf{E}_{[2,4]}$ and decryption $\mathring{\mathbf{D}}_{[2,4]}^{\{C^0\}}$. From Proposition 2, the correctness of the above attack follows from the following proposition.

Proposition 4. *Let Δ be an arbitrary value in $\{0,1\}^n \setminus \{0^n\}$. For an arbitrary ciphertext $C^0 = (X_4^0, X_5^0) = \mathbf{E}_{[4]}(X_1^0, X_0^0)$,*

$$\begin{aligned}&\left(\mathbf{T}_{(0^n,\Delta)}^{\mathbf{f},s_4} \circ \mathring{\mathbf{D}}_{[4]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathbf{E}_{[4]} \circ \mathbf{T}_{(0^n,\Delta)}^{\mathbf{f},s_2} \circ \mathring{\mathbf{D}}_{[4]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)}\right)(X_4^0, X_5^0)\\ &= (X_1^0, X_0^0),\end{aligned}$$

where

$$\begin{aligned}&(X_1^i, X_0^i) = \mathring{\mathbf{D}}_{[4]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)}(X_4^{i-1}, X_5^{i-1}), i = 1, 3,\\ &(X_4^2, X_5^2) = \mathbf{E}_{[4]} \circ \mathbf{T}_{(0^n,\Delta)}^{\mathbf{f},s_2}(X_1^1, X_0^1),\\ &s_j = F_1(X_1^{j-1}) \oplus F_1(X_1^{j-1} \oplus \Delta), j = 2, 4.\end{aligned}$$

Success probability and complexity analysis. The above analysis shows that the plaintext is always recovered if the secret states s_2 and s_4 are correctly recovered. We now analyze the success probability of recovering these states. For the periodic function f defined in Eq. (14), we prove $\varepsilon(f) \leq 1/2$ with overwhelming probability in Appendix D. Consequently, according to Theorems 1 and 2, Simon's algorithm recovers such s with a success probability exponentially close to 1, using $\mathcal{O}(n)$ quantum queries to $\mathring{f}^{\mathcal{Z}}$ and $\mathcal{O}(n)$ qubits. This result applies to recovering s_2 and s_4. In summary, our plaintext recovery attack on the 4-round Feistel structure under qCCA using the forward round extension succeeds with probability exponentially close to 1. Combined with the other three classical queries, and considering that $\mathring{f}^{\mathcal{Z}}$ requires one call to $\mathbf{E}_{[4]}$ and one to $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$, the entire attack makes $\mathcal{O}(n)$ quantum queries, and requires $\mathcal{O}(n)$ qubits.

4-Round Feistel Plaintext Recovery Attack Using the Backward Round Extension For the 4-round Feistel structure under qCCA, the plaintext-recovery attack obtained via backward one-round extension proceeds analogously to the forward extension, differing in that it is applied to decryption queries and must recover two secret values of the form $F_4(\alpha_0) \oplus F_4(\alpha_1)$, where $\alpha_0, \alpha_1 \in \{0,1\}^n$ and $\alpha_0 \neq \alpha_1$. Owing to the structural symmetry between $\mathbf{E}_{[4]}$ and $\mathbf{D}_{[4]}$, their roles in (14) can be interchanged. Concretely, define the function $f' : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$ as

$$f'(\beta, x) = \mathbf{E}_{[4]}\Big(\mathbf{D}_{[4]}(\alpha_\beta, x) \oplus (0^n, \alpha_0 \oplus \alpha_1)\Big)_{[0:n-1]} \oplus \alpha_\beta. \tag{16}$$

It has the secret period $(1, F_4(\alpha_0) \oplus F_4(\alpha_1))$. Let $\mathcal{Z}'$ be the set of all (β, x) such that the input passed to $\mathbf{D}_{[4]}$ in Eq. (16) equals the target ciphertext, i.e.,

$$\mathcal{Z}' = \left\{(\beta, x) : (\alpha_\beta, x) = C^0\right\}.$$

Note that $|\mathcal{Z}'| \leq 1$. It is easy to verify that $\mathring{f}'^{\mathcal{Z}'}$ can be constructed from $\mathbf{E}_{[4]}$ and $\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}$ as follows.

$$\mathring{f}'^{\mathcal{Z}'}(\beta, x) = \mathbf{E}_{[4]}\Big(\mathring{\mathbf{D}}_{[4]}^{\{C^0\}}(\alpha_\beta, x) \oplus (0^n, \alpha_0 \oplus \alpha_1)\Big)_{[0:n-1]} \oplus \alpha_\beta. \tag{17}$$

Simon's algorithm can therefore be used to recover this secret value. Consequently, the backward extension attack retains the same complexity as the forward attack, requiring $\mathcal{O}(n)$ quantum queries and qubits.

Remark. In Appendix E, we present another plaintext recovery attack against 4-round Feistel under qCCA with $\mathcal{O}(n)$ quantum queries by extending a plaintext recovery attack against 2-round Feistel under CCA. The objective is achieved by prepending and appending one round to each classical query in the 2-round Feistel attack.

5 Applications to Feistel-Based Constructions

5.1 Applications to UFLM

Dai et al. [9] introduced the two-branch UFLM framework to unify the study of Feistel and Lai-Massey structures. Let F_i be a keyed function on $\{0,1\}^n$ in the i-th round. Generally, F_i is nonlinear. Let

$$\varphi = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \varphi^{-1} = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

be a public invertible linear transformation and its inversion over $\{0,1\}^{2n}$, where A_{ij} and B_{ij} for $i, j \in \{1,2\}$ are matrices over $\{0,1\}^n$. The round transformation for the i-th UFLM round is defined as $\widetilde{\mathbf{R}}_i(X, Y) \coloneqq \varphi(X, F_i(X) \oplus Y)$ with variant

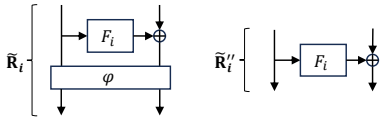


Fig. 9: Round transformation of UFLM $\widetilde{\mathbf{R}}_i$ and its variant $\widetilde{\mathbf{R}}''_i$.

$\widetilde{\mathbf{R}}''_i(X,Y) \coloneqq (X, F_i(X) \oplus Y)$ for $X, Y \in \{0,1\}^n$. These transformations are illustrated in Fig. 9. The encryption of an r-round UFLM network is denoted by $\widetilde{\mathbf{E}}_{[r]} \coloneqq \widetilde{\mathbf{R}}''_r \circ \widetilde{\mathbf{R}}_{r-1} \circ \cdots \circ \widetilde{\mathbf{R}}_1$.

Dai et al. [9] determined the minimal number of rounds required for impossible differential, zero-correlation linear hull, and integral distinguishers in UFLM instances using bijective round functions, relating these bounds to the order of the underlying linear transformation φ. Furthermore, they established that a 4-round UFLM construction achieves sPRP security under the condition that A_{12} is invertible. This matches the security level of a 4-round Feistel structure. In fact, both Feistel and Lai-Massey structures are instances of this "invertible A_{12}" case.

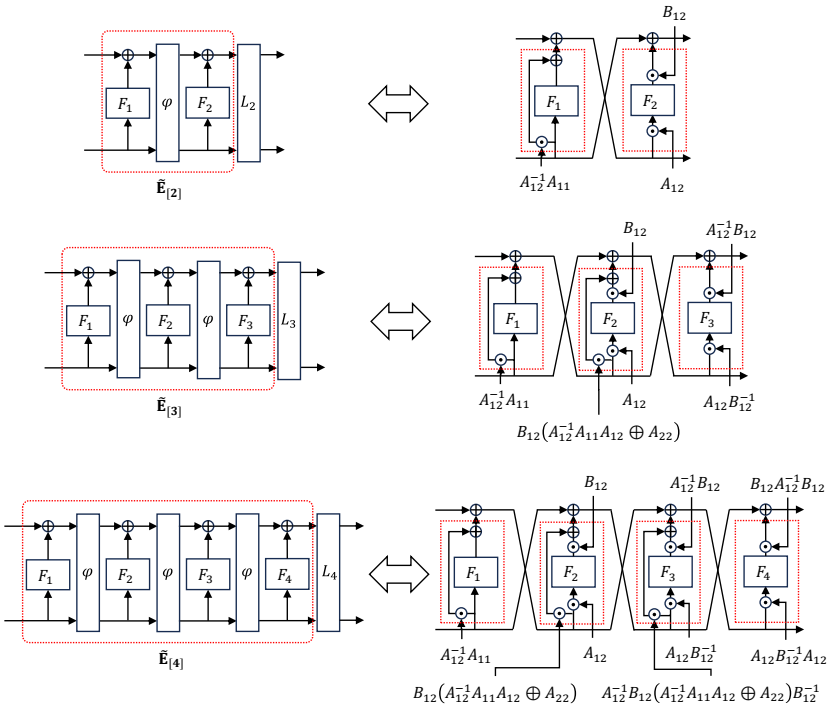


Fig. 10: Transformations between 2/3/4-round UFLM and 2/3/4-round Feistel instances.

Motivated by this result, we explore plaintext recovery attacks on the UFLM construction under the same invertibility assumption for A_{12}. The attacks presented in this section also rely on the invertibility of B_{12}. In Appendix F, we prove that the invertibility of B_{12} is equivalent to that of A_{12}, and that $B_{12} = (A_{21} \oplus A_{22}A_{12}^{-1}A_{11})^{-1}$. We generalize the plaintext recovery attacks previously applied to 2-4 round Feistel structures to the UFLM framework. We show that 2-round, 3-round, and 4-round UFLM are linearly equivalent to 2-round, 3-round, and 4-round Feistel, respectively, with round functions F_i' extended-linearly equivalent to the corresponding F_i in the UFLM, as illustrated in Fig. 10.

- Specifically, for 2-round UFLM, $L_2 \circ \widetilde{\mathbf{E}}_{[2]}$ with $L_2 = \begin{pmatrix} A_{12}^{-1} & 0^n \\ B_{11} & B_{12} \end{pmatrix}$ is equivalent to the encryption algorithm of 2-round Feistel $\mathbf{E}'_{[2]}$ with round functions

$$F_1'(X) = F_1(X) \oplus (A_{12}^{-1}A_{11})X, \qquad F_2'(X) = B_{12}F_2(A_{12}X).$$

- For 3-round UFLM, $L_3 \circ \widetilde{\mathbf{E}}_{[3]}$ with $L_3 = \begin{pmatrix} B_{12}A_{12}^{-1} & 0^n \\ A_{12}^{-1}B_{11} & A_{12}^{-1}B_{12} \end{pmatrix}$ is equivalent to the encryption algorithm of 3-round Feistel $\mathbf{E}'_{[3]}$ with round functions

$$\begin{aligned} F_1'(X) &= F_1(X) \oplus (A_{12}^{-1}A_{11})X, \\ F_2'(X) &= B_{12}F_2(A_{12}X) \oplus B_{12}(A_{12}^{-1}A_{11}A_{12} \oplus A_{22})X, \\ F_3'(X) &= A_{12}^{-1}B_{12}F_3(A_{12}B_{12}^{-1}X). \end{aligned}$$

- For 4-round UFLM, $L_4 \circ \widetilde{\mathbf{E}}_{[4]}$ with $L_4 = \begin{pmatrix} A_{12}^{-1}B_{12}A_{12}^{-1} & 0^n \\ B_{12}A_{12}^{-1}B_{11} & B_{12}A_{12}^{-1}B_{12} \end{pmatrix}$ is equivalent to the encryption algorithm of 4-round Feistel $\mathbf{E}'_{[4]}$ with round functions

$$\begin{aligned} F_1'(X) &= F_1(X) \oplus (A_{12}^{-1}A_{11})X, \\ F_2'(X) &= B_{12}F_2(A_{12}X) \oplus B_{12}(A_{12}^{-1}A_{11}A_{12} \oplus A_{22})X, \\ F_3'(X) &= A_{12}^{-1}B_{12}F_3(A_{12}B_{12}^{-1}X) \oplus A_{12}^{-1}B_{12}(A_{12}^{-1}A_{11}A_{12} \oplus A_{22})B_{12}^{-1}X, \\ F_4'(X) &= B_{12}A_{12}^{-1}B_{12}F_4(A_{12}B_{12}^{-1}A_{12}X). \end{aligned}$$

Therefore, the results for 2/3/4-round UFLM align with the security level of the Feistel structure of corresponding rounds.

5.2 Applications to Feistel-FK

Feistel-FK is a specialized Feistel variant where the i-th round function is defined as $G_i(\cdot) \oplus K_i$, with G_i being a public function and K_i the i-th round key. Li et al. [22] previously introduced distinguishing attacks on 4-round Feistel-FK under CPA using two encryption queries, and Ito et al. [18] introduced distinguishing attacks on 5-round Feistel-FK under qCPA, and 6-round Feistel-FK under qCCA, both requiring $\mathcal{O}(n)$ quantum queries.

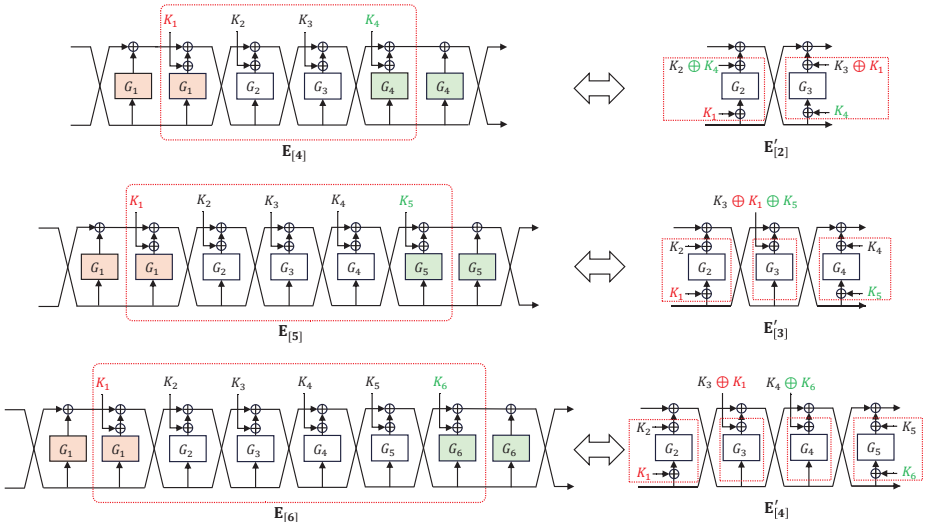


Fig. 11: Transformations between 4/5/6-round Feistel-FK and 2/3/4-round Feistel instances.

We advance these results to plaintext recovery. Indeed, for any $r \geq 4$, the r-round Feistel-FK encryption algorithm $\mathbf{E}_{[r]}$ can be reduced to an $(r-2)$-round Feistel network $\mathbf{E}'_{[r-2]}$, defined by $\mathbf{E}'_{[r-2]} \coloneqq \overline{\mathbf{R}}_r \circ \mathbf{E}_{[r]} \circ \overline{\mathbf{R}}'_1$, where $\overline{\mathbf{R}}'_i(X,Y) \coloneqq (Y, X \oplus G_i(Y))$ and $\overline{\mathbf{R}}_i(X,Y) \coloneqq (Y \oplus G_i(X), X)$ for $X, Y \in \{0,1\}^n$. The resulting transformations for 4/5/6-round Feistel-FK are illustrated in Fig. 11. Then the plaintext recovery results for r-round Feistel-FK ($r = 4, 5, 6$) align with the security level of $(r-2)$-round Feistel structure.

5.3 Applications to AEZ-Core, FMix, OleF, Double-Decker

Beyond abstract Feistel networks, several enciphering schemes explicitly rely on small-round Feistel-like structures, ensuring that the ciphertext maintains the same length as the plaintext.

AEZ-core AEZ [14] is a robust authenticated encryption (RAE) scheme based on the AES round function and supports arbitrary input lengths. The core encryption of AEZ, called AEZ-core, is used to encipher strings of 32 bytes or more. For messages consisting of two blocks, the encryption and decryption procedures of AEZ-core coincide with those of a 4-round Feistel network. Our qCCA plaintext-recovery attack in Section 4.3 applies directly to AEZ-core.

FMix FMix [6] is an inverse-free, single-keyed tweakable enciphering scheme (TES). The encryption algorithm of FMix takes a tweak T and a whole-block plaintext of at least two blocks, and outputs a ciphertext of the same length; we denote this algorithm by $\mathbf{E}^{\mathsf{fmix}}$. The decryption algorithm of FMix takes the tweak T and a whole-block ciphertext of at least two blocks, and returns the

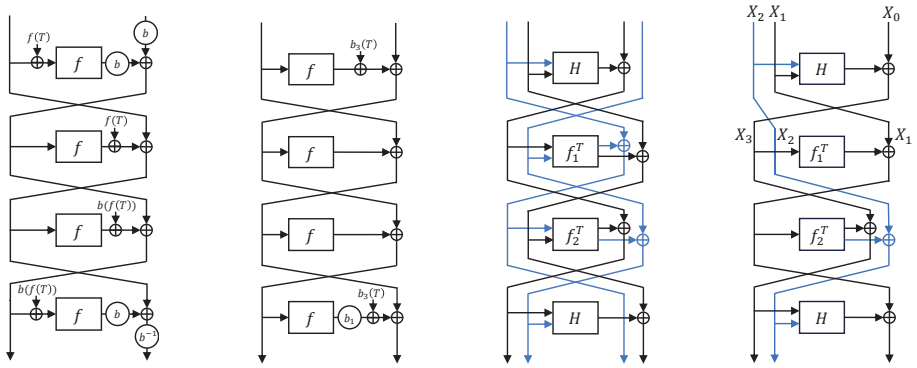


Fig. 12: Encryption of four Feistel-based enciphering constructions. (a) FMix on two-block plaintexts. (b) OleF on two-block plaintexts. (c) Double-decker: the black branch has a fixed length of n bits; the blue branch has an arbitrary size. (d) Docked-double-decker: the black branch has a fixed length of n bits; the blue branch has an arbitrary size.

plaintext; we denote this algorithm by $\mathbf{D}^{\mathsf{fmix}}$. Let n be the block length. The components used in FMix are an n-bit encryption function f and a balanced linear permutation b on n bits.

We focus on the case of two plaintext blocks. The encryption algorithm for FMix is depicted in Fig. 12a. Define $\mathbf{B}(X, Y) = (X, b(Y))$ for $X, Y \in \{0,1\}^n$. Then $\widetilde{\mathbf{E}}^{\mathsf{fmix}} := \mathbf{B} \circ \mathbf{E}^{\mathsf{fmix}} \circ \mathbf{B}^{-1}$ can be viewed as a special case of a 4-round Feistel network whose round functions are

$$F_1^T(X) = b\big(f(f(T) \oplus X)\big), \qquad F_2^T(X) = f(X) \oplus f(T),$$
$$F_3^T(X) = f(X) \oplus b(f(T)), \qquad F_4^T(X) = b\Big(f\big(b(f(T)) \oplus X\big)\Big),$$

where $X \in \{0,1\}^n$. $b(X, Y) = (X \oplus Y, X)$ is a software-friendly choice, where $X, Y \in \{0,1\}^{\frac{n}{2}}$. So we assume b is public. Then, given the oracles of $\mathbf{E}^{\mathsf{fmix}}$ and $\mathbf{D}^{\mathsf{fmix}}$, one is able to construct oracles of $\widetilde{\mathbf{E}}^{\mathsf{fmix}}$ and $(\widetilde{\mathbf{E}}^{\mathsf{fmix}})^{-1}$. If the same tweak T is used for every encryption and decryption query, the plaintext-recovery attack against 4-round Feistel under qCCA is directly applicable.

OleF OleF [7] is a single-key inverse-free online cipher (OC). Let n denote the block length. OleF employs a keyed function f on n bits and four balanced linear permutations b_1, b_2, b_3, b_4, where each b_i is defined as $b_i(t) := \alpha^i \cdot t$ for a primitive field element α and finite-field multiplication '$\cdot$'.

As illustrated in Fig. 12b for two-block plaintexts, OleF can be viewed as a 4-round Feistel network. Its round functions are

$$F_1^T(X) = f(X) \oplus b_3(T), \qquad F_2(X) = f(X),$$
$$F_3(X) = f(X), \qquad F_4^T(X) = b_1(f(X)) \oplus b_3(T),$$

where T is a tweak and $X \in \{0,1\}^n$. If the same tweak T is used for every encryption and decryption query, the plaintext-recovery attack on 4-round Feistel under qCCA can be applied directly.

Double-Decker Double-decker [13] is a tweakable wide-block cipher mode. It employs two independently keyed doubly-extendable cryptographic (deck) functions f_1 and f_2, each accepting arbitrarily long input and producing a potentially unbounded output, together with a separately keyed hash function H that outputs n bits. These three components constitute double-decker.

As illustrated in Fig. 12c, double-decker can be viewed as a 4-round Feistel network with two arbitrarily sized branches. The corresponding round functions are

$$F_1(X) = H(X) \parallel 0^*, \qquad F_2^T(Y) = f_1^T(Y),$$
$$F_3^T(X) = f_2^T(X), \qquad F_4(Y) = 0^* \parallel H(Y),$$

where T is a tweak, $X, Y \in \{0,1\}^{n+*}$. Assume that the same tweak T is used for every encryption and decryption query. Even when the two branch lengths differ, the function appearing in Eq. (14) remains periodic. Hence, the plaintext-recovery attack against 4-round Feistel under qCCA remains directly applicable.

Docked Double-Decker Docked-double-decker [13] is a tweakable wide-block cipher mode derived from double-decker by shifting the bulk of the input from the deck functions to the keyed hash functions. The construction uses the same components as double-decker: two independently keyed deck functions f_1, f_2 and a separately keyed hash function H with n-bit output. Denote the encryption and decryption algorithms of docked-double-decker by $\mathbf{E}^{\mathsf{ddd}}$ and $\mathbf{D}^{\mathsf{ddd}}$, respectively.

The encryption algorithm for docked-double-decker is depicted in Fig. 12d. The input is parsed as (X_2, X_1, X_0) where $X_1, X_0 \in \{0,1\}^n$ and $X_2 \in \{0,1\}^*$. Define $X_3 = H(X_2 \parallel X_1) \oplus X_0$. The first round is the transformation $\mathbf{R}^{\mathsf{ddd}}(X_2, X_1, X_0) := (X_3, X_2, X_1)$. Interpreting its output as the two-branch state $(X_3 \parallel X_2,\ X_1)$, the subsequent construction forms a 3-round Feistel network with round functions

$$F_1^T(X \parallel Y) = f_1^T(X), \qquad F_2^T(Z) = f_2^T(Z),$$
$$F_3(X \parallel Y) = H(X \parallel Y),$$

where $X, Z \in \{0,1\}^n$, $Y \in \{0,1\}^*$, and T is a tweak. Assume that the same tweak T is used for every encryption and decryption query. Consequently, a plaintext-recovery attack on 3-round Feistel can be applied to these last three rounds, using the transformation $\mathbf{T}_{(0^n \| 0^{|X_2|}, \Delta)}$ for any fixed $\Delta \in \{0,1\}^n \setminus \{0^n\}$.

To extend this attack to the full construction, we perform a forward-round extension. Applying the transformation $\mathbf{T}_{(0^{|X_2|},\, \Delta,\, H(X_2\|X_1)\oplus H(X_2\|(X_1\oplus\Delta)))}$ to (X_2, X_1, X_0) results, after $\mathbf{R}^{\mathsf{ddd}}$, in a state transformed by $\mathbf{T}_{(0^n,0^{|X_2|},\Delta)}$, which is exactly $\mathbf{T}_{(0^n\|0^{|X_2|},\Delta)}$. The secret value required to construct this transformation, $s \coloneqq H(\gamma \parallel \alpha_0) \oplus H(\gamma \parallel \alpha_1)$, with arbitrary $\gamma \in \{0,1\}^*$ and distinct $\alpha_0, \alpha_1 \in \{0,1\}^n$, can itself be recovered. Define the periodic function

$$
\begin{aligned}
f(\beta, x) = &\left(\mathbf{D}^{\mathsf{ddd}}\big(\mathbf{E}^{\mathsf{ddd}}(\gamma, \alpha_\beta, x) \oplus (0^n, 0^{|\gamma|}, \alpha_0 \oplus \alpha_1)\big)\right)_{[|\gamma|:|\gamma|+n-1]} \oplus \alpha_\beta \qquad (18)\\
=&f_1^T \begin{pmatrix} H(\gamma \parallel \alpha_\beta) \oplus x \\ \oplus f_2^T\left(f_1^T\left(H(\gamma \parallel \alpha_\beta) \oplus x\right) \oplus \alpha_\beta\right) \\ \oplus f_2^T\left(f_1^T\left(H(\gamma \parallel \alpha_\beta) \oplus x\right) \oplus \alpha_{\beta\oplus 1}\right) \end{pmatrix} \\
&\oplus f_1^T\left(H(\gamma \parallel \alpha_\beta) \oplus x\right) \oplus \alpha_0 \oplus \alpha_1
\end{aligned}
$$

for $\beta \in \{0,1\}$ and $x \in \{0,1\}^n$. Th e function f has period $(1, s)$. Denote the target ciphertext by C^*. Let $\mathcal{Z}$ be the set of all pairs (β, x) such that the input to $\mathbf{D}^{\mathsf{ddd}}$ in Eq. (18) equals C^*, i.e.,

$$
\mathcal{Z} = \left\{(\beta, x) : \mathbf{E}^{\mathsf{ddd}}(\gamma, \alpha_\beta, x) \oplus (0^n, 0^{|\gamma|}, \alpha_0 \oplus \alpha_1) = C^*\right\}.
$$

Actually, $|\mathcal{Z}| \leq 1$. It is easy to obtain that $\mathring{f}^{\mathcal{Z}}$ can be constructed from $\mathbf{E}^{\mathsf{ddd}}$ and $\mathring{\mathbf{D}}^{\mathsf{ddd}\,\{C^*\}}$ as follows.

$$
\mathring{f}^{\mathcal{Z}}(\beta, x) = \left(\mathring{\mathbf{D}}^{\mathsf{ddd}\,\{C^*\}}\big(\mathbf{E}^{\mathsf{ddd}}(\gamma, \alpha_\beta, x) \oplus (0^n, 0^{|\gamma|}, \alpha_0 \oplus \alpha_1)\big)\right)_{[|\gamma|:|\gamma|+n-1]} \oplus \alpha_\beta.
$$

By Theorems 1 and 2, applying Simons algorithm to $\mathring{f}^{\mathcal{Z}}$ recovers s using $\mathcal{O}(n)$ quantum queries and $\mathcal{O}(n)$ qubits. Hence, a plaintext-recovery attack on the full docked-double-decker is feasible with the same quantum resources.

6 Linking Improved Truncated Boomerang Differentials to Plaintext Recovery Attacks

While traditional (truncated) boomerang attacks focus primarily on distinguishing the cipher from a random permutation or recovering key material, their inherent structure also provides a powerful avenue for full plaintext recovery in a (quantum) chosen-ciphertext attack model. This connection provides a conceptual unification between the two attack paradigms. Notably, the object is general and not limited to Feistel or UFLM.

6.1 Improved Truncated Boomerang Differential for Full Plaintext Recovery

Let $\mathbf{E}$ be a block cipher with a key space $\mathcal{K}$, a plaintext space $\mathcal{P}$, and a ciphertext space $\mathcal{C}$. Let $\mathbf{D}$ denote the decryption of $\mathbf{E}$, i.e., $\mathbf{D} = \mathbf{E}^{-1}$.

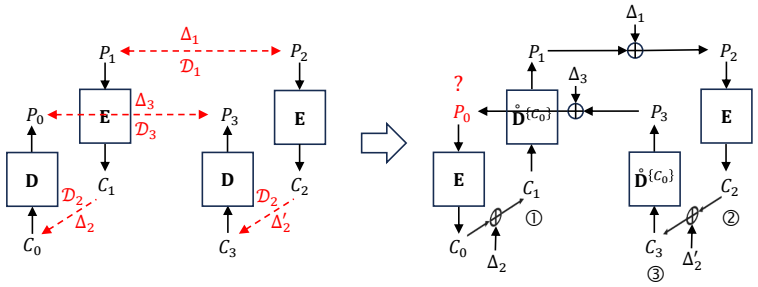


Fig. 13: From improved truncated boomerang differential to plaintext recovery attack.

Assumption. Specifically, assume $\mathbf{E}$ has an improved truncated boomerang differential $(\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3)$ with probability p and the zero difference is not in $\mathcal{D}_2$ and $\mathcal{D}_3$.

Then we can make a plaintext recovery attack against it, which requires 3 encryption and decryption queries.

Attack steps. Randomly choose a target plaintext $P_0 \xleftarrow{\$} \mathcal{P}$ and compute the target ciphertext $C_0 = \mathbf{E}(P_0)$. Given C_0, the adversary recovers the exact coset containing the corresponding plaintext P_0 through the following steps. The logic diagram of this attack is shown in Fig. 13.

(1) Choose a difference $\Delta_2 \leftarrow\!\$\, \mathcal{D}_2$. Query the decryption oracle with ciphertext $C_1 = C_0 \oplus \Delta_2$ and obtain $P_1 = \mathring{\mathbf{D}}^{\{C_0\}}(C_1)$.
(2) Choose a difference $\Delta_1 \leftarrow\!\$\, \mathcal{D}_1$. Query the encryption oracle with plaintext $P_2 = P_1 \oplus \Delta_1$, and obtain $C_2 = \mathbf{E}(P_2)$.
(3) Choose a difference $\Delta_2' \leftarrow\!\$\, \mathcal{D}_2$. Query the decryption oracle with ciphertext $C_3 = C_2 \oplus \Delta_2'$ and obtain $P_3 = \mathring{\mathbf{D}}^{\{C_0\}}(C_3)$.
(4) Output the coset $\{P_3 \oplus \Delta_3 | \Delta_3 \in \mathcal{D}_3\}$.

Success probability. The probability of this attack is bounded by

$$\Pr_{\substack{K \leftarrow\!\$\, \mathcal{K}; P_0 \leftarrow\!\$\, \mathcal{P}; \\ \Delta_1 \leftarrow\!\$\, \mathcal{D}_1; \\ \Delta_2, \Delta_2' \leftarrow\!\$\, \mathcal{D}_2}} \left[P_0 \in \left\{ \mathring{\mathbf{D}}^{\{C_0\}}(C_3) \oplus \Delta_3 | \Delta_3 \in \mathcal{D}_3 \right\} \middle| \begin{array}{l} \mathring{\mathbf{D}}^{\{C_0\}}(C_1) \oplus P_2 = \Delta_1, \\ \mathbf{E}(P_0) \oplus C_1 = \Delta_2, \\ \mathbf{E}(P_2) \oplus C_3 = \Delta_2' \end{array} \right]. \tag{19}$$

Since the zero difference is not in $\mathcal{D}_2$, we have $C_1 = C_0 \oplus \Delta_2 \neq C_0$, and thus $\mathring{\mathbf{D}}^{\{C_0\}}(C_1) = \mathbf{D}(C_1)$. Similarly, because the zero difference is not in $\mathcal{D}_3$, it holds that $\mathbf{D}(C_3) \neq \mathbf{D}(C_0)$. Since $\mathbf{D}$ is a permutation, this implies $C_3 \neq C_0$, and hence $\mathring{\mathbf{D}}^{\{C_0\}}(C_3) = \mathbf{D}(C_3)$. Then we can safely replace the punctured oracle $\mathring{\mathbf{D}}^{\{C_0\}}$ with the standard permutation $\mathbf{D}$ in the subsequent probabilistic analysis. Furthermore, since $\mathbf{E}$ is a permutation, the uniform distribution of P_0 implies

that $C_0 = \mathbf{E}(P_0)$ is uniformly distributed over $\mathcal{C}$. For any fixed Δ_2, the uniformity of C_0 implies that $C_1 = C_0 \oplus \Delta_2$ is also uniform over $\mathcal{C}$. As $\mathbf{D}$ is exactly applied to C_1 (since $C_1 \neq C_0$), the plaintext $P_1 = \mathbf{D}(C_1)$ is uniformly distributed over $\mathcal{P}$. Consequently, by changing the random variable from P_0 to P_1, Eq. (19) equals

$$\Pr_{\substack{K \leftarrow\$\mathcal{K}; P_1 \leftarrow\$\mathcal{P}; \\ \Delta_1 \leftarrow\$\mathcal{D}_1; \Delta_2, \Delta_2' \leftarrow\$\mathcal{D}_2}} \left[\mathbf{D}(C_0) \oplus \mathbf{D}(C_3) \in \mathcal{D}_3 \,\middle|\, \begin{array}{l} P_1 \oplus P_2 = \Delta_1, \\ C_0 \oplus \mathbf{E}(P_1) = \Delta_2, \\ \mathbf{E}(P_2) \oplus C_3 = \Delta_2' \end{array} \right] = p$$

by the assumption that $\mathbf{E}$ exhibits an improved truncated boomerang differential $(\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3)$ with probability p.

When $p = 1$, the plaintext recovery attack succeeds with probability 1 for any given target ciphertext C_0 in the ciphertext space.

Example 1. A 3-round Feistel cipher $\mathbf{E}_{[3]}$ operating on a $2n$-bit block exhibits an improved truncated boomerang differential $(\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3)$ with probability 1. Specifically, for $\mathcal{D}_1 = \mathcal{D}_2 = \mathcal{D}_3 = \{(0^n, \Delta)\}$ where $\Delta \in \{0,1\}^n \setminus \{0^n\}$, the differential holds. Applying our plaintext recovery method in this section, for an arbitrary ciphertext C_0, an adversary can always output a set that contains precisely one element, namely the corresponding plaintext. This confirms the known attack in Section 3.2.

6.2 Plaintext Recovery for Extended Ciphers

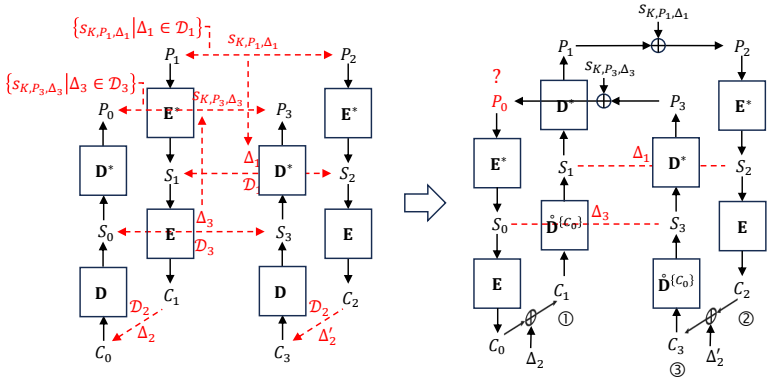


Fig. 14: Plaintext recovery attack via improved truncated boomerang differential on $\mathbf{E}$ and traditional differentials on $\mathbf{E}^*$.

Merely applying the improved truncated boomerang differential does not suffice to extend the plaintext-recovery attack to a sufficiently large number of rounds. Suppose the attack currently targets the cipher $\mathbf{E}$. To achieve a plaintext-recovery attack on the extended cipher $\widetilde{\mathbf{E}} = \mathbf{E} \circ \mathbf{E}^*$, if one can recover from

$\widetilde{\mathbf{E}}$ certain secret states that enable high-probability differentials over $\mathbf{E}^*$, the attack on the full $\widetilde{\mathbf{E}}$ becomes feasible. To describe it, we accordingly define $\mathbf{D}^* \coloneqq \mathbf{E}^{*-1}, \mathbf{D} \coloneqq \mathbf{E}^{-1}$, and $\widetilde{\mathbf{D}} \coloneqq \widetilde{\mathbf{E}}^{-1}$. Let $\mathcal{K}$, $\mathcal{P}$, and $\mathcal{C}$ denote the key space, the plaintext space, and the ciphertext space, respectively.

Randomly choose a target plaintext $P_0 \xleftarrow{\$} \mathcal{P}$ and compute the target ciphertext $C_0 = \widetilde{\mathbf{E}}(P_0)$.

Assumptions. Specifically, assume:

(1) $\mathbf{E}$ has an improved truncated boomerang differential $(\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3)$ with probability p and the zero difference is not in $\mathcal{D}_2$ and $\mathcal{D}_3$.
(2) For a uniformly random key K, any $P \in \mathcal{P}$ and $\Delta \in \mathcal{D}_1 \cup \mathcal{D}_3$, there exists a secret input difference $s_{K,P,\Delta}$ such that $\mathbf{E}^*(P) \oplus \mathbf{E}^*(P \oplus s_{K,P,\Delta}) = \Delta$. $s_{K,P,\Delta}$ can be recovered with probability close to 1 using Simons algorithm with quantum queries to $\widetilde{\mathbf{E}}$ and $\widetilde{\mathbf{D}}^{\circ\{C_0\}}$.

Example 2. Take the 4-round Feistel with $2n$-bit encryption $\mathbf{E}_{[4]} = \mathbf{E}_{[2,4]} \circ \mathbf{R}_1$ as an example.

(1) $\mathbf{E}_{[2,4]}$ has an improved truncated boomerang differential $(\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3)$ with probability 1 for any $\mathcal{D}_1 = \mathcal{D}_2 = \mathcal{D}_3 = \{(0^n, \Delta')\}$ where $\Delta' \in \{0,1\}^n \setminus \{0^n\}$.
(2) For a uniformly random key K and any $P = (X_1, X_0) \in \{0,1\}^n$ and $\Delta = (0^n, \Delta') \in \mathcal{D}_1 \cup \mathcal{D}_3$, there exists a secret input difference $s_{K,P,\Delta} = (\Delta', F_1(X_1) \oplus F_1(X_1 \oplus \Delta'))$ such that $\mathbf{R}_1(P) \oplus \mathbf{R}_1(P \oplus s_{K,P,\Delta}) = \Delta$. $s_{K,P,\Delta}$ can be recovered with probability close to 1 using Simons algorithm.

Attack steps. Given C_0, the attacker recovers a precise coset containing the corresponding plaintext P_0 through the following steps. The logic diagram of this attack is shown in Fig. 14.

(1) Choose a difference $\Delta_2 \leftarrow\!\$\, \mathcal{D}_2$. Query the decryption oracle with ciphertext $C_1 = C_0 \oplus \Delta_2$ and obtain $P_1 = \widetilde{\mathbf{D}}^{\circ\{C_0\}}(C_1)$.
(2) Choose a difference $\Delta_1 \leftarrow\!\$\, \mathcal{D}_1$. Recover the secret input difference s_{K,P_1,Δ_1} such that $\mathbf{E}^*(P_1) \oplus \mathbf{E}^*(P_1 \oplus s_{K,P_1,\Delta_1}) = \Delta_1$.
(3) Query the encryption oracle with plaintext $P_2 = P_1 \oplus s_{K,P_1,\Delta_1}$ and obtain $C_2 = \widetilde{\mathbf{E}}(P_2)$.
(4) Choose a difference $\Delta_2' \leftarrow\!\$\, \mathcal{D}_2$. Query the decryption oracle with ciphertext $C_3 = C_2 \oplus \Delta_2'$ and obtain $P_3 = \widetilde{\mathbf{D}}^{\circ\{C_0\}}(C_3)$.
(5) For all $\Delta_3 \in \mathcal{D}_3$, recover secret input differences s_{K,P_3,Δ_3} such that $\mathbf{E}^*(P_3) \oplus \mathbf{E}^*(P_3 \oplus s_{K,P_3,\Delta_3}) = \Delta_3$.
(6) Output the coset $\{P_3 \oplus s_{K,P_3,\Delta_3} | \Delta_3 \in \mathcal{D}_3\}$.

Success probability. This attack succeeds with probability almost p. When $p = 1$, the plaintext recovery attack succeeds with probability close to 1 for any given target ciphertext C_0 in the ciphertext space.

Remark. Similarly, one may also apply the improved truncated boomerang differential to the former part and combine it with compatible differential propagations in the latter section to mount a plaintext recovery attack. Furthermore, such an attack remains feasible even when compatible difference propagations are introduced both before and after the section where the improved boomerang differential is applied.

7 Conclusions

We have shown that, for low-round two-branch Feistel-type constructions, existing distinguishing attacks can be *systematically lifted* to full plaintext-recovery attacks with essentially the same query complexity, in both classical and quantum models. Our work develops a unified recovery framework that transforms partial structural leakage into complete plaintext reconstruction. The framework applies uniformly to Feistel networks, Feistel–FK constructions, and the more general UFLM structure, and extends naturally to the quantum setting via Simon-type period-finding techniques. Consequently, the same structural weakness propagates to several modern Feistel-based enciphering designs, including AEZ-core, FMix, OleF, double-decker, and docked-double-decker. We further clarify when improved truncated boomerang differentials inherently imply full plaintext recovery, thereby conceptually unifying these two lines of cryptanalysis.

Traditionally, the cryptanalysis of block ciphers has focused on *key recovery* as the primary objective. Our results indicate that *plaintext recovery* provides a complementary analytic viewpoint, particularly well suited to constructions that explicitly employ low-round Feistel-type structures. For general block-cipher analysis, this perspective should be interpreted as a conceptual lens rather than a universal replacement for key-recovery-based methods.

In practical deployments, significantly more rounds are typically employed, with round functions that may not be ideal random functions. Characterizing the exact round thresholds at which full plaintext recovery becomes feasible, or can be provably ruled out, under realistic instantiations remains an important open problem for future research.

Acknowledgments. The authors would like to thank the anonymous reviewers of EUROCRYPT 2026 and CRYPTO 2026 for their many helpful comments and suggestions. We are especially grateful to one reviewer of CRYPTO 2026 for pointing out a deeper connection between the Feistel structure and UFLM, which greatly helped simplify the presentation of the paper. The work is supported by the National Key Research and Development Program of China (No. 2023YFB3105802), the National Key Laboratory of Security Communication Foundation (2024, 6142103042409) and the Research Foundation of Ningbo University of Technology (25KQ076).

References

1. Bariant, A., Leurent, G.: Truncated boomerang attacks and application to AES-based ciphers. In: Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part IV. pp. 3–35. Lecture Notes in Computer Science, Springer (2023). https://doi.org/10.1007/978-3-031-30634-1_1 7
2. Bellare, M., Hoang, V.T., Tessaro, S.: Message-recovery attacks on Feistel-based format preserving encryption. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. pp. 444–455. ACM (2016). https://doi.org/10.1145/2976749.2978390 2
3. Bellare, M., Ristenpart, T., Rogaway, P., Stegers, T.: Format-preserving encryption. In: Selected Areas in Cryptography, 16th Annual International Workshop, SAC 2009, Revised Selected Papers. pp. 295–312. Lecture Notes in Computer Science, Springer (2009). https://doi.org/10.1007/978-3-642-05445-7_19 2
4. Bhati, A.S., Andreeva, E.: Breaking the IEEE encryption standard XCB-AES in two queries. In: Advances in Cryptology - CRYPTO 2025 - 45th Annual International Cryptology Conference, Proceedings, Part V. pp. 172–199. Lecture Notes in Computer Science, Springer (2025). https://doi.org/10.1007/978-3-032-01901-1_6 3, 4
5. Bhaumik, R., Cogliati, B., Ethan, J., Jha, A.: Mind the bad norms - revisiting compressed oracle-based quantum indistinguishability proofs. In: Advances in Cryptology - ASIACRYPT 2024 - 30th International Conference on the Theory and Application of Cryptology and Information Security, Proceedings, Part IX. pp. 215–247. Lecture Notes in Computer Science, Springer (2024). https://doi.org/10.1007/978-981-96-0947-5_8 2
6. Bhaumik, R., Nandi, M.: An inverse-free single-keyed tweakable enciphering scheme. In: Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Proceedings, Part II. pp. 159–180. Lecture Notes in Computer Science, Springer (2015). https://doi.org/10.1007/978-3-662-48800-3_7 21
7. Bhaumik, R., Nandi, M.: Olef: an inverse-free online cipher. an online SPRP with an optimal inverse-free construction. IACR Trans. Symmetric Cryptol. **2016**(2), 30–51 (2016). https://doi.org/10.13154/TOSC.V2016.I2.30-51 22
8. Carolan, J.: Compressed permutation oracles. IACR Cryptol. ePrint Arch. **2025**, 1734 (2025), https://eprint.iacr.org/2025/1734 2
9. Dai, Z., Guo, C., Li, C.: UFLM: A unified framework for Feistel structure and Lai-Massey structure. In: Progress in Cryptology - INDOCRYPT 2024 - 25th International Conference on Cryptology in India, Proceedings, Part I. pp. 117–142. Lecture Notes in Computer Science, Springer (2024). https://doi.org/10.1007/978-3-031-80308-6_6 18, 19
10. Dobraunig, C., Matusiewicz, K., Mennink, B., Tereschenko, A.: Efficient instances of docked double decker with AES, and application to authenticated encryption. In: Advances in Cryptology - EUROCRYPT 2025 - 44th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part I. pp. 62–92. Lecture Notes in Computer Science, Springer (2025). https://doi.org/10.1007/978-3-031-91107-1_3 3
11. Durak, F.B., Vaudenay, S.: Breaking the FF3 format-preserving encryption standard over small domains. In: Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Proceedings, Part II. pp. 679–707.

Lecture Notes in Computer Science, Springer (2017). https://doi.org/10.1007/978-3-319-63715-0_23 2
12. Dworkin, M.: Recommendation for block cipher modes of operation. NIST special publication **800**, 38B (2001), https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38G.pdf 2, 3
13. Gunsing, A., Daemen, J., Mennink, B.: Deck-based wide block cipher modes and an exposition of the blinded keyed hashing model. IACR Trans. Symmetric Cryptol. **2019**(4), 1–22 (2019). https://doi.org/10.13154/TOSC.V2019.I4.1-22 23
14. Hoang, V.T., Krovetz, T., Rogaway, P.: Robust authenticated-encryption AEZ and the problem that it solves. In: Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part I. pp. 15–44. Lecture Notes in Computer Science, Springer (2015). https://doi.org/10.1007/978-3-662-46800-5_2 21
15. Hoang, V.T., Miller, D., Trieu, N.: Attacks only get better: How to break FF3 on large domains. In: Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part II. pp. 85–116. Lecture Notes in Computer Science, Springer (2019). https://doi.org/10.1007/978-3-030-17656-3_4 2
16. Hoang, V.T., Tessaro, S., Trieu, N.: The curse of small domains: New attacks on format-preserving encryption. In: Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Proceedings, Part I. pp. 221–251. Lecture Notes in Computer Science, Springer (2018). https://doi.org/10.1007/978-3-319-96884-1_8 2
17. Hosoyamada, A., Iwata, T.: 4-round Luby-Rackoff construction is a qPRP: tight quantum security bound. Cryptology ePrint Archive **2019**, 243 (2019), https://eprint.iacr.org/2019/243 2
18. 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, Proceedings. pp. 391–411. Lecture Notes in Computer Science, Springer (2019). https://doi.org/10.1007/978-3-030-12612-4_20 2, 16, 20
19. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Breaking symmetric cryptosystems using quantum period finding. In: Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Proceedings, Part II. pp. 207–237. Lecture Notes in Computer Science, Springer (2016). https://doi.org/10.1007/978-3-662-53008-5_8 2, 6, 15
20. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Breaking symmetric cryptosystems using quantum period finding. CoRR **abs/1602.05973** (2016), http://arxiv.org/abs/1602.05973 35
21. Kuwakado, H., Morii, M.: Quantum distinguisher between the 3-round Feistel cipher and the random permutation. In: IEEE International Symposium on Information Theory, ISIT 2010, Proceedings. pp. 2682–2685. IEEE (2010). https://doi.org/10.1109/ISIT.2010.5513654 2, 14
22. Li, Z., Fan, S., Gao, F., Hao, Y., Hu, X., Wan, L., Sun, H., Su, Q.: Conditional constant function problem and its quantum solutions: Attacking Feistel ciphers. IEEE Internet Things J. **13**(2), 2127–2143 (2026). https://doi.org/10.1109/JIOT.2025.3627427 20
23. Luby, M., Rackoff, C.: Pseudo-random permutation generators and cryptographic composition. In: Proceedings of the 18th Annual ACM Symposium on Theory of Computing. pp. 356–363. ACM (1986). https://doi.org/10.1145/12130.12167 2

24. 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 2
25. Minematsu, K.: Parallelizable rate-1 authenticated encryption from pseudorandom functions. In: Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques. Proceedings. pp. 275–292. Lecture Notes in Computer Science, Springer (2014). https://doi.org/10.1007/978-3-642-55220-5_16 3
26. Patarin, J.: Pseudorandom permutations based on the D.E.S. scheme. In: International Symposium on Coding Theory and Applications (held in Europe). pp. 193–204. Springer (1990). https://doi.org/10.1007/3-540-54303-1_131 2
27. Schur, J.: Über potenzreihen, die im innern des einheitskreises beschränkt sind. Journal für die reine und angewandte Mathematik **147**, 205–232 (1917), http://eudml.org/doc/149467 41
28. Simon, D.R.: On the power of quantum computation. SIAM J. Comput. **26**(5), 1474–1483 (1997). https://doi.org/10.1137/S0097539796298637 2, 6
29. Wagner, D.A.: The boomerang attack. In: Fast Software Encryption, 6th International Workshop, FSE '99, Proceedings. pp. 156–170. Lecture Notes in Computer Science, Springer (1999). https://doi.org/10.1007/3-540-48519-8_12 7
30. Wang, P., Mao, S., Xu, R., Jing, J., Wang, Y.: How to recover the full plaintext of XCB. In: Advances in Cryptology - CRYPTO 2025 - 45th Annual International Cryptology Conference, Proceedings, Part V. pp. 230–259. Lecture Notes in Computer Science, Springer (2025). https://doi.org/10.1007/978-3-032-01901-1_8 3, 4
31. Wang, X., Xiang, Z., Zhang, S., Chen, S., Zeng, X.: Quantum chosen-ciphertext attacks based on Simon's algorithm against unified structures. In: Topics in Cryptology - CT-RSA 2025 - Cryptographers' Track at the RSA Conference 2025, Proceedings. pp. 99–122. Lecture Notes in Computer Science, Springer (2025). https://doi.org/10.1007/978-3-031-88661-4_5 2, 7
32. Xiang, Z., Wang, X., Yu, B., Sun, B., Zhang, S., Zeng, X., Shen, X., Li, N.: Links between quantum distinguishers based on Simon's algorithm and truncated differentials. IACR Trans. Symmetric Cryptol. **2024**(2), 296–321 (2024). https://doi.org/10.46586/TOSC.V2024.I2.296-321 2

A On the Relation Between Plaintext Recovery and Distinguishing Attacks

The following counterexample shows that distinguishing attacks do not generically imply plaintext recovery attacks. Hence, in general, plaintext recovery is strictly stronger than distinguishing.

Let π be a random permutation on n bits. Suppose $\pi(0^n) = b$ and $\pi(a) = 0^n$. Define a new permutation π' that is identical to π except that the images of 0^n and a are swapped, namely,

$$\pi'(x) = \begin{cases} \pi(x), & \text{if } x \neq 0^n, a, \\ 0^n, & \text{if } x = 0^n, \\ b, & \text{if } x = a. \end{cases}$$

π' is distinguishable but plaintext-recovery secure under CPA.

– A single encryption query suffices to distinguish π' from a random permutation with high success probability under CPA. Query the plaintext 0^n and obtain the ciphertext C. For π', we always have $C = 0^n$. By contrast, for a uniformly random permutation, the event $C = 0^n$ occurs with probability only 2^{-n}. Hence, by checking whether $C = 0^n$, one distinguishes π' from a random permutation with probability close to 1.
– However, π' satisfies plaintext recovery security under CPA. Let C^* be the target ciphertext corresponding to a random plaintext $P^* \leftarrow\!\$\ \{0,1\}^n$. The probability that $P^* = 0^n$ or $P^* = a$ is negligible, so we may assume $P^* \notin \{0^n, a\}$ and therefore $C^* \notin \{0^n, b\}$. For any input P distinct from 0^n and a, we have $\pi'(P) = \pi(P)$. Thus, for any $P \neq P^*$, the value $\pi'(P)$ is independent of C^*. Consequently, the only generic way to recover P^* using encryption queries is exhaustive search over the plaintext space, querying π' on each candidate and checking whether the output matches C^*. This requires $\mathcal{O}(2^n)$ queries.

π' is distinguishable but plaintext-recovery secure under CCA.

– The CPA distinguishing attack above immediately yields a CCA distinguishing attack.
– Nevertheless, π' still satisfies plaintext recovery security under CCA. Let C^* be the target ciphertext corresponding to a random plaintext $P^* \leftarrow\!\$\ \{0,1\}^n$, and again assume $P^* \notin \{0^n, a\}$ and $C^* \notin \{0^n, b\}$. For any plaintext query $P \neq P^*$, the ciphertext $\pi'(P)$ is independent of C^*. For any decryption query $C \neq C^*$, the plaintext $\pi'^{-1}(C)$ is independent of P^*. Hence, encryption and decryption queries can at best eliminate candidates from the plaintext space: an encryption query rules out the queried plaintext unless its ciphertext equals C^*, while a decryption query rules out the returned plaintext unless the queried ciphertext is the challenge ciphertext, which is forbidden. Therefore, recovering P^* still requires exhaustive search over the plaintext space, with complexity $\mathcal{O}(2^n)$.

B Proof of Theorem 2

Simon's algorithm repeats the following five quantum steps.

(1) Starting with a $(n+d)$-qubit state $|0^n\rangle|0^d\rangle$, one applies a Hadamard transform $H^{\otimes n}$ to the first register to obtain the quantum superposition

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle|0^d\rangle.$$

(2) A quantum query to the function $\mathring{f}^{\{x^*\}}$ maps this to the state
$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle|\mathring{f}^{\{x^*\}}(x)\rangle$
$= \frac{1}{\sqrt{2^n}}|x^*\rangle|\perp\rangle +$
$\frac{1}{\sqrt{2^n}} \left(|x^* \oplus s\rangle + \sum_{t:t\in \mathrm{Coll}_{f(x^*)}} (|x^* \oplus t\rangle + |x^* \oplus s \oplus t\rangle) \right) |f(x^*)\rangle +$
$\frac{1}{\sqrt{2^n}} \sum_{f(x):f(x)\neq f(x^*)} \left(|x\rangle + |x \oplus s\rangle + \sum_{t:t\in \mathrm{Coll}_{f(x)}} (|x \oplus t\rangle + |x \oplus s \oplus t\rangle) \right) |f(x)\rangle,$
where for each x, $\mathrm{Coll}_{f(x)}$ is a set of representatives such that $\mathrm{Coll}_{f(x)}$ and its coset $\{t \oplus s : t \in \mathrm{Coll}_{f(x)}\}$ partition the set $\{\tau \in \{0,1\}^n \setminus \{0^n, s\} : f(x \oplus \tau) = f(x)\}$.
(3) Measuring the second register in the computational basis yields a value v and collapses the first register to a state $|\varphi\rangle$.
(4) Applying again the Hadamard transform $H^{\otimes n}$ to the first register gives $|\psi\rangle$.
(5) Measuring the state in the computational basis yields a vector y.

Now we analyze separately for each distinct value v measured in step (3).

Case 1: measured value $v = \perp$. This happens with probability $\frac{1}{2^n}$. In this case, the first register collapses to the state

$$|\varphi\rangle = |x^*\rangle.$$

Then in step (4) we apply $H^{\otimes n}$ to $|\varphi\rangle$ and obtain

$$\begin{aligned} |\psi\rangle =& \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} (-1)^{x^*\cdot y}|y\rangle \\ =& \frac{1}{\sqrt{2^n}} \sum_{y:y\cdot s=0} (-1)^{x^*\cdot y}|y\rangle + \frac{1}{\sqrt{2^n}} \sum_{y:y\cdot s=1} (-1)^{x^*\cdot y}|y\rangle. \end{aligned}$$

Step (5) measures y. If $y \cdot s = 1$, the subroutine fails. This event has probability

$$|\{y : y \cdot s = 1\}| \cdot \left(\frac{(-1)^{x^*\cdot y}}{\sqrt{2^n}} \right)^2 = 2^{n-1} \cdot \frac{1}{2^n} = \frac{1}{2}.$$

Therefore, when the measurement in step (3) yields $v = \perp$, the subroutine of Simon's algorithm fails with probability $\frac{1}{2} \cdot \frac{1}{2^n} = \frac{1}{2^{n+1}}$.

Case 2: measured value $v = f(x^*)$. This occurs with probability $\frac{\ell}{2^n}$ where $\ell \coloneqq 1 + 2|\,\mathrm{Coll}_{f(x^*)}\,|$. In this case, the first register collapses to

$$|\varphi\rangle = \frac{1}{\sqrt{\ell}} \left(|x^* \oplus s\rangle + \sum_{t\in \mathrm{Coll}_{f(x^*)}} (|x^* \oplus t\rangle + |x^* \oplus s \oplus t\rangle) \right).$$

Then step (4) applies $H^{\otimes n}$ to $|\varphi\rangle$ and yields

$$|\psi\rangle = \frac{1}{\sqrt{\ell \cdot 2^n}} \sum_{y\in\{0,1\}^n} \left[(-1)^{(x^* \oplus s)\cdot y} + \sum_{t\in \mathrm{Coll}_{f(x^*)}} (-1)^{(x^*\oplus t)\cdot y}(1+(-1)^{s\cdot y}) \right] |y\rangle.$$

Step (5) measures y. If $y \cdot s = 1$, the subroutine fails. Note that $y \cdot s = 1$ makes the factor $1+(-1)^{s\cdot y}$ in the latter part of the amplitude vanish. Then this event has probability

$$|\{y : y\cdot s = 1\}| \cdot \left(\frac{(-1)^{(x^*\oplus s)\cdot y}}{\sqrt{\ell \cdot 2^n}} \right)^2 = 2^{n-1} \cdot \frac{1}{\ell \cdot 2^n} = \frac{1}{\ell \cdot 2}.$$

Hence when step (3) measures $v = f(x^*)$, the subroutine of Simon's algorithm fails with probability $\frac{1}{\ell\cdot 2} \cdot \frac{\ell}{2^n} = \frac{1}{2^{n+1}}$.

Case 3: measured value $v \notin \{\perp, f(x^*)\}$, which occurs with probability $\frac{r}{2^n}$ where $r \coloneqq 2 + 2|\,\mathrm{Coll}_{f(x)}\,|$. The first register collapses to

$$|\varphi\rangle = \frac{1}{\sqrt{r}} \left(|x\rangle + |x\oplus s\rangle + \sum_{t\in\mathrm{Coll}_{f(x)}} (|x\oplus t\rangle + |x\oplus s\oplus t\rangle) \right).$$

Then step (4) applies $H^{\otimes n}$ to $|\varphi\rangle$ and yields

$$|\psi\rangle = \frac{1}{\sqrt{r\cdot 2^n}} \sum_{y\in\{0,1\}^n} (-1)^{x\cdot y}(1+(-1)^{s\cdot y}) \sum_{t\in \mathrm{Coll}_{f(x)}\cup\{0^n\}} (-1)^{t\cdot y}|y\rangle.$$

If $y\cdot s = 1$, the factor $1+(-1)^{s\cdot y}$ in the amplitude vanishes, so any measured y must satisfy $y\cdot s = 0$. However, the measured y may also satisfy $y\cdot t = 0$ for some $t \in \{0,1\}^n \backslash \{0^n, s\}$. For example, if $\mathrm{Coll}_{f(x)} = \{t\}$ with $t \in \{0,1\}^n\backslash\{0^n, s\}$. Then the expression becomes

$$|\psi\rangle = \frac{1}{\sqrt{r\cdot 2^n}} \sum_{y\in\{0,1\}^n} (-1)^{x\cdot y}(1+(-1)^{s\cdot y})(1+(-1)^{t\cdot y})|y\rangle.$$

In this case, any measured y must satisfy both $y\cdot s = 0$ and $y\cdot t = 0$.

We now compute the failure probability of Simon's algorithm after cn subroutine executions. Let the outputs be $y_1, y_2, \ldots, y_{cn}$. Two events can cause failure.

Event A: some y_i satisfies $y_i \cdot s = 1$.

Assuming event A does not occur, the following event may still lead to failure.

Event B: $\dim(\mathrm{Span}\{y_1,\ldots,y_{cn}\}) \le n-2$.

We analyze the probabilities of these events separately.

Event A can occur only in Case 1 and Case 2. From the analysis of those cases, the probability of event A is

$$\Pr[A] \le \sum_{i=1}^{cn} \left(\frac{1}{2^{n+1}} + \frac{1}{2^{n+1}} \right) = \frac{cn}{2^n}.$$

We next analyze the probability of event B. Following the analysis in [20, Appendix A], the equation

$$\Pr[B] \leq \max_{t \in \{0,1\}^n \setminus \{0^n, s\}} \left[2 \left(\begin{array}{l} \frac{1}{2^{n+1}} \cdot \sum_{x \in \{0,1\}^n} \langle \mathring{f}^{\{x^*\}}(x) \mid \mathring{f}^{\{x^*\}}(x) \rangle + \\ \frac{1}{2^{n+1}} \cdot \sum_{x \in \{0,1\}^n} \langle \mathring{f}^{\{x^*\}}(x \oplus t) \mid \mathring{f}^{\{x^*\}}(x) \rangle \end{array} \right)^c \right]^n$$

holds. Based on this, it follows that

$$\begin{aligned} \Pr[B] &\leq \max_{t \in \{0,1\}^n \setminus \{0^n, s\}} \left[2 \left(\frac{1}{2} + \frac{1}{2^{n+1}} \cdot \sum_{x \in \{0,1\}^n} \langle f(x \oplus t) \mid f(x) \rangle \right)^c \right]^n \\ &\leq \left(2 \left(\frac{1 + \varepsilon(f)}{2} \right)^c \right)^n . \end{aligned}$$

In summary, Simon's algorithm succeeds with probability at least

$$1 - \frac{cn}{2^n} - \left(2 \left(\frac{1 + \varepsilon(f)}{2} \right)^c \right)^n .$$

C Formal Definitions for Plaintext Recovery Attacks

In this part, we formally define plaintext recovery attacks against a block cipher under (quantum) chosen-plaintext and (quantum) chosen-ciphertext attack models.

$\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}cpa}}_{\mathbf{E}}$	$\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}cca}}_{\mathbf{E}}$
$K \leftarrow\!\!\$\ \mathcal{K}$	$K \leftarrow\!\!\$\ \mathcal{K}$
$P^* \leftarrow\!\!\$\ \mathcal{P}$	$P^* \leftarrow\!\!\$\ \mathcal{P}$
$C^* \leftarrow \mathbf{E}_K(P^*)$	$C^* \leftarrow \mathbf{E}_K(P^*)$
$P \leftarrow \mathcal{A}^{\textsc{Enc}}(C^*)$	$P \leftarrow \mathcal{A}^{\textsc{Enc},\textsc{Dec}}(C^*)$
if $P = P^*$ **then**	**if** $P = P^*$ **then**
return 1	**return** 1
else	**else**
return 0	**return** 0
proc $\textsc{Enc}(P)$	**proc** $\textsc{Dec}(C)$
$C \leftarrow \mathbf{E}_K(P)$	$P \leftarrow \mathring{\mathbf{D}}_K^{\{C^*\}}(C)$
return C	**return** P

Fig. 15: The experiments $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}cpa}}_{\mathbf{E}}$ and $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}cca}}_{\mathbf{E}}$, where $\mathring{\mathbf{D}}_K^{\{C^*\}}$ denotes the decryption algorithm punctured at C^*, as defined in Eq. (1).

Definition 1 (PR-CPA/CCA). *Let* $\mathbf{E}$ *be a block cipher over key space* $\mathcal{K}$*, plaintext space* $\mathcal{P}$ *and ciphertext space* $\mathcal{C}$*, and let* $\mathbf{D} = \mathbf{E}^{-1}$ *denote its inverse. For any probabilistic polynomial-time adversary* $\mathcal{A}$*, we define its advantage for plaintext recovery under chosen-plaintext attack (PR-CPA) and chosen-ciphertext attack (PR-CCA) against* $\mathbf{E}$ *as*

$$\mathbf{Adv}^{\mathit{pr\text{-}cpa}}_{\mathbf{E}}(\mathcal{A}) \coloneqq \Pr\left[\mathrm{Exp}^{\mathcal{A},\mathit{pr\text{-}cpa}}_{\mathbf{E}} \Rightarrow 1\right],$$
$$\mathbf{Adv}^{\mathit{pr\text{-}cca}}_{\mathbf{E}}(\mathcal{A}) \coloneqq \Pr\left[\mathrm{Exp}^{\mathcal{A},\mathit{pr\text{-}cca}}_{\mathbf{E}} \Rightarrow 1\right].$$

Here, $\mathrm{Exp}^{\mathcal{A},\mathit{pr\text{-}cpa}}_{\mathbf{E}}$ *and* $\mathrm{Exp}^{\mathcal{A},\mathit{pr\text{-}cca}}_{\mathbf{E}}$ *are defined as in Fig. 15. In the experiment* $\mathrm{Exp}^{\mathcal{A},\mathit{pr\text{-}cpa}}_{\mathbf{E}}$*, the adversary* $\mathcal{A}$ *is allowed to make at most* q *encryption queries. In the experiment* $\mathrm{Exp}^{\mathcal{A},\mathit{pr\text{-}cca}}_{\mathbf{E}}$*, the adversary* $\mathcal{A}$ *is allowed to make at most* q *queries in total, including both encryption and decryption queries.*

$\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcpa}}_{\mathbf{E}}$

$K \leftarrow\!\!\$\ \mathcal{K}$
$P^* \leftarrow\!\!\$\ \mathcal{P}$
$C^* \leftarrow \mathbf{E}_K(P^*)$
$P \leftarrow \mathcal{A}^{\mathrm{qEnc}}(C^*)$
if $P = P^*$ **then**
  return 1
else
  return 0

$\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcca}}_{\mathbf{E}}$

$K \leftarrow\!\!\$\ \mathcal{K}$
$P^* \leftarrow\!\!\$\ \mathcal{P}$
$C^* \leftarrow \mathbf{E}_K(P^*)$
$P \leftarrow \mathcal{A}^{\mathrm{qEnc},\mathrm{qDec}}(C^*)$
if $P = P^*$ **then**
  return 1
else
  return 0

proc $\mathrm{qEnc}\left(\sum_{P,S} \alpha_{P,S}|P\rangle|S\rangle\right)$

$$\sum_{P,S} \alpha_{P,S}|P\rangle|S\rangle \leftarrow \sum_{P,S} \alpha_{P,S}|P\rangle|S \oplus \mathbf{E}_K(P)\rangle$$

return $\displaystyle\sum_{P,S} \alpha_{P,S}|P\rangle|S\rangle$

proc $\mathrm{qDec}\left(\sum_{C,S} \alpha_{C,S}|C\rangle|S\rangle\right)$

$$\sum_{C,S} \alpha_{C,S}|C\rangle|S\rangle \leftarrow \sum_{C,S} \alpha_{C,S}|C\rangle|S \oplus \mathring{\mathbf{D}}^{\{C^*\}}_K(C)\rangle$$

return $\displaystyle\sum_{C,S} \alpha_{C,S}|C\rangle|S\rangle$

Fig. 16: The experiments $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcpa}}_{\mathbf{E}}$ and $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcca}}_{\mathbf{E}}$, where $\mathring{\mathbf{D}}^{\{C^*\}}_K$ denotes the decryption algorithm punctured at C^*, as defined in Eq. (1).

Definition 2 (PR-qCPA/qCCA). *Let* $\mathbf{E}$ *be a block cipher over key space* $\mathcal{K}$*, plaintext space* $\mathcal{P}$ *and ciphertext space* $\mathcal{C}$*, and let* $\mathbf{D} = \mathbf{E}^{-1}$ *denote its inverse. For any probabilistic polynomial-time quantum adversary* $\mathcal{A}$*, we define its advantage for plaintext recovery under quantum chosen-plaintext attack (PR-qCPA) and quantum chosen-ciphertext attack (PR-qCCA) against* $\mathbf{E}$ *as*

$$\mathbf{Adv}^{\mathsf{pr\text{-}qcpa}}_{\mathbf{E}}(\mathcal{A}) \coloneqq \Pr\left[\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcpa}}_{\mathbf{E}} \Rightarrow 1\right],$$
$$\mathbf{Adv}^{\mathsf{pr\text{-}qcca}}_{\mathbf{E}}(\mathcal{A}) \coloneqq \Pr\left[\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcca}}_{\mathbf{E}} \Rightarrow 1\right].$$

Here, $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcpa}}_{\mathbf{E}}$ *and* $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcca}}_{\mathbf{E}}$ *are defined as in Fig. 16. In the experiment* $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcpa}}_{\mathbf{E}}$*, the adversary* $\mathcal{A}$ *is allowed to make at most* q *quantum encryption queries. In the experiment* $\mathrm{Exp}^{\mathcal{A},\mathsf{pr\text{-}qcca}}_{\mathbf{E}}$*, the adversary* $\mathcal{A}$ *is allowed to make at most* q *quantum queries in total, including both quantum encryption and quantum decryption queries.*

D Proof of $\varepsilon(f) \leq 1/2$ for 4-Round Feistel Structure

The function f in Eq. (14) is described as

$$f(\beta, x) = \alpha_0 \oplus \alpha_1 \oplus F_2(F_1(\alpha_\beta) \oplus x) \oplus F_2 \begin{bmatrix} F_1(\alpha_\beta) \oplus x \oplus \\ F_3\left[\alpha_\beta \oplus F_2(F_1(\alpha_\beta) \oplus x)\right] \oplus \\ F_3\left[\alpha_{\beta\oplus 1} \oplus F_2(F_1(\alpha_\beta) \oplus x)\right] \end{bmatrix},$$

where $\beta \in \{0,1\}, x \in \{0,1\}^n$ and α_0, α_1 are two arbitrary different fixed elements in $\{0,1\}^n$. Now we examine whether $\varepsilon(f) \leq \frac{1}{2}$. Let

$$e(\beta, x) \coloneqq F_1(\alpha_\beta) \oplus x,$$
$$p(\beta, x) \coloneqq e(\beta, x) \oplus F_3\left[\alpha_\beta \oplus F_2(e(\beta, x))\right] \oplus F_3\left[\alpha_{\beta\oplus 1} \oplus F_2(e(\beta, x))\right].$$

For any $(t_1, t_2) \in \{0,1\} \times \{0,1\}^n \setminus \{(0, 0^n), (1, s)\}$, it follows that $e(\beta, x) \neq e(\beta \oplus t_1, x \oplus t_2)$ and

$$\begin{aligned} & f(\beta, x) = f(\beta \oplus t_1, x \oplus t_2) \\ \Leftrightarrow & F_2(e(\beta, x)) \oplus F_2(p(\beta, x)) = F_2(e(\beta \oplus t_1, x \oplus t_2)) \oplus F_2(p(\beta \oplus t_1, x \oplus t_2)). \end{aligned} \tag{20}$$

We consider the following cases:

(1) $p(\beta, x) \notin \{e(\beta, x), e(\beta \oplus t_1, x \oplus t_2), p(\beta \oplus t_1, x \oplus t_2)\}$: In this case, $F_2(p(\beta, x))$ is a new sample. By the randomness of F_2, Eq. (20) holds with probability $\frac{1}{2^n}$.

(2) $p(\beta, x) = e(\beta, x)$: This condition is equivalent to

$$F_3\left[\alpha_\beta \oplus F_2(e(\beta, x))\right] \oplus F_3\left[\alpha_{\beta\oplus 1} \oplus F_2(e(\beta, x))\right] = 0^n.$$

By the randomness of F_3, this case happens with probability $\frac{1}{2^n}$.

(3) $p(\beta, x) = e(\beta \oplus t_1, x \oplus t_2)$: This condition is equivalent to

$$F_3\left[\alpha_\beta \oplus F_2(e(\beta, x))\right] \oplus F_3\left[\alpha_{\beta\oplus 1} \oplus F_2(e(\beta, x))\right] = e(\beta, x) \oplus e(\beta \oplus t_1, x \oplus t_2).$$

By the randomness of F_3, this case happens with probability $\frac{1}{2^n}$.

(4) $p(\beta, x) = p(\beta \oplus t_1, x \oplus t_2)$: This condition is equivalent to

$$\begin{aligned} &F_3\left[\alpha_\beta \oplus F_2(e(\beta, x))\right] \oplus F_3\left[\alpha_{\beta\oplus 1} \oplus F_2(e(\beta, x))\right] \oplus \\ &F_3\left[\alpha_{\beta\oplus t_1} \oplus F_2(e(\beta \oplus t_1, x \oplus t_2))\right] \oplus F_3\left[\alpha_{\beta\oplus t_1\oplus 1} \oplus F_2(e(\beta \oplus t_1, x \oplus t_2))\right] \\ =&e(\beta, x) \oplus e(\beta \oplus t_1, x \oplus t_2). \end{aligned} \tag{21}$$

Further, we consider the following cases:

1) $F_2(e(\beta, x)) \notin \{F_2(e(\beta\oplus t_1, x\oplus t_2)), F_2(e(\beta\oplus t_1, x\oplus t_2))\oplus\alpha_0\oplus\alpha_1\}$: In this case, the four inputs of F_3 in Eq. (21) are distinct. By the randomness of F_3, Eq. (21) holds with probability $\frac{1}{2^n}$.
2) $F_2(e(\beta, x)) \in \{F_2(e(\beta \oplus t_1, x \oplus t_2)), F_2(e(\beta \oplus t_1, x \oplus t_2)) \oplus \alpha_0 \oplus \alpha_1\}$: In this case, Eq. (21) reduces to $0^n = e(\beta, x) \oplus e(\beta \oplus t_1, x \oplus t_2)$, which holds with probability 0.

Therefore,

$$\Pr_{\beta,x}\left[f(\beta, x) = f(\beta \oplus t_1, x \oplus t_2)\right] \le \frac{1}{2^n} + \frac{1}{2^n} + \frac{1}{2^n} + \frac{1}{2^n} = \frac{4}{2^n} \le \frac{1}{2}$$

for $n \ge 3$. So we obtain $\varepsilon(f) \le \frac{1}{2}$.

E Plaintext Recovery Attacks on Feistel: From 2-Round to 4-Round

E.1 Full Plaintext Recovery Attack on 2-Round Feistel under CCA

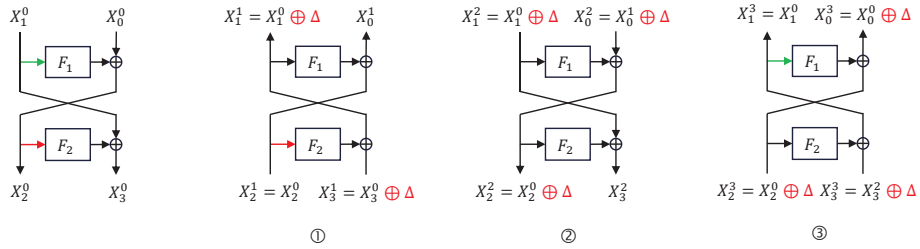


Fig. 17: The process of $\mathring{\mathbf{D}}_{[2]}^{\{C^0\}}$, $\mathbf{E}_{[2]}$, and $\mathring{\mathbf{D}}_{[2]}^{\{C^0\}}$ (Query 1 to Query 3). The values in the red (green) branches are identical.

In this section, we present a full plaintext recovery attack against 2-round Feistel. It performs 3 classical queries by alternating between decryption and encryption oracles.

Attack steps. Given an arbitrary ciphertext $C^0 = (X_2^0, X_3^0) = \mathbf{E}_{[2]}(X_1^0, X_0^0)$, the attacker recovers the corresponding plaintext (X_1^0, X_0^0) through the following steps. The process is illustrated in Fig. 17.

(1) Fix an arbitrary value Δ in $\{0,1\}^n \setminus \{0^n\}$.
(2) Query the decryption oracle with ciphertext $(X_2^1, X_3^1) = \mathbf{T}_{(0^n,\Delta)}(X_2^0, X_3^0)$ and obtain $(X_1^1, X_0^1) = \mathring{\mathbf{D}}_{[2]}^{\{C^0\}}(X_2^1, X_3^1)$.
(3) Query the encryption oracle with plaintext $(X_1^2, X_0^2) = \mathbf{T}_{(0^n,\Delta)}(X_1^1, X_0^1)$, and obtain $(X_2^2, X_3^2) = \mathbf{E}_{[2]}(X_1^2, X_0^2)$.
(4) Query the decryption oracle with ciphertext $(X_2^3, X_3^3) = \mathbf{T}_{(0^n,\Delta)}(X_2^2, X_3^2)$ and obtain $(X_1^3, X_0^3) = \mathring{\mathbf{D}}_{[2]}^{\{C^0\}}(X_2^3, X_3^3)$.
(5) Compute the entire target plaintext $(X_1^0, X_0^0) = \mathbf{T}_{(0^n,\Delta)}(X_1^3, X_0^3)$.

Correctness. The correctness of the above attack follows from the following proposition.

Proposition 5. *Let Δ be an arbitrary element in $\{0,1\}^n \setminus \{0^n\}$. For an arbitrary ciphertext $C^0 = (X_2^0, X_3^0) = \mathbf{E}_{[2]}(X_1^0, X_0^0)$,*

$$\left(\mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[2]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathbf{E}_{[2]} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[2]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)}\right)(X_2^0, X_3^0) = (X_1^0, X_0^0).$$

Proof. The first decryption query uses input $(X_2^1, X_3^1) = \mathbf{T}_{(0^n,\Delta)}(X_2^0, X_3^0)$. Since $\Delta \neq 0^n$, this ciphertext differs from the target ciphertext (X_2^0, X_3^0); hence $\mathring{\mathbf{D}}_{[2]}^{\{C^0\}}(X_2^1, X_3^1) = \mathbf{D}_{[2]}(X_2^1, X_3^1) \neq \perp$. Moreover, the transformation $\mathbf{T}_{(0^n,\Delta)}$ from (X_2^0, X_3^0) to (X_2^1, X_3^1) ensures that its output satisfies $X_1^1 = X_1^0 \oplus \Delta$.

The second encryption query input $(X_1^2, X_0^2) = \mathbf{T}_{(0^n,\Delta)}(X_1^1, X_0^1)$ ensures that its output satisfies $X_2^2 = X_2^1 \oplus \Delta = X_2^0 \oplus \Delta$, given that the input satisfies $X_1^2 = X_1^1 = X_1^0 \oplus \Delta$.

The third decryption query uses input $(X_2^3, X_3^3) = \mathbf{T}_{(0^n,\Delta)}(X_2^2, X_3^2)$. Since $X_2^3 = X_2^2 = X_2^0 \oplus \Delta \neq X_2^0$, the ciphertext (X_2^3, X_3^3) differs from the target ciphertext (X_2^0, X_3^0); hence $\mathring{\mathbf{D}}_{[2]}^{\{C^0\}}(X_2^3, X_3^3) = \mathbf{D}_{[2]}(X_2^3, X_3^3) \neq \perp$. Moreover, the transformation $\mathbf{T}_{(0^n,\Delta)}$ from (X_2^2, X_3^2) to (X_2^3, X_3^3) ensures that its output satisfies $X_1^3 = X_1^2 \oplus \Delta = X_1^0$. Combining this with the previous conclusion $X_2^3 = X_2^0 \oplus \Delta$, we further deduce that $X_0^3 = X_0^0 \oplus \Delta$. Consequently, $\mathbf{T}_{(0^n,\Delta)}(X_1^3, X_0^3) = (X_1^0, X_0^0)$.

Hence,

$$\left(\mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[2]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathbf{E}_{[2]} \circ \mathbf{T}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}_{[2]}^{\{C^0\}} \circ \mathbf{T}_{(0^n,\Delta)}\right)(X_2^0, X_3^0) = (X_1^0, X_0^0).$$

□

Success probability and complexity analysis. From the above analysis, it is clear that this plaintext recovery attack on the 2-round Feistel structure succeeds with probability 1, using only 3 classical encryption and decryption queries.

E.2 Full Plaintext Recovery Attack on 4-Round Feistel under qCCA

We extend the plaintext recovery attack against 2-round Feistel presented in Section E.1 to 4-round Feistel using both the forward and the backward single-round extension methods in Section 4.1. This attack makes $\mathcal{O}(n)$ quantum encryption and decryption queries.

Attack steps. Given an arbitrary ciphertext $C^0 = (X_4^0, X_5^0) = \mathbf{E}_{[4]}(X_1^0, X_0^0)$, the attacker recovers the corresponding plaintext (X_1^0, X_0^0) through the following steps. The process of 3 classical queries is illustrated in Fig. 18.

(1) Fix an arbitrary value Δ in $\{0,1\}^n \setminus \{0^n\}$.
(2) Recover the secret state $s_1 \coloneqq F_4(X_4^0) \oplus F_4(X_4^0 \oplus \Delta)$.
(3) Query the decryption oracle with ciphertext $(X_4^1, X_5^1) = \mathbf{T}^{\mathsf{b},s_1}_{(0^n,\Delta)}(X_4^0, X_5^0)$ and obtain $(X_1^1, X_0^1) = \mathring{\mathbf{D}}^{\{C^0\}}_{[4]}(X_4^1, X_5^1)$.
(4) Recover the secret state $s_2 \coloneqq F_1(X_1^1) \oplus F_1(X_1^1 \oplus \Delta)$.
(5) Query the encryption oracle with plaintext $(X_1^2, X_0^2) = \mathbf{T}^{\mathsf{f},s_2}_{(0^n,\Delta)}(X_1^1, X_0^1)$, and obtain $(X_4^2, X_5^2) = \mathbf{E}_{[4]}(X_1^2, X_0^2)$.
(6) Recover the secret state $s_3 \coloneqq F_4(X_4^2) \oplus F_4(X_4^2 \oplus \Delta)$.
(7) Query the decryption oracle with ciphertext $(X_4^3, X_5^3) = \mathbf{T}^{\mathsf{b},s_3}_{(0^n,\Delta)}(X_4^2, X_5^2)$ and obtain $(X_1^3, X_0^3) = \mathring{\mathbf{D}}^{\{C^0\}}_{[4]}(X_4^3, X_5^3)$.
(8) Recover the secret state $s_4 \coloneqq F_1(X_1^3) \oplus F_1(X_1^3 \oplus \Delta)$.
(9) Compute the entire target plaintext $(X_1^0, X_0^0) = \mathbf{T}^{\mathsf{f},s_4}_{(0^n,\Delta)}(X_1^3, X_0^3)$.

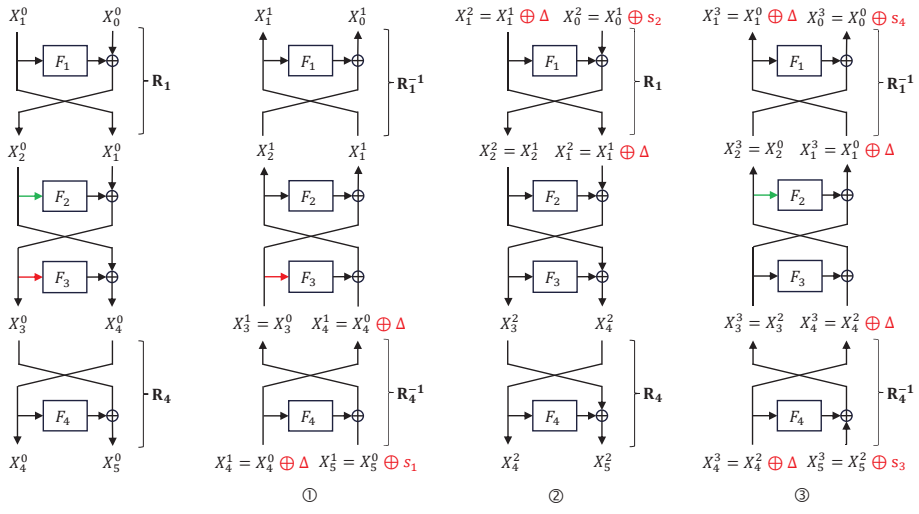


Fig. 18: The process of $\mathring{\mathbf{D}}^{\{C^0\}}_{[4]}$, $\mathbf{E}_{[4]}$, and $\mathring{\mathbf{D}}^{\{C^0\}}_{[4]}$ (Query 1 to Query 3). The values in the red (green) branches are identical.

Given that $\Delta \neq 0^n$, it is feasible to recover s_2 and s_4 (resp. s_1 and s_3) by running Simon's algorithm using quantum queries to the (punctured) periodic function defined in Eq. (15) (resp. Eq. (17)).

Correctness. Building on Proposition 5, we can directly obtain the following proposition.

Proposition 6. *Let Δ be an arbitrary element in $\{0,1\}^n \backslash \{0^n\}$. For an arbitrary ciphertext $C^0 = (X_4^0, X_5^0) = \mathbf{E}_{[4]}(X_1^0, X_0^0)$,*

$$\left(\mathbf{T}^{\mathsf{f},s_4}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}^{\{C^0\}}_{[4]} \circ \mathbf{T}^{\mathsf{b},s_3}_{(0^n,\Delta)} \circ \mathbf{E}_{[4]} \circ \mathbf{T}^{\mathsf{f},s_2}_{(0^n,\Delta)} \circ \mathring{\mathbf{D}}^{\{C^0\}}_{[4]} \circ \mathbf{T}^{\mathsf{b},s_1}_{(0^n,\Delta)}\right)(X_4^0, X_5^0) = (X_1^0, X_0^0),$$

where

$$\begin{aligned}
&(X_1^i, X_0^i) = \mathring{\mathbf{D}}^{\{C^0\}}_{[4]} \circ \mathbf{T}^{\mathsf{b},s_i}_{(0^n,\Delta)}(X_4^{i-1}, X_5^{i-1}), i = 1, 3,\\
&(X_4^2, X_5^2) = \mathbf{E}_{[4]} \circ \mathbf{T}^{\mathsf{f},s_2}_{(0^n,\Delta)}(X_1^1, X_0^1),\\
&s_i = F_4(X_4^{i-1}) \oplus F_4(X_4^{i-1} \oplus \Delta), i = 1, 3,\\
&s_j = F_1(X_1^{j-1}) \oplus F_1(X_1^{j-1} \oplus \Delta), j = 2, 4.
\end{aligned}$$

Success probability and complexity analysis. The analysis of the success probability and complexity follows from the underlying processes of Simon's algorithm. This algorithm is also employed in the plaintext recovery attacks against 4-round Feistel via the forward and backward extension methods in Section 4.3. Therefore, the success probability of our attack equals the product of those of the plaintext recovery attacks in Section 4.3, while its complexity equals the sum. Consequently, our attack also succeeds with probability exponentially close to 1, and makes $\mathcal{O}(n)$ quantum queries to both $\mathbf{E}_{[4]}$ and $\mathring{\mathbf{D}}^{\{C^0\}}_{[4]}$ along with $\mathcal{O}(n)$ qubits.

F Equivalence Between the Invertibility of A_{12} and B_{12}

To begin, we derive the invertibility of B_{12} from the invertibility of A_{12}. Let I_n and 0_n denote the $n \times n$ identity and zero matrices, respectively. By referring to the Schur determinant formula [27], we proceed as follows:

$$\begin{aligned}
\begin{pmatrix} A_{12}^{-1} & 0_n \\ A_{22}A_{12}^{-1} & I_n \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} &= \begin{pmatrix} A_{12}^{-1}A_{11} & I_n \\ A_{21} \oplus A_{22}A_{12}^{-1}A_{11} & 0_n \end{pmatrix} \\
&= \begin{pmatrix} I_n & A_{12}^{-1}A_{11} \\ 0_n & A_{21} \oplus A_{22}A_{12}^{-1}A_{11} \end{pmatrix} \begin{pmatrix} 0_n & I_n \\ I_n & 0_n \end{pmatrix}.
\end{aligned}$$

Taking determinants yields $\det(A_{12}^{-1})\det(\varphi) = \det(A_{21} \oplus A_{22}A_{12}^{-1}A_{11}) \cdot 1$. The fact that A_{12} and φ are both invertible implies $\det(A_{12}^{-1}) \neq 0$ and $\det(\varphi) \neq 0$.

Thus $\det(A_{21} \oplus A_{22}A_{12}^{-1}A_{11}) \neq 0$ as well, which implies $A_{21} \oplus A_{22}A_{12}^{-1}A_{11}$ is invertible. From the equation $\varphi\varphi^{-1} = I_{2n}$, we have:

$$\begin{cases} A_{11}B_{12} \oplus A_{12}B_{22} = 0_n, \\ A_{21}B_{12} \oplus A_{22}B_{22} = I_n. \end{cases}$$

Solving these yields $B_{12} = (A_{21} \oplus A_{22}A_{12}^{-1}A_{11})^{-1}$, which implies B_{12} is invertible.

By interchanging φ and φ^{-1}, we analogously deduce the invertibility of A_{12} from that of B_{12}.