Improved quantum attack on Type-1 Generalized Feistel Schemes and Its application to CAST-256

Boyu Ni[1] and Xiaoyang Dong[2]

[1] Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, School of Mathematics, Shandong University, Jinan 250100, China.
[2] Institute for Advanced Study, Tsinghua University, Beijing 100084, China.
xiaoyangdong@tsinghua.edu.cn

Abstract. Generalized Feistel Schemes (GFS) are important components of symmetric ciphers, which have been extensively researched in classical setting. However, the security evaluations of GFS in quantum setting are rather scanty.

In this paper, we give more improved polynomial-time quantum distinguishers on Type-1 GFS in quantum chosen-plaintext attack (qCPA) setting and quantum chosen-ciphertext attack (qCCA) setting. In qCPA setting, we give new quantum polynomial-time distinguishers on $(3d-3)$-round Type-1 GFS with branches $d \geq 3$, which gain $d-2$ more rounds than the previous distinguishers. Hence, we could get better key-recovery attacks, whose time complexities gain a factor of $2^{\frac{(d-2)n}{2}}$. In qCCA setting, we get $(3d-3)$-round quantum distinguishers on Type-1 GFS, which gain $d-1$ more rounds than the previous distinguishers.

In addition, we give some quantum attacks on CAST-256 block cipher. We find 12-round and 13-round polynomial-time quantum distinguishers in qCPA and qCCA settings, respectively, while the best previous one is only 7 rounds. Hence, we could derive quantum key-recovery attack on 19-round CAST-256. While the best previous quantum key-recovery attack is on 16 rounds. When comparing our quantum attacks with classical attacks, our result also reaches 16 rounds on CAST-256 with 128-bit key under a competitive complexity.

Keywords: Generalized Feistel Scheme, Quantum attack, Simon's algorithm, CAST-256

1 Introduction

Feistel block ciphers are featured by the efficient Feistel network, whose encryption and decryption process are based on similar operations. This design has been extensively researched [1–4] and adopted by many standard block ciphers, such as DES, Triple-DES, Camellia [5], GOST [6]. Feistel network was also generalized to form Generalized Feistel Networks (GFN), which adopts more branches and different operations between the branches. At CRYPTO 1989, Zheng et al. [7] summarised some Generalized Feistel Networks as 3 types GFN, namely Type-1, Type-2 and Type-3 GFN. In addition, some other Generalized Feistel Networks were invented by Anderson and Biham [8], Lucks [9] and Schneier and Kelsey [10]. Many important primitives employed GFN, such as block ciphers CAST-256 [11], CLEFIA [12], Simpira [13] as well as hash functions MD5 and SHA-1. The Generalized Feistel Network inherits the advantages of Feistel Network. Besides, it allows a small round function to construct a cipher with a larger block size, which is beneficial to lightweight implementations. In this paper, we focus on Type-1 and Type-2 GFN. As the important block cipher CAST-256 and CLEFIA adopt Type-1 and Type-2 GFN, respectively, we also denote Type-1 GFN as CAST256-like GFN and denote Type-2 GFN as CLEFIA-like GFN.

Classically, Luby and Rackoff [14] proved that a three-round Feistel scheme is a secure pseudo-random permutation. At Asiacrypt 2000, Moriai and Vaudenay [15] studied some Generalized Feistel Schemes (GFS) and proved a 7-round 4-branch CAST256-like GFS and 5-round 4-branch CLEFIA-like GFS are secure pseudo-random permutations. Later, Hoang and Rogaway [16] improved and generalised the provable-security analysis of Type-1, Type-2 and Type-3 GFS. Generic attacks on those constructions are also widely studied, such as birthday attack [17], meet-in-the-middle attack [18], differential attacks [19, 20], and Patarin et al.'s attacks [21–23].

Recently, the security evaluation of symmetric ciphers against quantum adversaries becomes a hot topic in communities. In the 2000s, it was a common belief that quantum attacks on symmetric primitives are of minor concern, as they mainly consist of employing Grover's algorithm [24] to generically speed up search (sub-)problems. However, Kuwakado and Morii [25] found the first the polynomial-time quantum distinguisher on 3-round Feistel using quantum period finding algorithm, i.e., Simon's algorithm [26]. Later, various quantum attacks against symmetric primitives were invented, such as key-recovery attacks

against Even-Mansour constructions [27], forgery or key-recovery attacks against block cipher based MACs [28, 29], key-recovery attacks against FX constructions [30], and so on.

According to Zhandry's work [31], there are two different models for quantum cryptanalysis against symmetric ciphers, i.e., standard security (also denoted as Q1) and quantum security (also denoted as Q2). In Q1 model, the adversaries could only collect data classically and processes them with local quantum computers. While in Q2 model, the adversaries could query the oracle with quantum superpositions of inputs, and obtain the corresponding superposition of outputs. The Q2 model is theoretically interesting. Moreover, as stated by Ito et al. [32], *"the threat of this attack model becomes significant if an adversary has access to its white-box implementation. Because arbitrary classical circuit can be converted into quantum one, the adversary can construct a quantum circuit from the classical source code given by the white-box implementation"*. In this paper, we assume that the adversaries come from Q2 model.

There are already some papers investigate Feistel schemes or GFS against Q2 adversaries. Besides Kuwakado and Morii [25]'s work, Ito et al. [32] extended the quantum distinguisher to 4-round Feistel construction under quantum chosen-ciphertext setting. Based on the Grover-meet-Simon algorithm by Leander and May [30], Hosoyamada et al. [33] and Dong et al. [34] introduced some quantum key-recovery attacks on Feistel schemes. Dong et al. [35] gave some quantum distinguishers and key-recovery attacks on some GFS. Dong et al. [36] and Bonnetain et al. [37] studied 2K-/4K-Feistel schemes against quantum slide attacks. Notably, Hosoyamada and Iwata [38] proved a tight quantum security bound of the 4-Round Luby-Rackoff construction recently.

Table 1. Rounds of quantum distinguishers on Type-1 GFS

Source	Setting	r	$d=3$	$d=4$	$d=5$	$d=6$	$d=7$	...
[35]	qCPA	$2d-1$	5	7	9	11	13	...
Section 4.1	qCPA	$3d-3$	6	9	12	15	18	...
Section 4.2	qCCA	$3d-2$	7	10	13	16	19	...

Table 2. Key-recovery attacks on Type-1 GFS ($d \geq 3$) in quantum settings

Source	Distinguisher	Key-recovery rounds	Complexity (log)	Trivial bound (log)
[35]	$2d-1$	$r \geq d^2$	$T + \frac{(r-d^2+d-2)n}{2}$	$\frac{rn}{2}$
Section 4.1	$3d-3$	$r \geq d^2$	$T + \frac{(r-d^2)n}{2}$	$\frac{rn}{2}$

T: $(\frac{1}{2}d^2 - \frac{3}{2}d + 2) \cdot \frac{n}{2}$

Table 3. Quantum attacks on CAST-256†

Source	Setting	Distinguisher	Attacked rounds					
			$r=14$	$r=15$	$r=16$	$r=17$	$r=18$	$r=19$
[35]	qCPA	7	2^{74}	$2^{92.5}$	2^{111}	–	–	–
Section 5.1	qCPA	12	2^{37}	$2^{55.5}$	2^{74}	$2^{92.5}$	2^{111}	–
Section 5.2	qCCA	13	$2^{18.5}$	2^{37}	$2^{55.5}$	2^{74}	$2^{92.5}$	2^{111}

†: Note that for 256-bit key size version, the trivial bound is 2^{128} by Grover algorithm.

Our contribution

In this paper, we give some improve attacks on Type-1 Generalized Feistel Schemes (GFS) in Q2 model with quantum chosen-plaintext attack (qCPA) setting and quantum chosen-ciphertext attack (qCCA) setting, respectively.

First, in qCPA setting, we give new quantum polynomial-time distinguishers on $(3d-3)$-round Type-1 GFS with branches $d \geq 3$, which gain $d-2$ more rounds than the previous distinguishers. Based on Leander and May's algorithm [30], we could get better key-recovery attacks, whose time complexities gain a factor of $2^{\frac{(d-2)n}{2}}$. Second, when considering qCCA setting, we get $(3d-3)$-round quantum distinguishers

Table 4. Comparisons between the classical and quantum attacks on CAST-256

Source	Key	Attack	round	Data	Time
[39]	128	boomerang	16	$2^{49.3}$	–
Section 5.2	128	qCCA	16	–	$2^{55.5}$
[40]	192	linear	24	$2^{124.1}$	$2^{156.52}$
Section 5.2	192	qCCA	17	–	2^{74}
[41]	256	multidim.ZC	28	$2^{98.8}$	$2^{246.9}$
Section 5.2	256	qCCA	19	–	2^{111}

on Type-1 GFS, which gain $d-1$ more rounds than the previous distinguishers. The distinguishers and the key-recovery attacks on Type-1 GFS are summarized in Table 1 and 2.

In addition, we also evaluate CAST-256 block cipher in qCPA and qCCA settings. We find 12-round and 13-round polynomial-time quantum distinguishers in qCPA and qCCA settings, respectively. Note that the best previous one is 7 rounds. Hence, we could derive quantum key-recovery attack on 19-round CAST-256. While the best previous quantum key-recovery attack is on 16 rounds. The results are summarized in Table 3. We also compare our quantum attacks with classical attacks in Table 4. When the key size of CAST-256 is 128, our result also reaches 16 rounds with a competitive complexity.

2 Notations

x_j^0 the jth branch in the input;
x_j^i the jth branch in the output of ith round, $i \geq 1$, $j \geq 1$;
d the branch number of CAST256-like GFS;
$2d$ the branch number of RC6/CLEFIA-like GFS;
R^i the ith ($i \geq 1$) round function of Type-1 (CAST256-like) GFS, the input and output are n-bit string, n-bit key is absorbed by R^i;
R_j^i the jth ($1 \leq j \leq d$) round function in the ith ($i \geq 1$) round function of Type-2 (RC6/CLEFIA -like) GFS, the input and output are n-bit string, n-bit key is absorbed by R_j^i.

3 Related works

3.1 Simon's algorithm

Given a function f $\{0,1\}^n \to \{0,1\}^n$, that is known to be invariant under some n-bit XOR period a, find a. In other words, find a by given: $f(x) = f(y) \leftrightarrow x \oplus y \in \{0^n, a\}$.

Classically, the optimal time to solve the problem is $\mathcal{O}(2^{n/2})$. However, Simon [26] gives a quantum algorithm that provides exponential speedup and only requires $\mathcal{O}(n)$ quantum queries to find a. The algorithm includes five quantum steps:

I. Initializing two n-bit quantum registers to state $|0\rangle^{\otimes n}|0\rangle^{\otimes n}$, one applies Hadamard transform to the first register to attain an equal superposition:

$$H^{\otimes n}|0\rangle|0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle|0\rangle. \tag{1}$$

II. A quantum query to the function f maps this to the state

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

III. Measuring the second register, the first register collapses to the state:

$$\frac{1}{\sqrt{2}}(|z\rangle + |z \oplus a\rangle).$$

IV. Applying Hadamard transform to the first register, we get:

$$\frac{1}{\sqrt{2}} \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{y \cdot z}(1 + (-1)^{y \cdot a})|y\rangle.$$

V. The vectors y such that $y \cdot a = 1$ have amplitude 0. Hence, measuring the state yields a value y that $y \cdot a = 0$.

Repeat $\mathcal{O}(n)$ times, one obtains a by solving a system of linear equations. However, Kaplan et al. [28] and Santoli [42] showed that Simon's promise may be weakened at the cost of computing many vectors y that $y \cdot a = 0$. At Asiacrypt 2017, Leander and May [30] assume that $f(x)$ behaves as a random periodic function with period a, and show that any function value $f(x)$ has only two preimages with probability at least $\frac{1}{2}$. Moreover, they show that $l = 2(n + \sqrt{n})$ repetitions of the Simon's algorithm are sufficient to compute a. The probability is greater than $\frac{4}{5}$ that it contains at least $n-1$ linearly independent vectors y that are orthogonal to a (Lemma 4, [30]).

At ISIT 2010, Kuwakado and Morii [25] introduced a quantum distinguish attack on 3-round Feistel scheme using Simon's algorithm. As shown in Figure 1, α_0 and α_1 are arbitrary constants:

$$\begin{aligned} f : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\ b, x &\mapsto \alpha_b \oplus x_2^3, \text{ where } (x_1^3, x_2^3) = E(\alpha_b, x), \\ f(b,x) = \ & R^2(R^1(\alpha_b) \oplus x)). \end{aligned}$$

f is periodic function that $f(b,x) = f(b \oplus 1, x \oplus R^1(\alpha_0) \oplus R^1(\alpha_1))$. Then using Simon's algorithm, one can get the period $s = 1||R^1(\alpha_0) \oplus R^1(\alpha_1)$ in polynomial time.

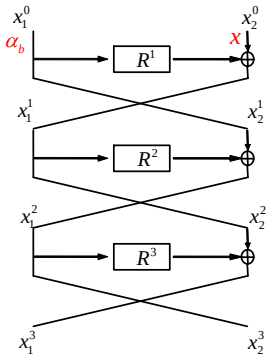

Fig. 1. 3-round quantum distinguisher

3.2 Hosoyamada and Sasaki's Method to Truncate Outputs of Quantum Oracles

As shown in Figure 1, in Kuwakado and Morii's quantum distinguisher, one has to truncate the output $2n$ bits of E to obtain the right half n bits, namely x_2^3. However, Kaplan et al. [28] and Hosoyamada et al. [33] pointed out that in quantum setting it is not trivial to truncated the entangled $2n$ qubits to n qubits, since the usual truncation destroys entanglements.

At SCN 2018, Hosoyamada and Sasaki [33] introduced a method to simulate truncation of outputs of quantum oracles without destroying quantum entanglements. Let $\mathcal{O} : |x\rangle|y\rangle|z\rangle|w\rangle \mapsto |x\rangle|y\rangle|z \oplus \mathcal{O}_L(x,y)\rangle|w \oplus \mathcal{O}_R(x,y)\rangle$ be the encryption oracle E, where $\mathcal{O}_L, \mathcal{O}_R$ denote the left n bits and right n bits of the complete encryption, respectively. The goal is to simulate oracle $\mathcal{O}_R : |x\rangle|y\rangle|w\rangle \mapsto |x\rangle|y\rangle|w \oplus \mathcal{O}_R(x,y)\rangle$. Hosoyamada and Sasaki first simulate a tweaked $\mathcal{O}_R$, i.e., $\mathcal{O}'_R : |x\rangle|y\rangle|w\rangle|0^n\rangle \mapsto |x\rangle|y\rangle|w \oplus \mathcal{O}_R(x,y)\rangle|0^n\rangle$. Let $|+\rangle := H^n|0^n\rangle$, where H^n is an n-bit Hadamard gate. Thus, $\mathcal{O}|x\rangle|y\rangle|+\rangle|w\rangle \mapsto |x\rangle|y\rangle|+\rangle|w \oplus \mathcal{O}_R(x,y)\rangle$. Then, they define $\mathcal{O}'_R := (I \otimes H^n) \circ \texttt{Swap} \circ \mathcal{O} \circ \texttt{Swap} \circ (I \otimes H^n)$, where $\texttt{Swap}$ is an operator that swaps last $2n$-qubits: $|x\rangle|y\rangle|z\rangle|w\rangle \mapsto |x\rangle|y\rangle|w\rangle|z\rangle$. So $\mathcal{O}'_R|x\rangle|y\rangle|w\rangle|0^n\rangle = |x\rangle|y\rangle|w \oplus \mathcal{O}_R(x,y)\rangle|0^n\rangle$. Hence, $\mathcal{O}_R$ could be simulated given the complete encryption oracle $\mathcal{O}$ using ancilla qubits.

3.3 Grover's algorithm

Given an unordered set of $N = 2^n$ items, Grover's algorithm is to find the unique element that satisfies some condition. In other words, given a quantum oracle $\mathcal{O}$ which performs the operation $\mathcal{O}|x\rangle = (-1)^{f(x)}|x\rangle$, where $f(x) = 0$ for all $0 \le x < 2^n$ except x_0, for which $f(x_0) = 1$, find x_0. While the best

classical algorithm for a search over unordered data requires $\mathcal{O}(N)$ time, Grover's algorithm performs the search on a quantum computer in only $\mathcal{O}(\sqrt{N})$ operations, a quadratic speedup. The steps of the algorithm are as follows:

1. Initialization of a n-bit register $|0\rangle^{\otimes n}$. Apply the Hadamard transform to the first register to attain an equal superposition that can be given as follows:

$$H^{\otimes n}|0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle = |\varphi\rangle. \tag{2}$$

2. Construct an oracle $\mathcal{O}$: $|x\rangle \to (-1)^{f(x)}|x\rangle$, where $f(x) = 1$ if x is the correct state; otherwise, $f(x) = 0$.
3. Define the Grover iteration as $(2|\varphi\rangle\langle\varphi| - I)\mathcal{O}$, and apply it $R \approx \frac{\pi}{4}\sqrt{2^n}$ times:

$$[(2|\varphi\rangle\langle\varphi| - I)\mathcal{O}]^R|\varphi\rangle \approx |x_0\rangle.$$

4. return x_0.

3.4 Combining Grover and Simon's algorithms

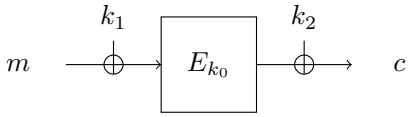

Fig. 2. FX constructions

At Asiacrypt 2017, Leander and May [30] gave a quantum key-recovery attack on FX-construction shown in Figure 2: $Enc(x) = E_{k_0}(x + k_1) + k_2$. They introduce the function $f(k, x) = Enc(x) + E_k(x) = E_{k_0}(x + k_1) + k_2 + E_k(x)$. For the correct key guess $k = k_0$, we have $f(k, x) = f(k, x + k_1)$ for all x. However, for $k \neq k_0$, $f(k, \cdot)$ is not periodic. They *combine Simon and Grover algorithm* to attack FX ciphers in the quantum-CPA model with complexity roughly 2^{32}.

Based on Leander and May's work, Hosoyamada and Sasaki [33], and Dong and Wang [34] appended several rounds to the 3-round Feistel distinguisher in Figure 1 to recover the keys of an r-round Feistel cipher in time $\mathcal{O}(2^{(r-3)n/2})$. See the previous papers [33, 34] for details.

3.5 Ito et al.'s attack on Feistel cipher

At RSA 2019, Ito et al.'s [32] gives new quantum distinguisher on 4-round Feistel cipher in quantum chosen-ciphertext attack (qCCA) setting.

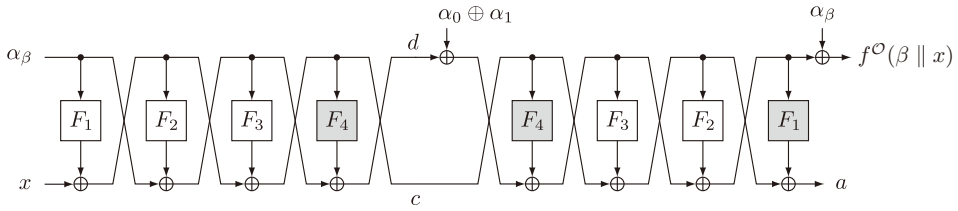

Fig. 3. Ito et al.'s 4-round quantum distinguisher on Feistel [32]

As shown in Figure 3, plaintext (α_β, x) is first encrypted by 4-round Feistel to get the ciphertext (d, c), then a new tweaked ciphertext $(d \oplus \alpha_0 \oplus \alpha_1, c)$ is decrypted by the inverse 4-round Feistel to get new plaintext. Then, Ito et al. defined function:

$$
\begin{aligned}
f^{\mathcal{O}} = & \alpha_0 \oplus \alpha_1 \oplus F_2(x \oplus F_1(\alpha_\beta)) \\
& \oplus F_2(x \oplus F_1(\alpha_\beta) \oplus F_3(\alpha_\beta \oplus F_2(x \oplus F_1(\alpha_\beta)))) \\
& \oplus F_3(\alpha_\beta \oplus \alpha_0 \oplus \alpha_1 \oplus F_2(x \oplus F_1(\alpha_\beta))),
\end{aligned} \tag{3}
$$

where β is 0 or 1, and α_β is constant. Hence, $f^{\mathcal{O}}$ has period $s = 1 \| F_1(\alpha_0) \oplus F_1(\alpha_1)$.

Combining with Leander and May's algorithm [30], Ito et al. gave key-recovery attacks on Feistel using the new distinguisher. In addition, in the key-recovery attacks, they did not to actually compute the period of $f^{\mathcal{O}}$, instead, they distinguish f by checking the dimension of the space spanned by the vectors given by the Simon's algorithm. Thus, there will not be a problem if there are several partial periods or periods other than s because it distinguishes f without computing s. In this paper, we also use this method to launch our key-recovery attacks on Type-1 GFS and CAST-256. For more details, we refer the readers to Ito et al.'s paper [32].

4 Quantum attack on Type-1 GFS

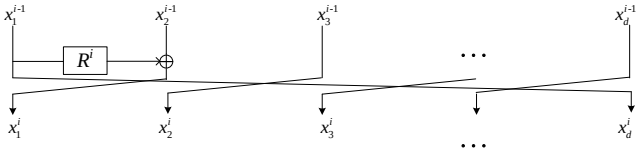

Fig. 4. Round i of Type-1 GFS with d branches

As shown in Figure 4, the input of the Type-1 cipher is divided into d branches, i.e. x_j^0 for $1 \le j \le d$, each of which has n-bit, so the blocksize is $d \times n$. R^i is the round function that absorbs n-bit secret key k_i and n-bit input.

Dong et al. [35] first considered some quantum distinguishers and key-recovery attacks on Type-1 GFS. In this section, we give some improved polynomial-time quantum distinguishers on Type-1 GFS in quantum chosen-plaintext attack (qCPA) setting and quantum chosen-ciphertext attack (qCCA) setting, respectively. Based on those distinguishers, some improved key-recovery attacks are achieved.

4.1 Quantum distinguishers on Type-1 GFS in qCPA setting

For the sake of clarity, we first give an example attack on Type-1 with $d = 4$, then extend to any case with $d \ge 3$. After that, the key-recovery attacks are given.

Example case of Type-1 with $d = 4$ in qCPA setting When $d = 4$, we have a 9-round quantum distinguisher as shown in Figure 5.

The encryption process of 9-round Type-1 GFS is $E(x_1^0, x_2^0, \alpha_b, x) = (x_1^9, x_2^9, x_3^9, x_4^9)$, where x_1^0 and x_2^0 are constants and $b = 0, 1$, α_0, α_1 are also constants, $\alpha_0 \neq \alpha_1$. Suppose $h(\alpha_b) = R^3(R^2(R^1(x_1^0) \oplus x_2^0) \oplus \alpha_b)$, Define $f(b, x) = x_2^9 \oplus \alpha_b = R^6(R^5(R^4(h(\alpha_b) \oplus x) \oplus x_1^0) \oplus R^1(x_1^0) \oplus x_2^0) \oplus R^2(R^1(x_1^0) \oplus x_2^0)$.

Hence, we deduce

$$
\begin{aligned}
f(0, x) &= R^6(R^5(R^4(h(\alpha_0) \oplus x) \oplus x_1^0) \oplus R^1(x_1^0) \oplus x_2^0) \oplus R^2(R^1(x_1^0) \oplus x_2^0) \\
&= f(1, x \oplus h(\alpha_0) \oplus h(\alpha_1)), \\
f(1, x) &= R^6(R^5(R^4(h(\alpha_1) \oplus x) \oplus x_1^0) \oplus R^1(x_1^0) \oplus x_2^0) \oplus R^2(R^1(x_1^0) \oplus x_2^0) \\
&= f(0, x \oplus h(\alpha_0) \oplus h(\alpha_1)).
\end{aligned} \tag{4}
$$

Thus, $f(b, x) = f(b \oplus 1, x \oplus h(\alpha_0) \oplus h(\alpha_1))$, $f(b, x)$ is a function with period $s = 1 \| h(\alpha_0) \oplus h(\alpha_1)$.

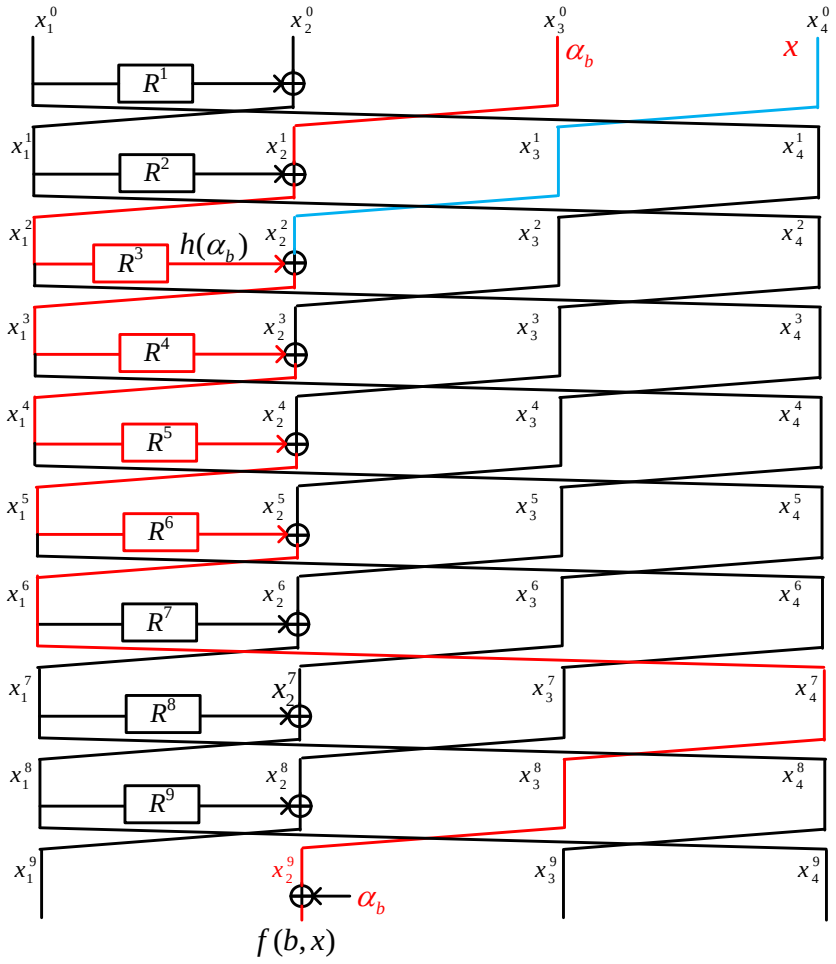

Fig. 5. 9-round distinguisher on Type-1 GFS with $d = 4$

Extending to any case of Type-1 with $d \geq 3$ in qCPA setting We construct the quantum distinguisher on the $(3d-3)$-round cipher. The intermediate state after the ith round is x_j^i for $1 \leq j \leq d$, especially the output of the $(3d-3)$th round is denoted as $x_1^{3d-3}||x_2^{3d-3}||...||x_d^{3d-3}$.

The encryption process of $(3d-3)$-round Type-1 GFS is $E(x_1^0, ..., x_{d-2}^0, \alpha_b, x)$, where $b = 0, 1$, and α_0, α_1 are arbitrary constants, $\alpha_0 \neq \alpha_1$, and $x_d^0 = x$. All remaining branches $x_1^0, x_2^0, ..., x_{d-2}^0$ are constants.

Similar to the attack on case $d = 4$, we also define $h(\alpha_b) = R^{d-1}(R^{d-2}(...R^3(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0)... \oplus x_{d-2}^0) \oplus \alpha_b)$.

Then, we define

$$f(b, x) = x_2^{3d-3} \oplus \alpha_b = R^{2d-2}(R^{2d-3}(...(h(\alpha_b) \oplus x)...) \oplus R^{d-3}(\ldots(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0 \ldots) \oplus x_{d-2}^0)) \\ \oplus R^{d-2}(R^{d-3}(\ldots(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0) \ldots) \oplus x_{d-2}^0) \tag{5}$$

With simple computation, we deduce $f(b, x) = f(b \oplus 1, x \oplus g(\alpha_0) \oplus g(\alpha_1))$. Therefore, function f satisfies Simon's promise with $s = 1||g(\alpha_0) \oplus g(\alpha_1)$. Thanks to Hosoyamada and Sasaki's work [33] shown in Sect. 3.2, we could truncate outputs of quantum oracles with ease, and hence f could be implemented as quantum oracle. The period s of f could be found using Simon's algorithm.

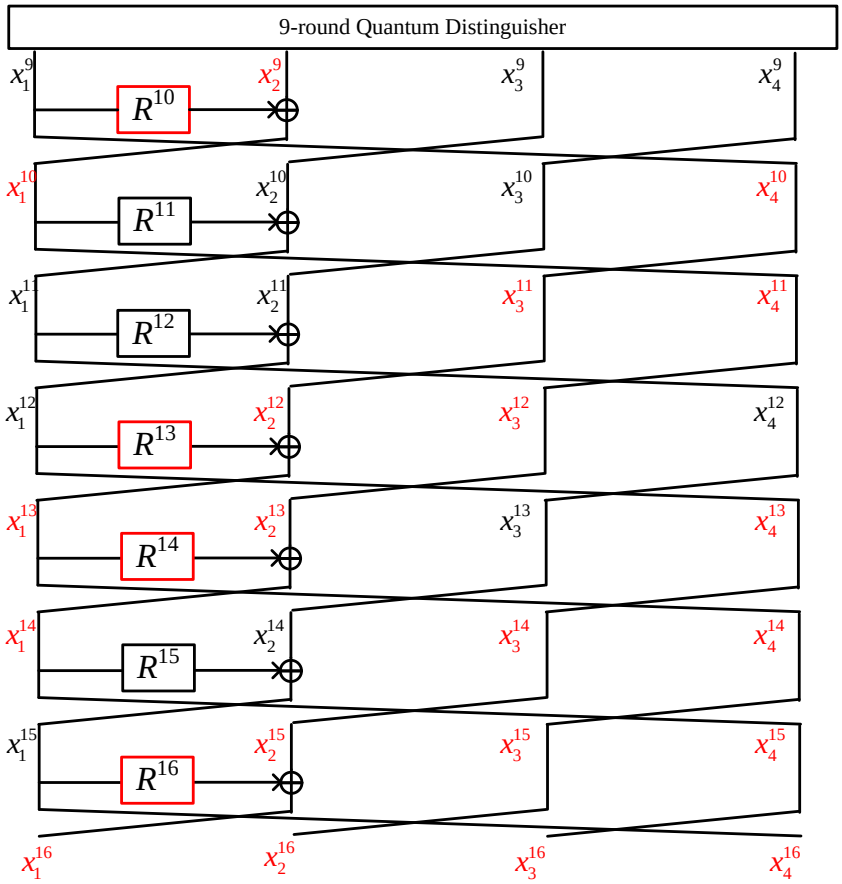

Fig. 6. 16-round quantum key-recovery attack on Type-1 GFS with $d = 4$

Quantum key-recovery attacks on Type-1 GFS in qCPA setting We give an example key-recovery attack on Type-1 GFS with $d = 4$ branches. Following the similar idea that combines Simon's and Grover's algorithms to attack Feistel structure [33, 34], we append 7 rounds under the 9-round distinguisher to launch the attack. As shown in Figure 6, there are $4n$-bit key needed to be guessed by Grover's algorithm

to compute x_2^9, which are highlighted in the red boxes of round functions. Note that, here we do not need Hosoyamada and Sasaki's method to truncate the output of encryption oracle. We just need to implement a function $h_D(k_{10}, k_{13}, k_{14}, k_{16}, x_1^{16}, x_2^{16}, x_3^{16}, x_4^{16}) = x_2^9$, where $k_{10}, k_{13}, k_{14}, k_{16}$ are round keys in $R^{10}, R^{13}, R^{14}, R^{16}$, , respectively. h_D decrypts $(x_1^{16}, x_2^{16}, x_3^{16}, x_4^{16})$ to get x_2^9 by guessing $k_{10}, k_{13}, k_{14}, k_{16}$.

Hence, the 16-round quantum key-recovery attack needs about 2^{2n} queries and $\mathcal{O}(n^2)$ qubits. If we attack $r > 16$ rounds, we need guess $(r-12)n$ key bits by Grover's algorithm. Thus, the the time complexity is $2^{\frac{(r-12)n}{2}}$.

Generally, for $d \geq 3$, we could get $(3d-3)$-round quantum distinguisher. We append $d^2 - 3d + 3$ rounds under the quantum distinguisher to attack $r_0 = d^2$ rounds Type-1 GFS. Similarly, we need to guess $(\frac{1}{2}d^2 - \frac{3}{2}d + 2)n$-bit key by Grover's algorithm. Thus, for r_0 rounds, the time complexity is $(\frac{1}{2}d^2 - \frac{3}{2}d + 2) \cdot \frac{n}{2}$ queries, and $\mathcal{O}(n^2)$ qubits are needed. If we attack $r > r_0$ rounds, we need guess $(\frac{1}{2}d^2 - \frac{3}{2}d + 2)n + (r - r_0)n$ key bits by Grover's algorithm. Thus, the time complexity is $2^{(\frac{1}{2}d^2 - \frac{3}{2}d + 2) \cdot \frac{n}{2} + \frac{(r-r_0)n}{2}}$.

4.2 Quantum distinguishers on Type-1 GFS in qCCA setting

Example case of Type-1 with $d = 4$ in qCCA setting When $d = 4$, we get 10-round quantum distinguisher as shown in Figure 7. Note that, for simplicity, we omit the last swap function.

The encryption process is $E(x_1^0, x_2^0, \alpha_b, x) = (x_1^{10}, x_2^{10}, x_3^{10}, x_4^{10})$, where $b = 0, 1$, and α_0, α_1 are arbitrary constants, $\alpha_0 \neq \alpha_1$, and $x_4^0 = x$. The branches x_1^0, x_2^0 are constants. The decryption is $D(x_1^{10}, x_2^{10} \oplus \alpha_0 \oplus \alpha_1, x_3^{10}, x_4^{10}) = (y_1^0, y_2^0, y_3^0, y_4^0)$, note that as shown in Figure 7, many internal states in the decryption phase (right side of Figure 7) are the same to that in the encryption phase (left side of Figure 7).

In the right side of Figure 7, following the red lines and blue lines, we define $f(b, x) = y_1^0 = R^4(R^7(g(b, x) \oplus \alpha_0 \oplus \alpha_1) \oplus R^7(g(b, x)) \oplus x_2^6) \oplus x_2^7$, where $g(b, x) = x_2^9$.

Then we try to compute the ANF of $g(b, x)$. We first denote $h(\alpha_b) = R^3(R^2(R^1(x_1^0) \oplus x_2^0) \oplus \alpha_b)$. Then, $g(b, x) = x_2^9 = R^6(R^5(R^4(h(\alpha_b) \oplus x) \oplus x_1^0) \oplus R^1(x_1^0) \oplus x_2^0) \oplus R^2(R^1(x_1^0) \oplus x_2^0) \oplus \alpha_b$. Through simple computation, we could deduce $g(0, x) \oplus \alpha_0 \oplus \alpha_1 = g(1, x \oplus h(\alpha_0) \oplus h(\alpha_1))$ and $g(1, x) \oplus \alpha_0 \oplus \alpha_1 = g(0, x \oplus h(\alpha_0) \oplus h(\alpha_1))$.

Hence, $g'(b, x) = R^7(g(b, x) \oplus \alpha_0 \oplus \alpha_1) \oplus R^7(g(b, x))$ is periodic, and the period is $1||h(\alpha_0) \oplus h(\alpha_1)$. Then, we rewrite $f(b, x) = R^4(g'(b, x) \oplus x_2^6) \oplus x_2^7$. Since $x_2^6 = h(\alpha_b) \oplus x_2^2 = h(\alpha_b) \oplus x$ holds, x_2^6 is a function on (b, x) with the same period to $g'(b, x)$, i.e. the period is also $1||h(\alpha_0) \oplus h(\alpha_1)$. Meanwhile, $x_2^7 = R^4(h(\alpha_b) \oplus x) \oplus x_1^0$, thus, it also has the same period $1||h(\alpha_0) \oplus h(\alpha_1)$.

So $f(b, x)$ is a function with period $1||h(\alpha_0) \oplus h(\alpha_1)$.

Extending to any case of Type-1 with $d \geq 3$ in qCCA setting We construct the new quantum distinguisher on the $(3d-2)$-round cipher. The intermediate state after the ith round is x_j^i for $1 \leq j \leq d$, especially the output of the $(3d-2)$th round is denoted as $x_1^{3d-2} \| x_2^{3d-2} \| ... \| x_d^{3d-2}$. The encryption process is $E(x_1^0, x_2^0, ..., x_{d-2}^0, \alpha_b, x) = (x_1^{3d-2}, x_2^{3d-2}, ..., x_d^{3d-2})$, where $b = 0, 1$, and α_0, α_1 are arbitrary constants, $\alpha_0 \neq \alpha_1$, and $x_d^0 = x$. The branches $x_1^0, x_2^0, ..., x_{d-2}^0$ are constants. So we also denote encryption process as $E(\alpha_b, x)$ for simplicity. The decryption is $D(x_1^{3d-2}, x_2^{3d-2} \oplus \alpha_0 \oplus \alpha_1, ..., x_d^{3d-2}) = (y_1^0, y_2^0, ..., y_d^0)$.

Define $g(b, x)$:

$$\begin{aligned} g(b, x) = x_2^{3d-3} = R^{2d-2}(\ldots(R^{d+1}(R^d(R^{d-1}(\ldots(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0) \ldots \oplus \alpha_b) \oplus x) \\ \oplus x_1^0) \oplus R^1(x_1^0) \oplus x_2^0) \ldots \oplus R^{d-3}(\ldots(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0) \ldots \oplus x_{d-3}^0) \oplus x_{d-2}^0) \\ \oplus R^{d-2}(R^{d-3}(\ldots(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0) \ldots \oplus x_{d-3}^0) \oplus x_{d-2}^0) \oplus \alpha_b. \end{aligned} \tag{6}$$

We also take the case $d = 4$ as an example shown in Figure 7, and we follow the red lines in the encryption process, and get

$$\begin{aligned} &x_i^{3d-2} = x_i^{3d-3} = R^{2d-4+i}(x_{i-1}^{3d-3}) \oplus x_2^{2d-5+i}, 3 \leq i \leq d, \\ &x_1^{3d-2} = x_1^{3d-3} = R^{3d-3}(x_d^{3d-3}) \oplus x_2^{3d-4}, \\ &x_2^{3d-2} = R^{3d-2}(x_1^{3d-2}) \oplus x_2^{3d-3}. \end{aligned} \tag{7}$$

Following the red lines in the decryption process of Figure 7, and get

$$y_1^0 = R^d(R^{2d-1}(x_2^{3d-3} \oplus \alpha_0 \oplus \alpha_1) \oplus x_3^{3d-2}) \oplus R^{2d}(x_3^{3d-2}) \oplus x_4^{3d-2}. \tag{8}$$

We construct function f as following:

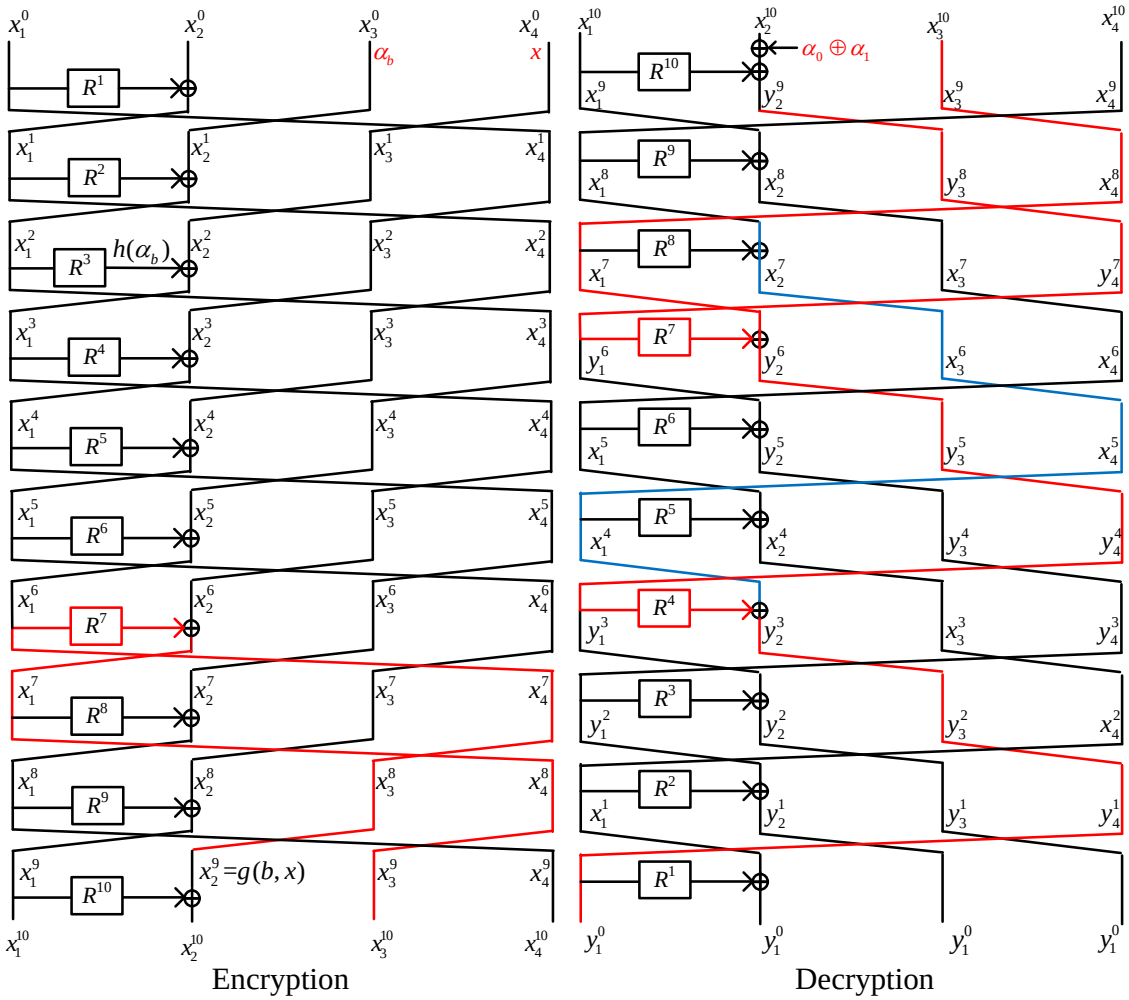

Fig. 7. 10-round distinguisher on CAST256-like GFS with $d = 4$

$$\begin{aligned} f : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\ b, x &\mapsto y_1^0, \text{ where } (x_1^{3d-2}, x_2^{3d-2}, ..., x_d^{3d-2}) = E(\alpha_b, x), \\ (y_1^0, y_2^0, ..., y_d^0) &= D(x_1^{3d-2}, x_2^{3d-2} \oplus \alpha_0 \oplus \alpha_1, ..., x_d^{3d-2}), \\ f(b,x) &= R^d(R^{2d-1}(x_2^{3d-3} \oplus \alpha_0 \oplus \alpha_1) \oplus x_3^{3d-2}) \oplus R^{2d}(x_3^{3d-2}) \oplus x_4^{3d-2}. \end{aligned}$$

f is derived in two steps: first, encrypting $(x_1^0, x_2^0, ..., x_{d-2}^0, \alpha_b, x)$ to get the cipher $(x_1^{3d-2}, x_2^{3d-2}, ..., x_d^{3d-2})$, second decrypting $(x_1^{3d-2}, x_2^{3d-2} \oplus \alpha_0 \oplus \alpha_1, ..., x_d^{3d-2})$ to get the plaintext $(y_1^0, y_2^0, ..., y_d^0)$, and we define $f = y_1^0$.

Define $h(\alpha_b) = R^{d-1}(R^{d-2}(R^{d-3}(\ldots(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0)\ldots \oplus x_{d-3}^0) \oplus x_{d-2}^0) \oplus \alpha_b)$. As shown in equation (6), we can know $g(0,x) = g(1, x \oplus h(\alpha_0) \oplus h(\alpha_1)) \oplus \alpha_0 \oplus \alpha_1$, and $g(0,x) \oplus \alpha_0 \oplus \alpha_1 = g(1, x \oplus h(\alpha_0) \oplus h(\alpha_1))$. So $g(b,x)$ has period $1||h(\alpha_0) \oplus h(\alpha_1)$. Meanwhile, $x_2^{2d-2} = R^{d-1}(R^{d-2}(R^{d-3}(\ldots(R^2(R^1(x_1^0) \oplus x_2^0) \oplus x_3^0)\ldots \oplus x_{d-3}^0) \oplus x_{d-2}^0) \oplus \alpha_b) \oplus x = h(\alpha_b) \oplus x$, we get the period of x_2^{2d-2} is $1||h(\alpha_0) \oplus h(\alpha_1)$. Similarly, in f function, according to the first equation of equations (7), $R^{2d}(x_3^{3d-2}) \oplus x_4^{3d-2} = x_2^{2d-1} = R^d(x_2^{2d-2}) \oplus x_1^0$ has the same period. Moreover, in f function, through equation (7), $R^{2d-1}(x_2^{3d-3} \oplus \alpha_0 \oplus \alpha_1) \oplus x_3^{3d-2} = R^{2d-1}(x_2^{3d-3} \oplus \alpha_0 \oplus \alpha_1) \oplus R^{2d-1}(x_2^{3d-3}) \oplus x_2^{2d-2} = R^{2d-1}(g(b,x) \oplus \alpha_0 \oplus \alpha_1) \oplus R^{2d-1}(g(b,x)) \oplus x_2^{2d-2}$ is periodic. Hence, $f(b,x) = f(b \oplus 1, x \oplus h(\alpha_0) \oplus h(\alpha_1))$, the period $s = 1||h(\alpha_0) \oplus h(\alpha_1)$,

Quantum key-recovery attacks on Type-1 GFS in qCCA setting We add $r-3d+2$ rounds before the $(3d-2)$-round distinguisher to launch the key-recovery attack by Leander and May's algorithm [30].

The attack procedures are as follows:

1. Run the quantum circuit that takes the intermediate state value $(x_1^{r-3d+2}, x_2^{r-3d+2}, ..., x_{d-2}^{r-3d+2}, \alpha_b, x)$ after the first $(r-3d+2)$ rounds and the subkeys for the first $(r-3d+2)$ rounds as input, and decrypt the first $(r-3d+2)$ rounds get the plaintext. Then use the encryption oracle E encrypt the plaintext $(x_1^0, x_2^0, ..., x_d^0)$ to get the ciphertext $(x_1^r, x_2^r, ..., x_d^r)$.
2. Run the quantum circuit, which takes the ciphertext $(x_1^r, x_2^r \oplus \alpha_0 \oplus \alpha_1, ..., x_d^r)$ and the $(r-3d+2)$ rounds subkeys as input. Make quantum decryption query D of the ciphertext to get the plaintext, and use the the plaintext and subkeys to decrypt the last $(r-3d+2)$ rounds to get the intermediate state $(y_1^{r-3d+2}, y_2^{r-3d+2}, ..., y_d^{r-3d+2})$.
3. Guess the subkeys of the first $(r-3d+2)$ rounds. For each guessed subkey, use the the $(3d-2)$ rounds distinguisher in E and D to check its correctness. If the distinguisher is a periodic permutation, then judge that the guess is correct. Otherwise judge that the guess is wrong.

For the r $(r > 3d-2)$ round, there are $(r-3d+2)n$-bit key needed to be guessed by Grover's algorithm. So the r-round quantum key-recovery attack needs about $2^{\frac{r-3d+2}{2}n}$ time and $\mathcal{O}(n^2)$ qubits.

5 Quantum attacks on CAST-256 block cipher

CAST-256 block cipher is a first-round AES candidate. It is composed of 48 rounds, including 24 rounds Type-1 GFN and 24 rounds inverse Type-1 GFN, as shown in Figure 8, which is composed of 9-round Type-1 GFN and 3-round inverse Type-1 GFN. The block size is 128 bits, which are divided into four 32-bit branches and the key size can be 128, 192 or 256 bits. Each round function absorbs 37-bit subkey. Our attack is quit general and does not need any other details of the cipher.

In this section, we give two quantum attacks on CAST-256 block cipher in qCPA setting and qCCA setting, respectively.

5.1 Quantum attack on CAST-256 in qCPA setting

As shown in Figure 8, we construct 12-round quantum distinguisher on CAST-256, which includes 9-round Type-1 GFN and 3-round inverse Type-1 GFN. The distinguisher is very similar to the 9-round distinguisher in Section 4.1.

Suppose $h(\alpha_b) = R^3(R^2(R^1(x_1^0) \oplus x_2^0) \oplus \alpha_b)$, Define $f(b,x) = x_2^{12} \oplus \alpha_b = R^6(R^5(R^4(h(\alpha_b) \oplus x) \oplus x_1^0) \oplus R^1(x_1^0) \oplus x_2^0) \oplus R^2(R^1(x_1^0) \oplus x_2^0)$. As shown in Section 4.1, $f(b,x) = f(b \oplus 1, x \oplus h(\alpha_0) \oplus h(\alpha_1))$, $f(b,x)$ is a function with period $s = 1 \| h(\alpha_0) \oplus h(\alpha_1)$.

As shown in Figure 9, when attacking r $(r > 12)$ rounds CAST-256, we have to guess all the subkeys in the last $r-12$ rounds, i.e. $(r-12) \times 37$-bit key. Thus, about $2^{\frac{(r-12)\times 37}{2}} = 2^{18.5r-222}$ Grover iterations are needed.

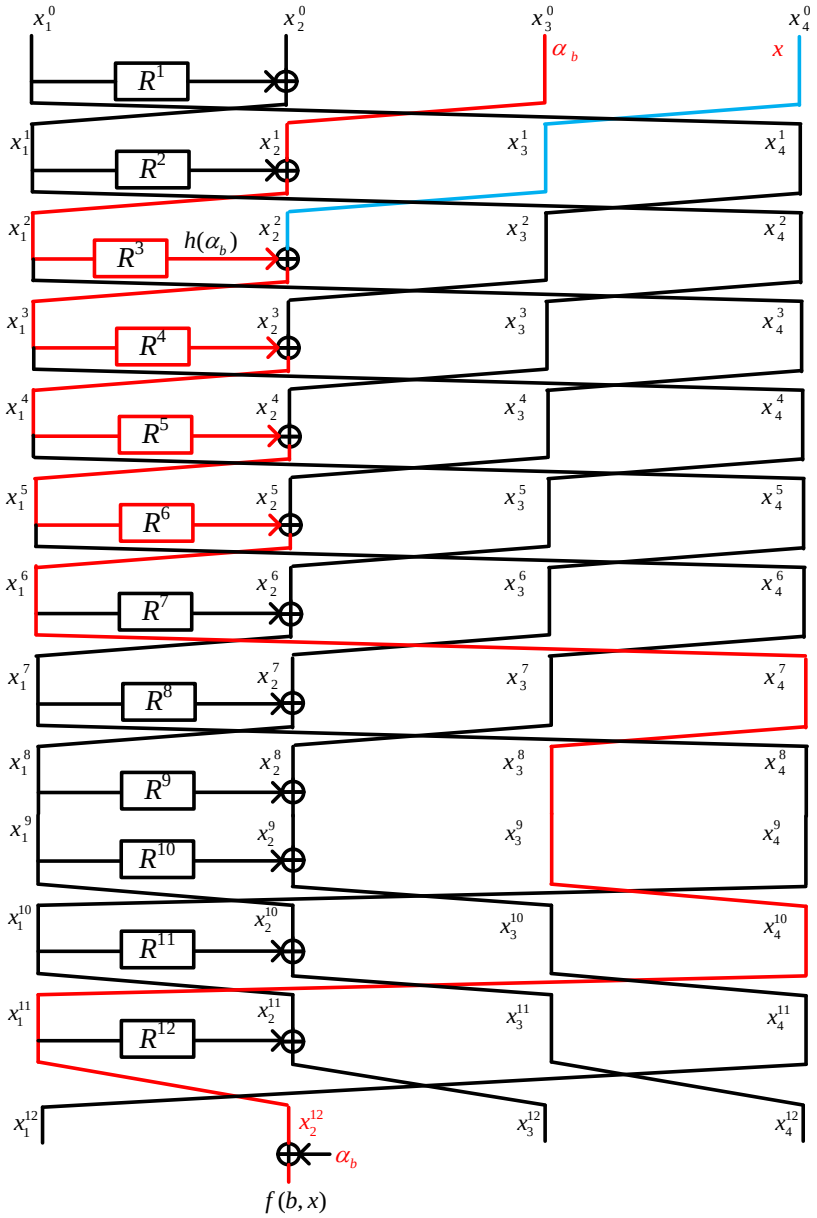

Fig. 8. 12-round distinguisher on CAST-256

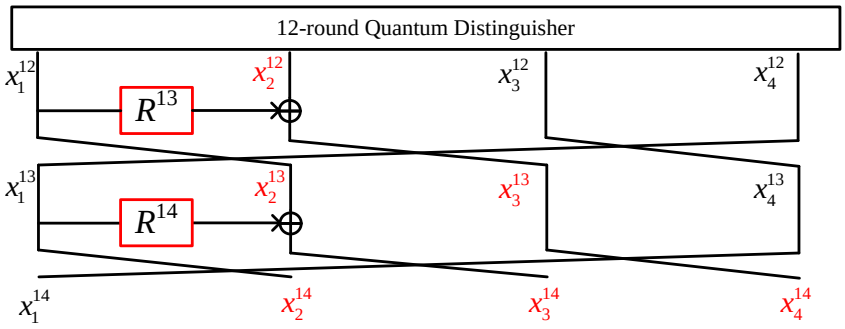

Fig. 9. 14-round attack on CAST-256

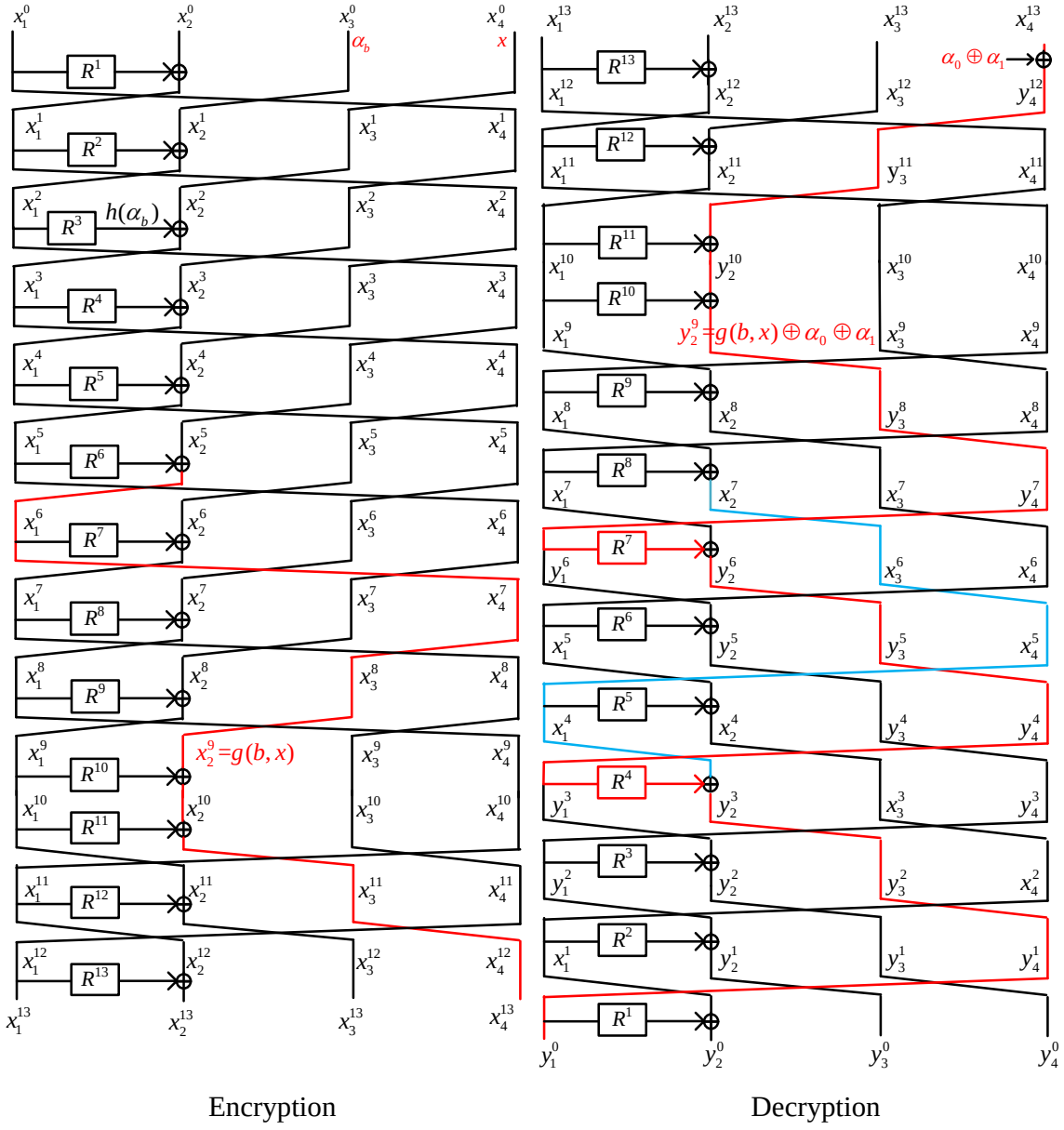

Fig. 10. 13-round distinguisher on CAST-256

5.2 Quantum attack on CAST-256 in qCCA setting

We construct a 13-round quantum distinguisher in qCCA setting as shown in Figure 10. The distinguisher is very similar to the 10-round distinguisher of Type-1 GFS in Section 4.2.

The encryption process is $E(x_1^0, x_2^0, \alpha_b, x) = (x_1^{13}, x_2^{13}, x_3^{13}, x_4^{13})$, where $b = 0, 1$, and α_0, α_1 are arbitrary constants, $\alpha_0 \neq \alpha_1$, and $x_4^0 = x$. The branches x_1^0, x_2^0 are constants. The decryption is $D(x_1^{13}, x_2^{13}, x_3^{13}, x_4^{13} \oplus \alpha_0 \oplus \alpha_1) = (y_1^0, y_2^0, y_3^0, y_4^0)$. Note that as shown in Figure 10, many internal states in the decryption phase (right side of Figure 10) are the same to that in the encryption phase (left side of Figure 10).

In the right side of Figure 10, following the red lines and blue lines, we define $f(b, x) = y_1^0 = R^4(R^7(g(b, x) \oplus \alpha_0 \oplus \alpha_1) \oplus x_1^7) \oplus x_2^7$, where $g(b, x) = x_2^9$. From the left side of Figure 10, we find $x_1^7 = R^7(g(b, x)) \oplus x_2^6$.

Then we try to compute the ANF of $g(b, x)$. We first denote $h(\alpha_b) = R^3(R^2(R^1(x_1^0) \oplus x_2^0) \oplus \alpha_b)$. Then, $g(b, x) = x_2^9 = R^6(R^5(R^4(h(\alpha_b) \oplus x) \oplus x_1^0) \oplus R^1(x_1^0) \oplus x_2^0) \oplus R^2(R^1(x_1^0) \oplus x_2^0) \oplus \alpha_b$. Similar to Section 4.2, $g(b, x)$ is a function with period $1||h(\alpha_0) \oplus h(\alpha_1)$. Meanwhile, $x_1^7 = R^7(g(b, x)) \oplus x_2^6 = R^7(g(b, x)) \oplus h(\alpha_b) \oplus x$ is also has period $1||h(\alpha_0) \oplus h(\alpha_1)$.

Hence, $g'(b, x) = R^7(g(b, x) \oplus \alpha_0 \oplus \alpha_1) \oplus x_1^7$ is also periodic. Then, we rewrite $f(b, x) = R^4(g'(b, x)) \oplus x_2^7$. Since, $x_2^7 = R^4(h(\alpha_b) \oplus x) \oplus x_1^0$, thus, it also has the same period $1||h(\alpha_0) \oplus h(\alpha_1)$. So $f(b, x)$ is a function with period $1||h(\alpha_0) \oplus h(\alpha_1)$.

Similar to the key-recovery attack in Section 4.2, when attacking r ($r > 13$) rounds CAST-256 using the 13-round distinguisher, we have to guess all the subkeys in the first $r - 13$ rounds, i.e. $(r - 13) \times 37$-bit key. Thus, about $2^{\frac{(r-13)\times 37}{2}} = 2^{18.5r-240.5}$ Grover iterations are needed.

6 Conclusion

In this paper, we give more improved polynomial-time quantum distinguishers on Type-1 GFS in quantum chosen-plaintext attack (qCPA) setting and quantum chosen-ciphertext attack (qCCA) setting. First, we give new qCPA quantum distinguishers on $(3d - 3)$-round Type-1 GFS with branches $d \geq 3$, which gain $d-2$ more rounds than the previous distinguishers. Hence, we could get better key-recovery attacks, whose time complexities gain a factor of $2^{\frac{(d-2)n}{2}}$. We also get $(3d - 3)$-round qCCA quantum distinguishers on Type-1 GFS, which gain $d - 1$ more rounds than the previous distinguishers. In addition, we also discuss some quantum attacks on CAST-256 block cipher.

References

1. Lars R. Knudsen. The security of feistel ciphers with six rounds or less. *J. Cryptology*, 15(3):207–222, 2002.
2. Takanori Isobe and Kyoji Shibutani. Generic key recovery attack on feistel scheme. *IACR Cryptology ePrint Archive*, 2015:526, 2015.
3. Jian Guo, Jérémy Jean, Ivica Nikolic, and Yu Sasaki. Meet-in-the-middle attacks on generic feistel constructions. In *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, pages 458–477, 2014.
4. Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. New attacks on feistel structures with improved memory complexities. In *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, pages 433–454, 2015.
5. Kazumaro Aoki, Tetsuya Ichikawa, Masayuki Kanda, Mitsuru Matsui, Shiho Moriai, Junko Nakajima, and Toshio Tokita. Camellia: A 128-Bit Block Cipher Suitable for Multiple Platforms - Design and Analysis. In Douglas R. Stinson and Stafford E. Tavares, editors, *SAC 2000*, volume 2012 of *Lecture Notes in Computer Science*, pages 39–56. Springer, 2001.
6. National soviet bureau of standards. information processing system - cryptographic protection - cryptographic algorithm gost 28147-89 (1989).
7. Yuliang Zheng, Tsutomu Matsumoto, and Hideki Imai. On the construction of block ciphers provably secure and not relying on any unproved hypotheses. In *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*, pages 461–480, 1989.
8. Ross J. Anderson and Eli Biham. Two practical and provably secure block ciphers: BEARS and LION. In *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 21-23, 1996, Proceedings*, pages 113–120, 1996.
9. Stefan Lucks. Faster luby-rackoff ciphers. In *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 21-23, 1996, Proceedings*, pages 189–203, 1996.

10. Bruce Schneier and John Kelsey. Unbalanced feistel networks and block cipher design. In *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 21-23, 1996, Proceedings*, pages 121–144, 1996.
11. First AES candidate conference. http://csrc.nist.gov/archive/aes/round1/conf1/aes1conf.htm.
12. Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata. The 128-bit blockcipher CLEFIA (extended abstract). In *Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers*, pages 181–195, 2007.
13. Shay Gueron and Nicky Mouha. Simpira v2: A family of efficient permutations using the AES round function. In *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, pages 95–125, 2016.
14. Michael Luby and Charles Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SIAM J. Comput.*, 17(2):373–386, April 1988.
15. Shiho Moriai and Serge Vaudenay. On the pseudorandomness of top-level schemes of block ciphers. In *Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings*, pages 289–302, 2000.
16. Viet Tung Hoang and Phillip Rogaway. On generalized feistel networks. In *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*, pages 613–630, 2010.
17. Charanjit S. Jutla. Generalized birthday arracks on unbalanced feistel networks. In *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, pages 186–199, 1998.
18. Jian Guo, Jérémy Jean, Ivica Nikolic, and Yu Sasaki. Meet-in-the-middle attacks on classes of contracting and expanding feistel constructions. *IACR Trans. Symmetric Cryptol.*, 2016(2):307–337, 2016.
19. Valérie Nachef, Emmanuel Volte, and Jacques Patarin. Differential attacks on generalized feistel schemes. In *Cryptology and Network Security - 12th International Conference, CANS 2013, Paraty, Brazil, November 20-22. 2013. Proceedings*, pages 1–19, 2013.
20. Ivan Tjuawinata, Tao Huang, and Hongjun Wu. Improved differential cryptanalysis on generalized feistel schemes. In *Progress in Cryptology - INDOCRYPT 2017 - 18th International Conference on Cryptology in India, Chennai, India, December 10-13, 2017, Proceedings*, pages 302–324, 2017.
21. Jacques Patarin, Valérie Nachef, and Côme Berbain. Generic attacks on unbalanced feistel schemes with contracting functions. In *Advances in Cryptology - ASIACRYPT 2006, 12th International Conference on the Theory and Application of Cryptology and Information Security, Shanghai, China, December 3-7, 2006, Proceedings*, pages 396–411, 2006.
22. Jacques Patarin, Valérie Nachef, and Côme Berbain. Generic attacks on unbalanced feistel schemes with expanding functions. In *Advances in Cryptology - ASIACRYPT 2007, 13th International Conference on the Theory and Application of Cryptology and Information Security, Kuching, Malaysia, December 2-6, 2007, Proceedings*, pages 325–341, 2007.
23. Emmanuel Volte, Valérie Nachef, and Jacques Patarin. Improved generic attacks on unbalanced feistel schemes with expanding functions. In *Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings*, pages 94–111, 2010.
24. Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 212–219, 1996.
25. Hidenori Kuwakado and Masakatu Morii. Quantum distinguisher between the 3-round feistel cipher and the random permutation. In *IEEE International Symposium on Information Theory, ISIT 2010, June 13-18, 2010, Austin, Texas, USA, Proceedings*, pages 2682–2685, 2010.
26. Daniel R. Simon. On the power of quantum computation. *SIAM J. Comput.*, 26(5):1474–1483, 1997.
27. Hidenori Kuwakado and Masakatu Morii. Security on the quantum-type even-mansour cipher. In *Proceedings of the International Symposium on Information Theory and its Applications, ISITA 2012, Honolulu, HI, USA, October 28-31, 2012*, pages 312–316, 2012.
28. Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. In *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, pages 207–237, 2016.
29. Xavier Bonnetain. Quantum key-recovery on full AEZ. In *Selected Areas in Cryptography - SAC 2017 - 24th International Conference, Ottawa, ON, Canada, August 16-18, 2017, Revised Selected Papers*, pages 394–406, 2017.
30. Gregor Leander and Alexander May. Grover meets simon - quantumly attacking the fx-construction. In *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II*, pages 161–178, 2017.

31. Mark Zhandry. How to construct quantum random functions. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 679–687, 2012.
32. Gembu Ito, Akinori Hosoyamada, Ryutaroh Matsumoto, Yu Sasaki, and Tetsu Iwata. Quantum chosen-ciphertext attacks against feistel ciphers. In *Topics in Cryptology - CT-RSA 2019 - The Cryptographers' Track at the RSA Conference 2019, San Francisco, CA, USA, March 4-8, 2019, Proceedings*, pages 391–411, 2019.
33. Akinori Hosoyamada and Yu Sasaki. Quantum demiric-selçuk meet-in-the-middle attacks: Applications to 6-round generic feistel constructions. In *Security and Cryptography for Networks - 11th International Conference, SCN 2018, Amalfi, Italy, September 5-7, 2018, Proceedings*, pages 386–403, 2018.
34. Xiaoyang Dong and Xiaoyun Wang. Quantum key-recovery attack on feistel structures. *SCIENCE CHINA Information Sciences*, 61(10):102501:1–102501:7, 2018.
35. Xiaoyang Dong, Zheng Li, and Xiaoyun Wang. Quantum cryptanalysis on some generalized feistel schemes. *IACR Cryptology ePrint Archive*, 2017:1249, 2017.
36. Xiaoyang Dong, Bingyou Dong, and Xiaoyun Wang. Quantum attacks on some feistel block ciphers. Cryptology ePrint Archive, Report 2018/504, 2018. https://eprint.iacr.org/2018/504.
37. Xavier Bonnetain, Mara Naya-Plasencia, and Andr Schrottenloher. On quantum slide attacks. Cryptology ePrint Archive, Report 2018/1067, 2018. https://eprint.iacr.org/2018/1067.
38. Akinori Hosoyamada and Tetsu Iwata. Tight quantum security bound of the 4-round luby-rackoff construction. Cryptology ePrint Archive, Report 2019/243, 2019. https://eprint.iacr.org/2019/243.
39. David A. Wagner. The boomerang attack. In *Fast Software Encryption, 6th International Workshop, FSE '99, Rome, Italy, March 24-26, 1999, Proceedings*, pages 156–170, 1999.
40. Meiqin Wang, Xiaoyun Wang, and Changhui Hu. New linear cryptanalytic results of reduced-round of CAST-128 and CAST-256. In *Selected Areas in Cryptography, 15th International Workshop, SAC 2008, Sackville, New Brunswick, Canada, August 14-15, Revised Selected Papers*, pages 429–441, 2008.
41. Andrey Bogdanov, Gregor Leander, Kaisa Nyberg, and Meiqin Wang. Integral and multidimensional linear distinguishers with correlation zero. In *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, pages 244–261, 2012.
42. Thomas Santoli and Christian Schaffner. Using simon's algorithm to attack symmetric-key cryptographic primitives. *Quantum Information & Computation*, 17(1&2):65–78, 2017.